

José Douglas Bertazzoni Zambianco

Sistema para Monitoramento Detalhado do Consumo Elétrico Residencial Utilizando Rede de Sensores ZigBee

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Eletrônica, Microeletrônica e Optoeletrônica.

Orientador: José Antonio Siqueira Dias

Campinas, SP
2010

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Z14s Zambianco, José Douglas Bertazzoni
Sistema para monitoramento detalhado do consumo
elétrico residencial utilizando rede de sensores ZigBee /
José Douglas Bertazzoni Zambianco. - -
Campinas, SP: [s.n.], 2010.

Orientador: José Antonio Siqueira Dias.
Dissertação de Mestrado - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Redes de computação. 2. Sensores. 3. Energia
elétrica - Consumo. I. Dias, José Antonio Siqueira. II.
Universidade Estadual de Campinas. Faculdade de
Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: System to detailed monitoring of residencial electical power
consumption using ZigBee Sensors network

Palavras-chave em Inglês: Computer networks, Sensors, Electric power consumption

Área de concentração: Eletrônica, Microeletrônica e Optoeletrônica

Titulação: Mestre em Engenharia Elétrica

Banca Examinadora: Elnatan Chagas Ferreira, Vicente Idalberto Becerra Sablón

Data da defesa: 30/09/2010

Programa de Pós-Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: José Douglas Bertazzoni Zambianco

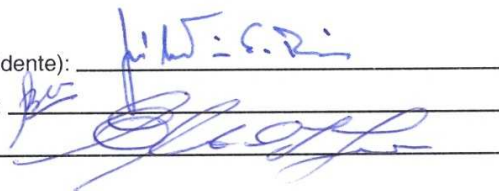
Data da Defesa: 30 de setembro de 2010

Título da Tese: "Sistema para monitoramento detalhado do consumo elétrico residencial utilizando rede de sensores ZigBee"

Prof. Dr. José Antonio Siqueira Dias (Presidente):

Prof. Dr. Vicente Idalberto Becerra Sablón:

Prof. Dr. Elnatan Chagas Ferreira:



Resumo

O setor elétrico vem buscando cada vez mais a melhoria da qualidade dos serviços prestados a seus consumidores. Dessa forma, há uma necessidade de entender o consumo elétrico de cada cliente, onde o perfil individualizado pode trazer informações importantes quanto a utilização da energia elétrica, tornando possível a melhoria constante para melhor atender este consumidor. Neste trabalho, é apresentado um projeto, seu desenvolvimento e implementação de uma rede de sensores monitorando o consumo de cada ponto de uma casa, considerando tomadas, lâmpadas e chuveiros afim de trazer a porcentagem de uso de cada ponto de consumo.

Palavras-chave: Medidores elétricos, ZigBee, Rede de Sensores.

Abstract

The electrical segment is always looking for better quality of services offered to customers. There is a demand of understand the consumption of electrical energy by each customer, it can bring an individualized profile that can show important informations related to the energy usage and how make it better. This works presents a project design, the development and implementation of a sensors network monitoring the power consumption of each point of a house, considering wall sockets, jack, electrical shower and lamps to take the percentage of each point usage.

Keywords: Electric meters, ZigBee, Sensor Networks.

Agradecimentos

A Deus pela oportunidade de realização desta obra.

Ao meu orientador Prof. José Antônio Siqueira Dias, sou grato pela orientação.

Ao Prof. Elnatan Chagas e ao Luis Duarte pela ajuda em todo o projeto.

Aos meus pais José e Maria pelo apoio durante esta jornada.

A AES Eletropaulo pelo apoio financeiro através do projeto P&D ANEEL contrato UNICAMP/AES Eletropaulo número 4200006254.

A minha família.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
Glossário	xii
1 Introdução	1
1.1 Consumo Energético e a Motivação do Projeto	1
1.2 Rede de Sensores sem Fio	3
1.3 Medidas Elétricas	4
1.4 Objetivo do Projeto	4
2 Redes de Comunicação e suas Tecnologias	6
2.1 Introdução	6
2.2 Sentido da Transmissão	6
2.3 Topologia	7
2.4 Classificação de Redes	10
2.5 Tipos de Redes Móveis	11
2.5.1 Redes Infra-Estruturadas	11
2.5.2 Redes <i>Ad-Hoc</i>	12
2.6 Comparação de Tecnologias	13
2.7 O Modelo ISO/OSI	17
3 A Rede ZigBee	20
3.1 Introdução	20
3.2 Arquitetura	21
3.2.1 A Camada Física	21
3.2.2 A Camada MAC	24
3.2.3 A Camada de Rede	29
3.2.4 A Camada de Aplicação	31
3.3 Os Elementos de Rede	34
3.4 Topologias de Rede	35
3.5 Inicialização dos Dispositivos	35
3.5.1 Inicialização de uma nova <i>PAN</i>	35

3.5.2	Principais iterações de uma <i>PAN</i>	36
3.6	Modulação do Sistema <i>ZigBee</i>	36
4	Desenvolvimento do Projeto	38
4.1	Introdução	38
4.2	<i>Hardware</i>	38
4.2.1	Medidores em Tomadas e Aparelhos Eletrônicos	41
4.2.2	Medidas em Lâmpadas	42
4.2.3	Medidas no Chuveiro	42
4.2.4	Interfaces	43
4.3	<i>Firmware</i>	44
4.3.1	Plataforma	44
4.3.2	Inicialização do Sistema	47
4.3.3	Configuração e Funcionamento do Sistema de Medida em Tomadas	48
4.3.4	Configuração e Funcionamento do Sistema de Medida em Lâmpadas	49
4.3.5	Configuração e Funcionamento do Sistema de Medida em Chuveiros	50
4.3.6	Armazenamento dos Dados	52
4.4	<i>USB Stick e Coleta de Dados</i>	54
4.4.1	<i>USB Driver</i> e Emulação de uma Porta Serial	54
4.4.2	<i>Software</i> de Comunicação entre o computador e o <i>USB Stick</i>	55
4.5	<i>Software</i> de Interface Gráfica com o Usuário	56
4.6	Otimização do Consumo dos Módulos	60
5	Resultados Experimentais	64
5.1	Introdução	64
5.2	Metodologia do Teste	64
5.2.1	A Rede	64
5.2.2	Os Sensores	66
5.2.3	Configuração do <i>Software</i> de Interface com o Usuário	66
5.3	Resultados Esperados	67
5.4	Resultados	68
6	Considerações Finais e Trabalhos Futuros	71
6.1	Trabalhos Futuros	72
	Referências bibliográficas	73

Lista de Figuras

1.1	Consumo percentual de energia elétrica residencial.	2
1.2	Diagrama do Sistema	4
2.1	Sentido de Transmissão	7
2.2	Topologia <i>Peer to Peer</i>	7
2.3	Topologia Barramento	8
2.4	Topologia Anel	8
2.5	Topologia Estrela	9
2.6	Topologia Árvore	9
2.7	Topologia <i>Mesh</i>	10
2.8	Classificação de Redes	11
2.9	Configuração de uma Rede Infra-Estruturada	12
2.10	Configuração de uma Rede <i>Ad-Hoc</i>	13
2.11	Comparação da BER e Relação Sinal Ruído entre Tecnologias [5]	16
2.12	Camadas do Modelo OSI	17
3.1	Aplicações utilizando o modelo ZigBee	20
3.2	Camadas do Protocolo ZigBee	21
3.3	Faixas de Frequência de Operação do Padrão IEEE 802.15.4	22
3.4	Diagrama em Blocos da Camada Física [7].	23
3.5	Estrutura do Pacote da Camada Física [7].	24
3.6	Diagrama em Blocos da Camada MAC [7].	25
3.7	Troca de Mensagens numa Rede <i>Beacon</i> [7].	26
3.8	Troca de Mensagens numa Rede Não- <i>Beacon</i> [7].	26
3.9	Exemplo de um <i>Superframe</i> [7].	27
3.10	<i>Superframe</i> com inatividade [7].	27
3.11	Algoritmo CSMA-CA [7].	28
3.12	Diagrama em Blocos da Camada de Rede [5].	30
3.13	Diagrama em Blocos da Camada de Aplicação [5].	31
3.14	Diagrama em Blocos da Camada do Sistema [5].	32
3.15	Modulação <i>DSSS</i>	36
4.1	<i>Kit</i> de desenvolvimento ATAVRRZRAVEN [9]	38
4.2	Blocos do <i>Kit</i> de desenvolvimento ATAVRRZRAVEN [9]	39
4.3	Blocos do Projeto	40

4.4	Blocos do <i>Kit</i> de desenvolvimento ATAVRRZRAVEN [9] <i>USB Stick</i>	40
4.5	Medidor de Tomada	41
4.6	Diagrama Esquemático do Circuito Medidor de Tomada	41
4.7	Diagrama em Blocos do Circuito Medidor de Lâmpadas	42
4.8	Diagrama em Blocos do Circuito do Chuveiro	43
4.9	Interfaces	44
4.10	Pilha da Plataforma <i>Bitcloud</i> [15]	46
4.11	Operação Síncrona e Assíncrona [15]	47
4.12	Algoritmo de Inicialização do Sistema	48
4.13	Registrador EICRA [13]	48
4.14	Algoritmo de Funcionamento do <i>software</i> para o Consumo em Lâmpada	50
4.15	Algoritmo de Funcionamento do <i>software</i> para o Consumo em Chuveiro	51
4.16	Mapa de Memória <i>EEPROM</i> limpa dos endereços 0x00 ao 0x90	53
4.17	Mapa de Memória <i>EEPROM</i> da medida do sensor armazenada	53
4.18	Mapa de Memória <i>EEPROM</i> Coordenador	53
4.19	Porta Serial no Gerenciador de Dispositivos	54
4.20	Configuração da Porta Serial	55
4.21	Estrutura de Comunicação da Porta Serial	55
4.22	Requisição e Aquisição das Medidas do Computador para a Central	56
4.23	Arquivo de Saída com os Valores Medidos de cada Sensor	56
4.24	Tabela de Entrada das Configurações de Cada Sensor	57
4.25	Campos de Configuração de Cada Sensor	57
4.26	Comunicação entre os <i>Softwares</i> de Interface Gráfica e Interface <i>USB-Stick</i>	58
4.27	Relatório de Saída	58
4.28	Impacto Cada Módulo visto de Forma Gráfica	59
4.29	Previsão de Consumo e Custo para o Total do Mês	60
4.30	Circuito para o Monitoramento do Consumo do Módulo	60
4.31	Gráfico de Consumo de um Módulo	61
4.32	Gráfico de Consumo de um Módulo sem o ATMEGA3209P [12]	62
4.33	<i>Setup</i> de Bancada e Medida de Corrente de Entrada do Módulo com Multímetro	62
5.1	Disposição dos Sensores	65
5.2	Configuração do <i>Software</i> de Interface com o Usuário	67
5.3	Arquivo de Saída	68
5.4	Relatório Final do Teste	69
5.5	Gráfico do Teste Final	69
5.6	Rede Final	70

Lista de Tabelas

1.1	Os Resultados do IASC 2001 para o Estado de São Paulo.	2
2.1	Comparação de Tecnologias para uma rede de Sensores [5]	14
4.1	Configuração das Placas de Monitoramento	43
4.2	Pinagem utilizada para cada tipo de placa	44
4.3	Tipos de Interrupções	49
4.4	Configuração da Interrupção para Interface com o Medidor de Tomada	49
4.5	Configuração da Interrupção para Interface com o Medidor de Lâmpada	50
4.6	Configuração do Conversor Analógico/Digital	51
5.1	Configuração dos Módulos	66
5.2	Comparação entre os Valores Medidos e os Estimados	70

Glossário

AF Application Framework

AIB APS Information Base

APDU Application Protocol Data Unit

API Application Programming Interface

APS Application Support Sub-Layer

BE expoente das fatias, que é relacionado ao número de fatias que o dispositivo deve esperar antes de qualquer tentativa de acesso ao canal

Bluetooth Tecnologia de transmissão sem fio.

BPSK Binary Phase Shift Keying.

BSP Board Support Package

CAP Contention Access Period

CCA Clear Channel Assessment

CFP Contention Free Period

Coordenador Só pode ser implementado através de um dispositivo FFD. O coordenador é responsável pela inicialização, distribuição de endereços, manutenção da Rede, reconhecimento de todos os Nós, entre outras funções podendo servir como ponte entre várias outras Redes Zig-Bee

CSMA – CA Carrier Sense Multiple Access - Collision Avoidance

CW tamanho da janela, definindo então o número de fatias de tempo que precisam estar inativos antes que a transmissão seja inicializada

ED Energy Detection.

enddevice É onde os atuadores ou sensores serão hospedados. Pode ser implementado através de um dos dispositivos FFD ou RFD.

FFD Full Function Device (Dispositivos de Funções Completas).

FireWire High Performance Serial Bus/HPSB.

GTS Guaranteed Time Slot

HAL Hardware Abstraction Layer

IrDA Infrared Data Association.

IRQ Interrupt Request

ISM Industrial-Scientific Medical.

ISP In System Programming

JTAG Joint Test Action Group

LCD Liquid Crystal Display

LQI Link Quality Indication.

MACPIB MAC PAN Information Base

MANET Mobile Ad hoc NETWORK.

MCPS – SAP MAC Common Part Layer - Service Access Point

MCU MicroController Unit

MLME – SAP MAC Layer Management Entity - Service Access Point

NB número de tempos que o algoritmo *CMA-CA* foi usado durante a transmissão, sendo então inicializado com zero a todo início de transmissão

NLDE Network Layer Data Entity

NLDE – SAP Network Layer Data Entity - Service Access Point

NLME Network Layer Management Entity

NLME – SAP Network Layer Management Entity - Service Access Point

NWKIB Network Information Base

O – PQSK Offset-Quadrature Phase Shift Keying.

OSI Open Systems Interconnection.

PHYPIB Physical PAN Information Base

PLME Physical Layer Management Entity

PLME – SAP Physical PAN Information Base - Service Access Point

RF Rádio Frequencia.

RF – SAP Radio Frequency - Service Access Point

RFD Reduced Function Device (Dispositivos de Funções Reduzidas).

Roteador Só pode ser implementado através de um dispositivo FFD. Tem as características de um Nó normal na Rede, mas com poderes extras de também exercer a função de roteador intermediário entre nós, sem precisar do coordenador.

SSP Security Service Provider

USART Universal Synchronous Asynchronous Receiver Transmitter

USBStick Ponte de rede conectado a USB de um computador responsável pela conexão e coleta de dados da central

USB Universal Serial Bus.

UWB Ultra-wideband.

Z – Wave Tecnologia de transmissão sem fio proprietária.

ZDO ZigBee Device Object

Capítulo 1

Introdução

Este trabalho faz parte de um projeto aprovado pela Agência Nacional de Energia Elétrica (ANEEL), realizado com apoio da AES Eletropaulo, que encarregou a Faculdade de Engenharia Elétrica e de Computação da Unicamp, através do Departamento de Eletrônica e Microeletrônica (DEMIC), a efetuar estudos envolvendo consumo de energia elétrica residencial.

O desenvolvimento do trabalho possui não apenas a parte técnica da medida de energia, mas também faz a análise de todos os dados coletados, de forma que o consumo de energia possa ser detalhado para o usuário final ou consumidor.

Os requisitos básicos do projeto são:

- Realização de um projeto com protótipos funcionais de um sistema que possua as medidas de energia consumida de cada ponto de energia em uma residência;
- A comunicação no sistema deve ser sem fio;
- O relatório final com os dados de consumo deve possuir uma descrição do total de energia elétrica consumida por cada ponto do monitoramento;

1.1 Consumo Energético e a Motivação do Projeto

Dados levantados pela Companhia Paranaense de Energia Elétrica (COPEL) [1] demonstram que o consumo de energia elétrica residencial, de uma forma geral, está de acordo com a distribuição apresentada na Fig. 1.1. Pode-se verificar que, normalmente, os determinantes no consumo de uma residência são o chuveiro, a geladeira, a iluminação, a televisão, a máquina de lavar e o ferro elétrico.

O perfil de utilização de energia elétrica pode ser apresentado para o proprietário ou usuário de cada residência. Esta informação seria muito útil para o consumidor verificar em qual equipamento o consumo está maior, e tomar alguma atitude para corrigir a deficiência.

Dados da Eletropaulo [2] demonstram que muitos usuários fazem reclamações junto a operadora de energia elétrica pelo alto valor na sua conta de luz. O procedimento nesses casos, é a visita de um técnico responsável para averiguar se há algum problema na residência, o que normalmente não acontece gerando insatisfação do cliente para com a operadora.

A Tab. 1.1 é o Índice ANEEL de Satisfação do Consumidor (IASC) 2001 para o Estado de São Paulo, feita pela Agência Reguladora de Saneamento e Energia do Estado de São Paulo (ARSESP)

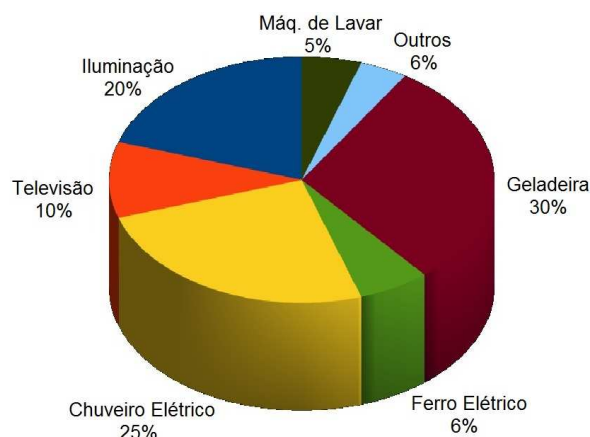


Fig. 1.1: Consumo percentual de energia elétrica residencial.

[3], para a classificação das operadoras, obtido através de entrevistas feitas para clientes de todas as operadoras de energia.

Tab. 1.1: Os Resultados do IASC 2001 para o Estado de São Paulo.

Classificação	Companhia	IASC
1	SANTA CRUZ	73,58
2	NACIONAL	72,26
3	MOCOCA	69,97
4	ELEKTRO	69,11
5	CAIUÁ	68,89
6	JAGUARI	68,01
7	EEVP	67,56
8	CPFL	67,55
9	EEB	66,13
10	CPEE	66,07
11	PIRATININGA	63,65
12	BANDEIRANTE	63,55
13	CSPE	60,30
14	ELETROPAULO	54,44

Este projeto tem por objetivo o detalhamento do consumo elétrico ao consumidor, para que possa tomar atitudes para a diminuição de sua conta de luz. Um exemplo simples: se luzes ficam acesas durante o dia e a noite, este consumo provavelmente terá um forte impacto no total consumido tornando assim a conta de luz mais alta, sem que o consumidor saiba que é isto que causa o gasto excessivo de energia. Outro exemplo seria o impacto no consumo causado por um determinado equipamento, como uma geladeira que pode estar com algum problema.

1.2 Rede de Sensores sem Fio

Uma rede de sensores sem fio pode ser definida como uma série de dispositivos independentes espalhados em uma determinada área que através de sensores estabelecem as medidas que devem ser entregues ou monitoradas. Todos os dispositivos se comunicam com um dispositivo central, que permite que os dados, sejam monitorados são armazenados, analisados, processados, etc. [4]. Uma rede de sensores pode conter uma quantidade muito grande de pontos ou nós que podem chegar a milhares, e pelo fato de ser sem fio, pode ser facilmente instalada e utilizada em locais onde o acesso é muito restrito e difícil. Outra característica importante, é o padrão ZigBee (802.15.4) [5] que estabelece o tipo de rede *ad-hoc* onde não há uma configuração prévia da rede fazendo com que a topologia seja criada *on the fly*.

Basicamente uma rede de sensores possui um coordenador, que é o elemento onde todos os valores medidos de outros sensores serão recebidos e armazenados, roteadores para a conexão com pontos muito distantes e os *endnodes* que são propriamente os elementos que possuem os sensores. Obviamente que roteadores e o próprio coordenador pode fazer medidas já que geralmente possuem os mesmos componentes dos *endnodes*, mas geralmente possuem apenas função de nós de rede já que o consumo de energia para apenas por serem coordenadores ou roteadores é maior por estarem com seus *transceivers* ligados por mais tempo.

Para que uma rede de sensores seja viável, alguns pontos precisam ser considerados [4]:

- **Tolerância a Falha:** Por serem de baixo custo, há risco de problemas devido a própria confiabilidade dos dispositivos. Dessa forma, o sistema deve suportar problemas como falta de energia, visibilidade de nós ou ainda danos físicos, fazendo com que a rede seja capaz de se auto reconfigurar e continuar funcionando mesmo com a perda de alguns nós.
- **Escalabilidade:** Os fatos de os nós serem de baixo custo, tamanho reduzido e pouco processamento interno contribuem para a formação de redes com uma escala muito alta, muito densa. Conseqüentemente deve-se ter um alto grau de redundância na transmissão de dados. Entretanto, isso pode ser um problema se considerarmos que quanto mais dados trafegando maior é a energia gasta. Dessa forma a agregação dos dados é uma tarefa muito importante para regiões da rede que possam diminuir esse tráfego dada a redundância de sinais.
- **Custo de produção:** Como dito anteriormente, a tendência de se ter um grande número de nós é muito grande, dessa forma o custo de cada nó deve ser o menor possível para que seja viável sua instalação comparada com uma rede de sensores com fio tradicional.
- **Consumo de energia:** Esse é outro fator de muita importância, pois nem todos os sensores possuem uma fonte de alimentação. Portanto, a economia de energia deve ser levada em conta para aplicações onde a alimentação dos sensores se dá através de baterias.

Uma rede de sensores pode ser utilizada para uma infinita variedade de possíveis soluções para automação e controle nas áreas residenciais, industriais, médica, militar, agrícola, etc.

1.3 Medidas Elébricas

O Projeto possui basicamente 3 tipos de medidas, e embora cada tipo de medida possuirá um circuito dedicado, irá utilizar os mesmos recursos de rede:

1. **Tomada** - O circuito da Tomada conta com um componente próprio para medidas de consumo e que pode ser diretamente ligado a rede elétrica junto a carga a ser medida. É baseado no C.I. medidor de energia elétrica AD71056 [6]. Dessa forma, cada dispositivo de consumo em uma casa (geladeira, televisão, etc.) possuirá um medidor to tipo Tomada em seu cabo de conexão com a rede elétrica.
2. **Lâmpada** - O circuito de medida de consumo das Lâmpadas é um medidor de intensidade de luz, assim quando uma lâmpada é ligada o circuito começa a contar o tempo que essa lâmpada ficou acesa. Portanto, toda lâmpada ou conjunto de lâmpadas deve possuir um dispositivo medidor de intensidade luminosa.
3. **Chuveiro** - Para a medida de consumo no chuveiro temos um outro problema que é o de instalação, pois não é recomendado nenhum tipo de intervenção na instalação elétrica da casa, dessa forma seria possível medir a corrente elétrica no chuveiro elétrico através de uma "garra" da mesma forma que um alicate amperímetro. A medida da corrente é proporcional ao valor da potência consumida pelo chuveiro.

Com os três tipos de medidores de consumo, todos os tipos de aparelho normalmente utilizados em uma residência poderão ser analisados e monitorados.

1.4 Objetivo do Projeto

O objetivo deste projeto é apresentar uma opção de monitoramento do consumo elétrico de uma residência, sendo apresentado ao consumidor uma lista com todos os pontos de monitoramento e seus respectivos consumos de eletrecidade.

A topologia básica da rede deve seguir a apresentada na Fig. 1.2, onde há uma central (coordenador) ligada a roteadores e *end devices*. Roteadores também podem fazer o papel de medidores mas para uma apresentação básica e introdutória da rede de sensores, uma topologia Estrela é mais simples de ser entendida. O detalhamento dos tipos de rede será abordado no Capítulo 2.

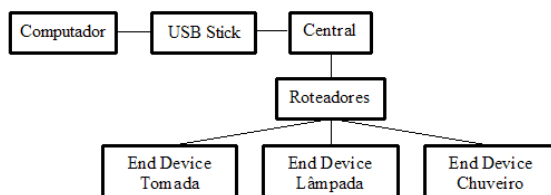


Fig. 1.2: Diagrama do Sistema

Dessa forma, todas as medidas de consumo são enviadas à central, a qual guarda todos os valores de cada ponto monitorado.

Um ponto de rede, chamado de *USB Stick*, é o responsável pela conexão entre a central e um computador. Assim, os valores de medidas armazenados na central é enviada ao computador para a geração do relatório de consumo de todos os dispositivos.

Capítulo 2

Redes de Comunicação e suas Tecnologias

2.1 Introdução

Atualmente, a transmissão de dados e informações tornaram-se imprescindíveis para indústrias, empresas, escolas, etc.

As redes variam muito, dependendo das necessidades de cada aplicação, podendo variar a velocidade, topologia, mobilidade entre outros pontos. Podemos citar alguns tipos de comunicação, indo desde os cabeados como a linha telefônica até as comunicações sem fio como o *Wi-Fi* e o utilizado no projeto *Zigbee* [5]. Obviamente locais onde não se tem problema quanto a espaço físico para cabos e tempo de instalação, a confiabilidade extrema uma rede cabeada é a mais indicada. Mas em outros casos, como na grande maioria das aplicações residenciais e, cada vez mais nas industriais, a comunicação sem fio é a mais indicada.

2.2 Sentido da Transmissão

A transmissão de dados pode ser feita de várias formas. Há tecnologias que operam utilizando apenas um sentido enviando comandos e informação para atuadores. Há também tecnologias onde todos os pontos podem transmitir e receber informação:

- ***Simplex***: é o tipo de ligação onde dados apenas circulam em um sentido, indo do emissor ao receptor. Normalmente utilizado quando não se tem a necessidade de dados trafegando nos dois sentidos como por exemplo uma impressão sendo enviada de um computador para uma impressora.
- ***Half Duplex***: é o tipo de ligação onde os dados circulam nos dois sentidos mas não os dois simultaneamente, sendo cada extremidade da ligação transmitindo por vez. Normalmente utilizada em conexões onde a taxa de comunicação é baixa e um comando pode esperar a ser processado como por exemplo rádios *Push-To-Talk*.
- ***Full Duplex***: é o tipo de ligação onde os dados circulam nos dois sentidos e simultaneamente. Dessa forma, cada extremidade da transmissão pode emitir e receber dados ao mesmo tempo. Podemos citar o caso de telefones celulares como exemplo.

A Fig. 2.1 ilustra os sentidos de transmissão.

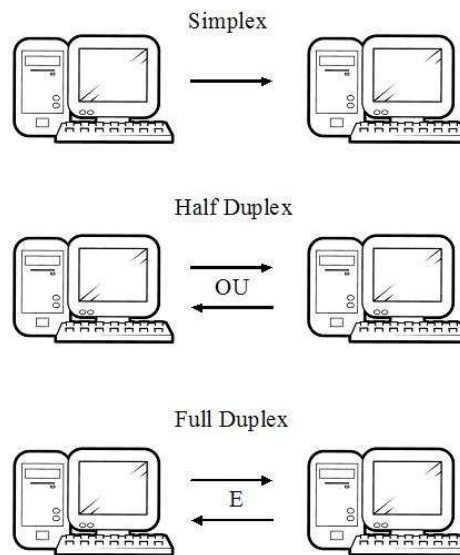


Fig. 2.1: Sentido de Transmissão

2.3 Topologia

A topologia de rede é a representação do tráfego de dados ou informações com os dispositivos conectados a esta rede.

Existem inúmeras topologias de rede que podem ser correlacionadas com a disposição física dos nós de rede ou a lógica das conexões.

Atualmente, os tipos de topologias mais utilizados são:

- **Peer to Peer:** é a topologia mais simples, onde apenas dois dispositivos estão conectados e é ilustrado na Fig. 2.2;

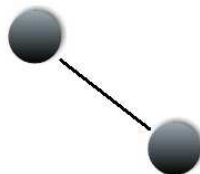


Fig. 2.2: Topologia *Peer to Peer*

- **Barramento:** esta topologia utiliza uma conexão física única entre todos os dispositivos da rede. Dessa forma sempre um dispositivo pode se comunicar por vez no barramento e todos

os outros pontos recebem a informação enviada, tornando a rede ocupada toda vez que um dispositivo envia informação. Muito utilizada em redes *Ethernet* mais antigas com cabo coaxial.

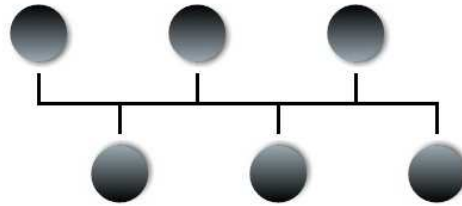


Fig. 2.3: Topologia Barramento

- **Anel:** neste tipo de topologia, as conexões são feitas em série, tendo portanto um circuito fechado no formato de um anel. Assim, o tráfego de dados acaba sendo em apenas uma direção passando de nó em nó até chegar em seu destino. Normalmente utilizada em enlaces de fibra óptica por ter ao menos uma redundância se algum nó de rede ou cabo apresentar falhas.

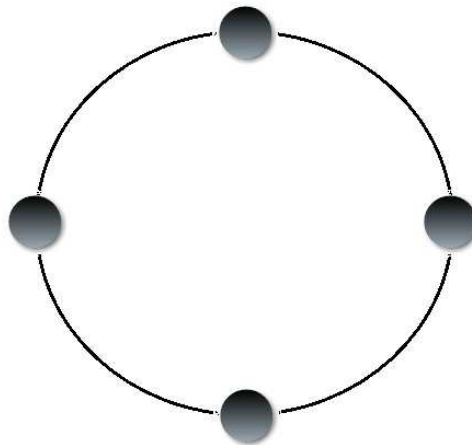


Fig. 2.4: Topologia Anel

- **Estrela:** é uma das mais usadas atualmente seja cabeada ou sem fio. Esta topologia consiste em um concentrador, ou central, definido como o ponto de rede onde toda informação é enviada ou recebida. Esta central tem a função de reenviar os dados para todas os dispositivos se comunicarem. Muito utilizada em pequenas redes residenciais ou em pequenos escritórios.

Uma rede ZigBee pode ser configurada para operar com topologia em estrela, dessa forma os nós de rede ficam concentrados e próximos ao coordenador que faria o papel da central juntamente com os pontos finais da rede ou *End nodes*;

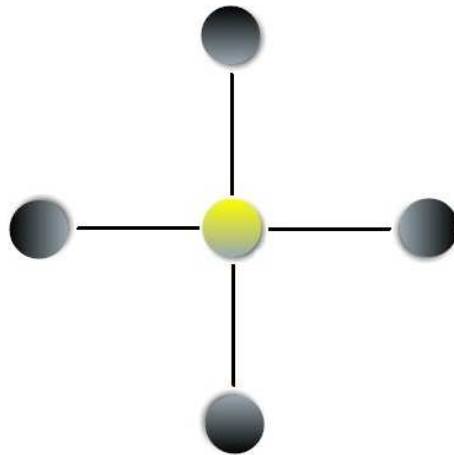


Fig. 2.5: Topologia Estrela

- **Árvore:** esta topologia pode ser descrita como ramificações de uma árvore genealógica, onde há uma "descendência" e uma "raiz". Dessa forma, redes em árvore podem se tornar muito grande com muitas ramificações. Na arquitetura ZigBee, uma rede com topologia em árvore é muito utilizada (ver Fig. 2.6);

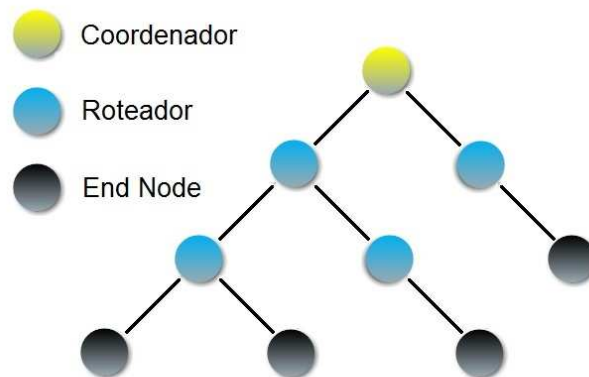
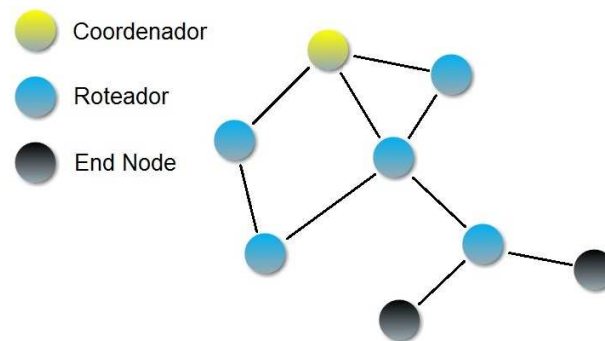


Fig. 2.6: Topologia Árvore

- **Mesh:** em ZigBee, esta é uma das configurações mais utilizadas pois é uma junção da topologia de Árvore e a Estrela. A grande vantagem é a utilização de dispositivos como se fossem repetidores para ampliar a área de cobertura (ver Fig. 2.7).

Fig. 2.7: Topologia *Mesh*

2.4 Classificação de Redes

Descrevendo brevemente cada classificação, bem como suas vantagens e desvantagens. A classificação das redes é baseada na área de abrangência do sinal transmitido, indo de redes pessoais com seus dispositivos de rede próximos, até redes com distribuição geográfica muito distante. A nomenclatura para redes sem fio, é feita com a adição de um "W" (de *wireless*) como, por exemplo, a WPAN.

- **PAN** - *Personal Area Network* é uma rede utilizada para a comunicação de dispositivos eletrônicos próximos uns dos outros, possuindo um alcance de poucos metros. Este tipo de rede normalmente compreende dispositivos pessoais para comunicação com dispositivos de outras pessoas, ou ainda para conexão com algum acesso a Internet. Dentre as tecnologias dentro de uma PAN estão a *USB*, *FireWire*, *IrDA*, *Bluetooth*, *UWB*, *Z-Wave* e *ZigBee*. O padrão para estas tecnologias é o IEEE 802.15 [7].
- **LAN** - *Local Area Network* é uma rede de pequena cobertura física, utilizada em residências, escritórios ou pequenos edifícios como uma escola. Normalmente uma LAN possui uma alta taxa de transferência de dados com pontos de rede distribuídos em pequenas áreas. Podem ser citadas como tecnologias muito utilizadas no passado dentre muitas a *ARCNET* e a *Token Ring*, atualmente as mais comuns são a *Ethernet* e a *Wi-Fi*. O padrão para estas tecnologias é o IEEE 802.11 [7].
- **MAN** - *Metropolitan Area Network* pode ser considerada uma versão ampliada de uma LAN pela semelhança de tecnologia aplicada. Por possuir um alcance maior na faixa dos 5km, ela pode ser utilizada por um grupo de vizinhos ou até mesmo uma cidade inteira. Pela alta taxa de transferência de dados, é capaz de transmitir vídeo, voz, etc. A mais conhecida tecnologia classificada como MAN é a *Wimax*. O padrão para estas tecnologias é o IEEE 802.16 [7].
- **WAN** - *Wide Area Network* é o tipo de rede utilizada pelas companhias de serviços telefônicos e servem para interligar cidades, países e continentes. A grande diferença para uma LAN ou MAN é a grande infra-estrutura utilizada nas centrais comutadoras, rádio terrestre ou ainda via

satélite. Como exemplos de tecnologias, pode-se citar a *X.25*, *Frame relay*, *ATM*, *Mobile-Fi*, etc. O padrão para estas tecnologias é o IEEE 802.20 [7].

- **RAN** - *Regional Area Network* é o tipo de rede utilizado para a interconexão de negócios, governos ou residências de uma determinada região geográfica. Essas conexões normalmente são caracterizadas pela alta velocidade de transmissão pela utilização de fibra óptica. Como exemplo, pode ser citada a *ATM*. O padrão para estas tecnologias é o IEEE 802.22 [7].

A classificação das redes pode ser vista na Fig. 2.8 que mostra principalmente o alcance de cada tipo de rede bem como sua nomenclatura de padrão e implementação.

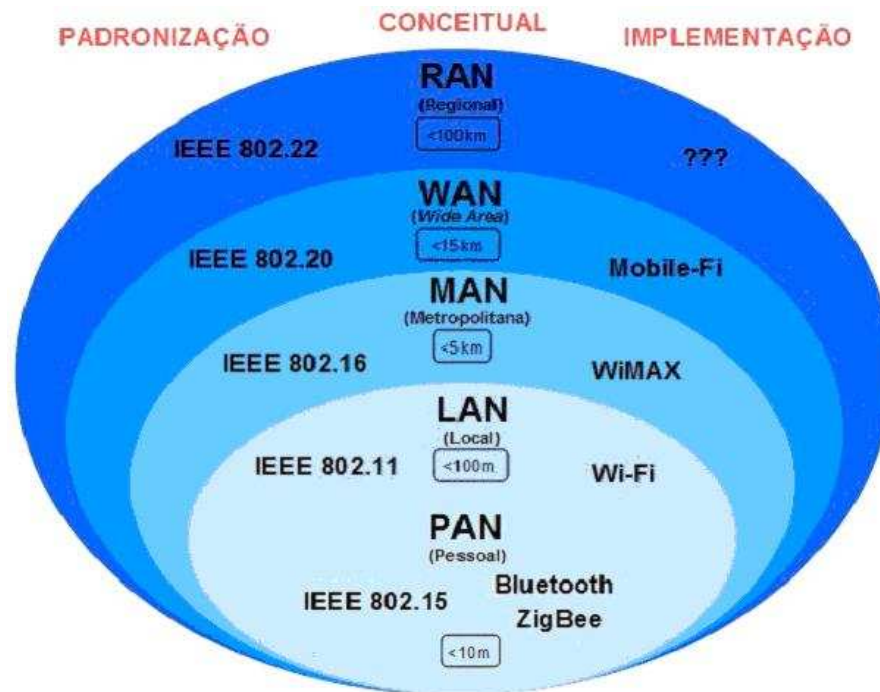


Fig. 2.8: Classificação de Redes

2.5 Tipos de Redes Móveis

Após o aperfeiçoamento das arquiteturas de rede, existem basicamente dois tipos de redes móveis sem fio: infra-estruturadas e *ad hoc*. Ambas serão discutidas pois são importantes no entendimento de como uma rede *ZigBee* é criada.

2.5.1 Redes Infra-Estruturadas

As redes infra-estruturadas utilizam uma configuração de arquitetura de rede em que o dispositivo móvel está sempre em contato direto com um ponto de acesso, o qual está ligado numa rede fixa.

Dessa forma, o funcionamento de uma rede infra-estruturada é parecida com uma rede de telefonia celular, onde cada telefone é conectado com uma central. A Fig. 2.9 ilustra como é a interconexão de uma rede infra-estruturada.

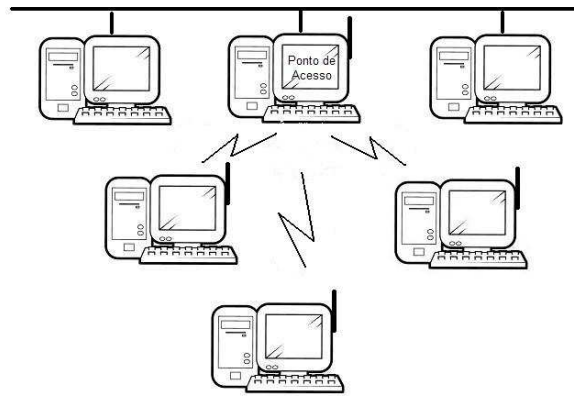


Fig. 2.9: Configuração de uma Rede Infra-Estruturada

Pode-se ver que mesmo se um ponto móvel quiser se comunicar com outro ponto móvel que está ao seu lado, a comunicação precisa ser feita através do Ponto de Acesso. Dessa forma, os nós móveis ficam impossibilitados de se comunicar de forma direta.

Na Fig. 2.9 pode-se ver que os pontos de acesso podem estar conectados a uma rede cabeada, assim podem servir de suporte para comunicação entre outros dispositivos que estão conectados a outras partes da rede.

2.5.2 Redes *Ad-Hoc*

As redes *Ad-Hoc* ou *MANET* tem a característica de todos os dispositivos serem capazes de se comunicar diretamente entre si. Nesse tipo de rede não há um ponto de acesso onde os dispositivos precisam se conectar para haver uma comunicação como na rede infra-estruturada. Dessa forma para que a rede se mantenha, os nós de rede acabam sendo dependentes uns dos outros. A Fig. 2.10 ilustra o funcionamento de uma rede *Ad-Hoc*.

Redes *Ad-Hoc* são indicadas em locais onde uma rede fixa não é possível e a configuração de uma rede infra-estruturada seja custosa e complicada. Neste projeto, temos uma rede *Ad-Hoc* dada a dificuldade de instalação dos nós, seja por problemas de passagem de fios ou pela complexidade de configuração para o instalador.

Como não há uma definição de pontos de acesso, os nós de rede são livres quanto a mobilidade tornando a topologia de rede imprevisível e podendo ser alterada com frequência, e conseqüentemente alterando a conectividade entre os dispositivos móveis frequentemente. Esse tipo de rede necessita de uma capacidade muito grande de adaptação e reconfiguração das rotas de comunicação e problemas quanto a consumo de energia e limitação de banda acabam surgindo.

Podemos então citar as vantagens e desvantagens de uma rede *Ad-Hoc* [8]:

- **Vantagens:**

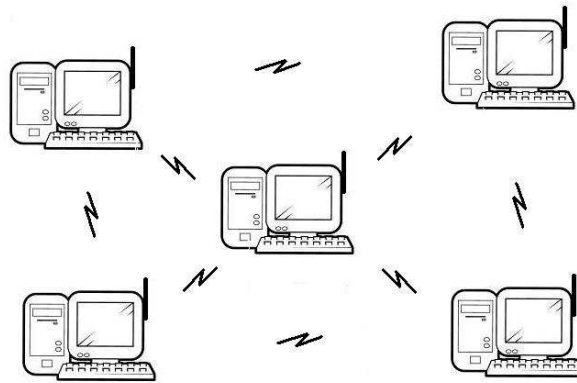


Fig. 2.10: Configuração de uma Rede *Ad-Hoc*

- **Instalação rápida:** Como o estabelecimento de uma rede *Ad-Hoc* é dinâmico, ela pode ser utilizada em situações ou locais que não possuem nenhum tipo de infra-estrutura ou instalação;
 - **Tolerância à falhas:** A adaptação e reconfiguração de rotas são constantes, qualquer perda de conectividade entre nós acaba não atrapalhando o funcionamento do sistema pois uma nova rota pode ser estabelecida fazendo com que falhas de transmissão entre nós possam ser fácil e rapidamente resolvidas;
 - **Conectividade:** A comunicação entre dois pontos de rede pode ser direta, havendo apenas a necessidade de estarem um ao alcance do outro;
 - **Mobilidade:** Nós podem se mover. Não há a necessidade de uma reconfiguração por parte do instalador pois a rede se encarrega de estabelecer as conexões entre os nós de rede, sendo esta a grande vantagem com relação a redes fixas.
- **Desvantagens:**
 - **Roteamento:** Para ser possível a mobilidade dos pontos de rede e possuir uma topologia completamente dinâmica, algoritmos de roteamento acabam aumentando o consumo dos dispositivos e principalmente o consumo de memória para alocação das tabelas de roteamento;
 - **Localização:** Como os dispositivos podem ser móveis e estarem se movendo, não há informação da posição geográfica deste dispositivo, diferentemente de redes previamente estruturadas onde a localização não é alterada;

2.6 Comparação de Tecnologias

A primeira definição de um projeto é a escolha da tecnologia a ser utilizada, levando em consideração pontos importantes de toda a rede implementada. Dessa forma, as principais tecnologias de comunicação aplicáveis ao projeto proposto podem ser comparadas numa tabela para que seja analisada de forma detalhada voltada para o melhor custo-benefício.

Tab. 2.1: Comparação de Tecnologias para uma rede de Sensores [5]

	ZigBee 802.15.4	WiFi 802.11b	Bluetooth 802.15.1
Complexidade	Simples	Muito Complexa	Complexa
Extensão	Sim	Não	Sim
Ciclo de Vida de Bateria	anos	horas	dias
Tamanho da Rede	65355	32	7
Taxa de Transmissão	20-250KB	11MB	720KB
Alcance Máximo (m)	100	100	10
Latência para Enumeração	30ms	3s	10s

A Tab. 2.1 compara as três principais tecnologias que poderiam ser utilizadas no projeto de monitoramento de consumo elétrico residencial.

- **Complexidade:** A complexidade do sistema é a abordagem dos mecanismos utilizados para a constituição de uma rede, seus protocolos, dificuldade de operação e instalação da rede. Neste quesito, este projeto precisa de uma baixa complexidade e precisa ser muito simples para ser configurada e de fácil instalação.

Dessa forma, o *ZigBee* é a tecnologia que mais se aplica, pois possui um protocolo muito simples em comparação com *WiFi* e *Bluetooth*. Dessa forma, a entrada de um nó na rede se torna muito mais rápida e simples. Isso implica diretamente na eficiência e energia gasta pelos dispositivos da rede principalmente na troca de informações, operações de associação e desassociação e principalmente na utilização do modo *sleep*. As vantagens de possuir um protocolo de baixa complexidade são o menor consumo de bateria e o menor tráfego de dados.

- **Extensão:** este quesito é relacionado a como um dispositivo novo pode entrar na rede, ou ainda um conjunto de dispositivos sendo adicionados a esta rede, bem como a migração de um dispositivo de uma determinada rede para outra.

Um dispositivo *Bluetooth* não possui esse tipo de funcionalidade muito importante para o projeto, já que o número de dispositivos não é previamente conhecido, pois o número de pontos podem variar dependendo das residências em centenas. Dessa forma, os dispositivos precisam ser de fácil associação, de movimentação e desassociação.

Uma rede *WiFi* possui mecanismos de mobilidade, mas isso deve ser previamente configurado, não se tornando de fácil instalação.

Já uma rede *ZigBee* foi projetada para este fim, onde novos dispositivos podem ser facilmente associados incluindo um conjunto de novos dispositivos. O mesmo vale para a mobilidade e desassociação, onde não há a necessidade de intervenção manual pois os dispositivos se reconfiguram automaticamente.

- **Ciclo de Vida de Bateria:** pela melhor performance e eficiência na utilização dos recursos de rede, a tecnologia *ZigBee* possui mecanismos para que a vida de uma simples bateria possa ser de anos.

Este tipo de característica é muito importante pois em muitos locais há a dificuldade de acesso e a durabilidade das baterias de alimentação é essencial para a viabilidade do projeto.

- **Tamanho da Rede:** O número de dispositivos é um número muito importante, numa residência de classe média alta esse número pode chegar facilmente a trezentos. Considerando as tecnologias *WiFi* e *Bluetooth*, isso seria impossível pois o número máximo de dispositivos são 32 e 7 respectivamente.

Obviamente que esses números são relacionados a um determinado ponto de acesso, mas podem se estender a maiores proporções com a adição de recursos de rede como roteadores e *switches*.

Para o caso da tecnologia *Zigbee*, este tipo de problema praticamente não existe pois uma rede pode suportar um número razoável de dispositivos quando comparado com outras tecnologias e no número necessário de dispositivos em aplicações em geral.

- **Taxa de Transmissão:** a tecnologia *Bluetooth* foi originalmente projetada para viabilizar pequenas transmissões, incluindo voz e posteriormente dados. Atualmente há vários dispositivos com essa tecnologia, cuja velocidade permite desde a comunicação entre um dispositivo de voz *Bluetooth* com um telefone celular até transferências de músicas, vídeos, etc. Portanto, é uma tecnologia que possui uma boa taxa de transmissão e se aplica em muitos casos onde o consumo e esta boa taxa de transmissão são desejáveis.

Analisando o *WiFi*, a taxa é bem maior que as outras duas tecnologias permitindo assim uma gama muito maior de aplicações, mas todas voltadas a equipamentos que possuam uma fonte de energia e não se preocupam tanto com o consumo. Normalmente é utilizada então em computadores, onde geralmente há uma rede elétrica próxima ou em *notebooks* com baterias de longa duração.

Neste projeto, a taxa de transmissão deve ser a mínima possível, tendo apenas o dado de consumo de cada nó, e que portanto 250KB é uma taxa muito mais do que necessária para o projeto.

- **Alcance Máximo:** esse item acaba sendo muito importante, pois dentro de uma casa há muitos obstáculos para que o sinal de transmissão se propague, isso inclui paredes, armários, etc. Dessa forma, o alcance para transmissão deve ser razoável que a comunicação seja viável mas também não deve ser muito grande pois afetaria diretamente no consumo do dispositivo.

A tecnologia *Bluetooth* possui um alcance máximo relativamente baixo, 10 metros sem obstáculos pode se tornar muito baixo se uma parede estiver no meio da comunicação. Tornaria inviável a utilização desta tecnologia para este projeto.

Por outro lado, *WiFi* e *ZigBee* possuem alcances similares em torno de 100 metros, isso proporciona uma boa comunicação mesmo que obstáculos como paredes estejam no caminho dos sinais transmitidos. Como essas duas tecnologias também possuem mecanismos de possível adição de nós para apenas aumentar o alcance (como roteadores), neste quesito elas se encaixam na implementação deste projeto.

- **Latência para Enumeração:** toda vez que um determinado dispositivo se junta a uma determinada rede, há um tempo para que este dispositivo seja associado e passe a se comunicar plenamente com os outros dispositivos da rede. Dessa forma, quanto mais rápido um dispositivo se junta a rede, mais rapidamente seus dados estarão no repositório. Problemas quanto a

mobilidade também são minimizados quando a velocidade de associação é rápida. No caso de um sistema *ZigBee*, a associação de um nó é muito rápida, sendo da ordem de mili segundos, enquanto outras tecnologias chegam a dezenas de segundos.

Além de todos os pontos comentados, há uma análise de da performance do protocolo ZigBee 802.15.4 sendo comparado com outros protocolos. Nesta análise, vemos a relação sinal/ruído SNR versus a taxa de erro (bit error rate - BER). O gráfico na Fig. 2.11 basicamente ilustra como os padrões se comportam em ambientes cujo nível de sinal ruído é baixo.

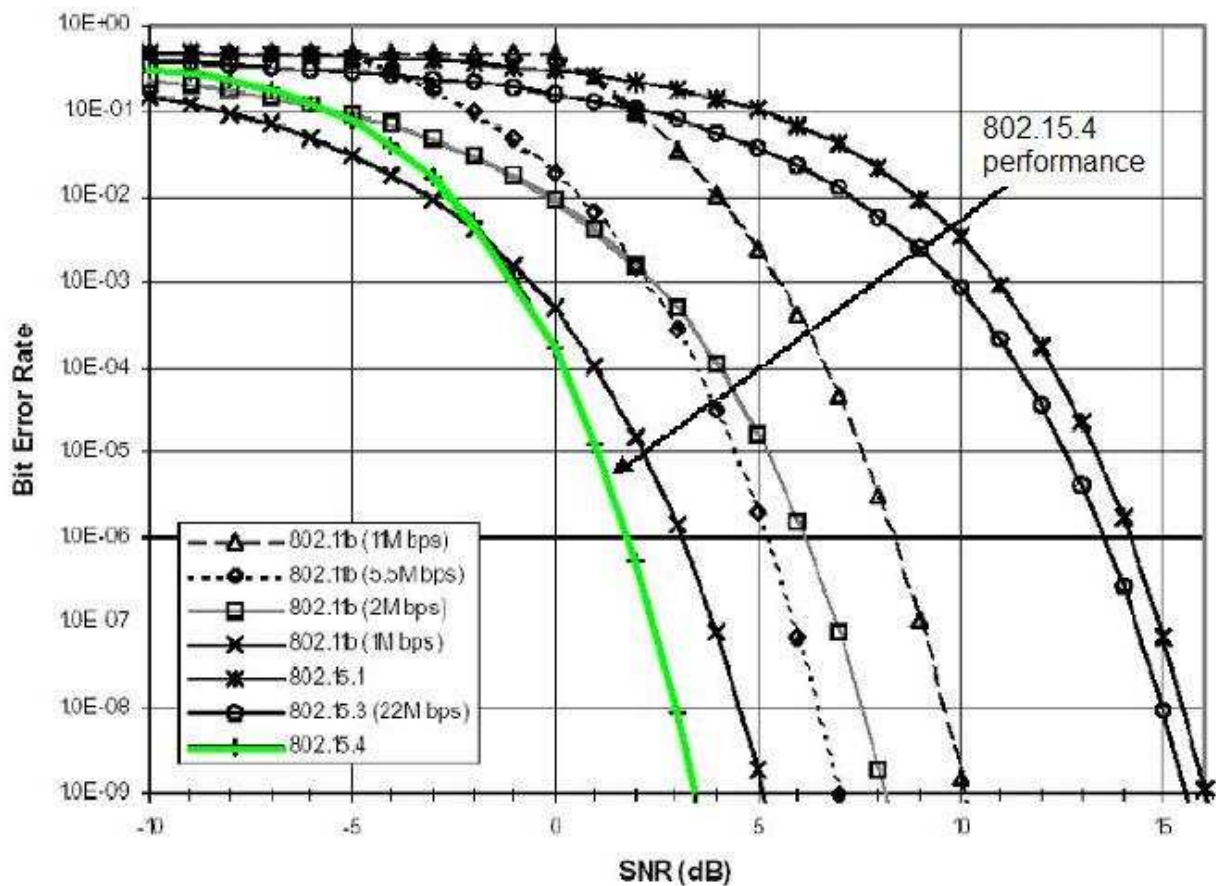


Fig. 2.11: Comparação da BER e Relação Sinal Ruído entre Tecnologias [5]

Analisando o gráfico da Fig. 2.11, em todos os pontos acima da relação sinal ruído de -2dB o padrão 802.15.4 é melhor que qualquer outro padrão pois o nível da taxa de bit de erro é o menor de todos. O padrão ZigBee apenas é pior em SNR menores que -2dB, o que é um ambiente com SNR extremamente baixo.

Segundo esta comparação, podemos verificar o porquê da escolha de implementação da rede sem fio utilizando a tecnologia *ZigBee*.

2.7 O Modelo ISO/OSI

O chamado modelo *OSI* (*Open Systems Interconnection*) pode ser definido como uma referência para a interconexão de sistemas abertos, isso significa que *software*, *hardware* ou tecnologias podem ser diferentes, mas as trocas de informações é possível pois as interfaces e protocolos são comuns e definidos.

Em resumo, é um padrão criado que possibilita diferentes fabricantes e tecnologias usarem em conjunto um ambiente heterogêneo de forma transparente.

O modelo *OSI* foi dividido em 7 camadas, onde as tarefas de cada camada são definidas bem como as interfaces de comunicação entre camadas. Obviamente que nem todo projeto ou sistema necessita de todas as camadas. Exemplificando, a rede *ZigBee* possui um número menor de camadas para melhorar tanto a performance quanto simplificar a arquitetura de rede.

As camadas OSI são ilustradas na Fig. 2.12



Fig. 2.12: Camadas do Modelo OSI

Descrevendo brevemente cada uma das camadas

- **Camada Física:** nesta camada é onde se encontram os meios físicos da comunicação do sistema, seja o cabeamento ou sem fio. Suas principais atribuições são:
 - Fazer com que os bits da comunicação seja movido pelo meio de transmissão;
 - É onde se define as principais características como taxa de dados, tensões, mecânica, etc.;
 - Responsável pelo controle de acesso ao meio;
 - Responsável pelo controle da velocidade de transmissão.
- **Camada de Enlace:** esta camada é definida como sendo a camada responsável pelo link de dados, e que portanto tem por objetivo adicionar um protocolo ao meio físico fazendo com que a comunicação seja razoavelmente confiável. Suas principais atribuições são:
 - Nesta camada, há a detecção de erros e opcionalmente já pode-se corrigí-los;
 - Responsável pela transmissão e recepção de quadros/dados;
 - Responsável pelo controle de fluxo de dados;

- Estabelece um determinado protocolo para a comunicação entre sistemas como o PPP, LAPB, etc.
- **Camada de Rede:** é responsável pelo endereçamento dos pacotes da rede, dessa forma é a camada responsável por converter endereços lógicos em endereços físicos, fazendo com que os pacotes transmitidos possam chegar ao seu destino. Suas atribuições são:
 - Responsável por determinar a rota dos pacotes para chegar ao destino;
 - Possui dados como prioridade e condições de tráfego de pacotes para tomar ações como reconfigurar uma rota;
 - Faz a conversão de endereços lógicos para endereços físicos;
 - Faz o tratamento de erros da rede;
 - Responsável pela fragmentação de pacotes;
 - Faz o controle de congestionamento da rede.
- **Camada de Transporte:** basicamente, esta camada recebe os dados provenientes da camada superior (Sessão) e os divide em pacotes a serem transmitidos pela camada inferior (Rede). Nesta camada é onde ocorre a separação das camadas de aplicação e camadas de nível físico. Há duas definições de modos de operação para o protocolo de transporte:

- Orientado a Conexão (Exemplo TCP).
- Não-Orientado a Conexão (Exemplo UDP).

A principal diferença entre os dois é de que o Não-Orientado a Conexão não garante a entrega da informação ao seu destino. Mas pode ser usada sem maiores problemas quando as camadas inferiores sejam confiáveis, como em redes locais.

- **Camada de Sessão:** a camada de Sessão faz com que dois dispositivos diferentes consigam estabelecer uma sessão de comunicação. Suas principais atribuições são:
 - Disponibilização de serviços para eventuais problemas de pane na rede.
 - Responsável pela abertura de portas onde aplicações diferentes possam utilizar da melhor forma os recursos de rede.
- **Camada de Apresentação:** esta camada é responsável basicamente pela tradução dos dados recebidos da camada superior (Aplicação) para um formato comum que será utilizado no protocolo de transmissão. Outra responsabilidade dessa camada é a compressão dos dados a fim de tornar a transmissão mais eficiente.
- **Camada de Aplicação:** a camada de Aplicação tem a função de possuir aplicativos padrão para comunicação que outros aplicativos podem usar. Seria uma padronização de tipos de aplicativos possíveis a utilizarem recursos de rede.

Pode-se citar como exemplo um FTP, que pode ser utilizado por outros aplicativos para a transmissão de dados.

Pode-se verificar que apesar das definições de cada camada, 7 camadas é um nível muito grande de redundância e até desperdício de recursos. Tomando como base uma rede TCP/IP que é a base da Internet, o número de camadas é menor, possuindo fusões de camadas para chegar ao melhor custo/benefício tanto em nível de *hardware* como em *software*.

Capítulo 3

A Rede ZigBee

3.1 Introdução

O padrão *ZigBee* é um conjunto de especificações que foi desenvolvido pela *ZigBee Alliance* [5] com a finalidade de criação de um padrão para redes com grande número de pontos, baixa taxa de transmissão de dados, protocolo aberto e, principalmente, de baixo consumo e baixo custo.

A Aliança *ZigBee* é uma organização criada com a missão de desenvolver produtos com a plataforma *ZigBee* a nível global, sendo então responsável também por controlar e monitorar os produtos lançados no mercado. Ela também tem o dever de certificar e verificar a conformidade de produtos *ZigBee*.

Atualmente, empresas como Ember, Freescale, Honeywell, Invensys, Mitsubishi, Motorola, Philips, Samsung e mais de 120 empresas fazem parte da Aliança *ZigBee*.

Dessa forma, a tecnologia *ZigBee* veio não apenas para atender um setor específico de mercado, mas sim para aplicações industriais, comerciais, residenciais e qualquer tipo de aplicação que envolva controle, monitoração e automação. Veja na Fig. 3.1.



Fig. 3.1: Aplicações utilizando o modelo ZigBee

Uma rede sem fio nunca foi bem aceita no setor industrial, dado o alto nível de ruído e interferências causadas por grandes motores e circuitos de potência, o que acaba por deixar as redes sem fio inviáveis em ambientes industriais. O *ZigBee* foi desenvolvido para suportar estes tipos de problemas e então acabar com o paradigma de redes sem fio na indústria. Portanto a tecnologia *ZigBee* não aplica-se somente a um ramo de aplicações domésticas ou simples usuários, mas também pode suportar grandes redes em uma indústria.

Em termos de história, o padrão IEEE 802.15.4 foi finalizado em maio de 2003, logo depois, a primeira especificação *ZigBee* 1.0 foi ratificada em Dezembro de 2004. Desde então há várias versões do padrão sendo a mais atual de 2007.

Durante este capítulo, será dada uma abordagem sobre o funcionamento de uma rede *ZigBee*, suas características, arquitetura, etc.

3.2 Arquitetura

A arquitetura *ZigBee* também é definida em camadas como o modelo *OSI* na Sessão 2.7. Dessa forma, cada uma das camadas possui atribuições e responsabilidades e as interfaces entre as camadas são baseadas em primitivas sendo passadas como mensagens entre as camadas.

Sendo a arquitetura *ZigBee* derivada da estrutura de referência do modelo *OSI* e tendo em vista que algumas camadas acabam por aumentar a complexidade da rede, consumo de transmissão e vários outros pontos que vão contra a proposta de uma rede eficiente de sensores, algumas das camadas de rede foram suprimidas e resultaram nas camadas ilustradas na Fig. 3.2.



Fig. 3.2: Camadas do Protocolo ZigBee

A Fig. 3.2 ilustra as camadas com uma diferenciação entre as duas camadas superiores e as duas inferiores. As definições são:

- A Camada Física e a Camada de Enlace que são as camadas inferiores na pilha são definidas pelo padrão IEEE 802.15.4.
- A Camada de Rede e Camada de Aplicação que são as camadas superiores na pilha são definidas pela aliança *ZigBee*.

3.2.1 A Camada Física

A Camada Física é a camada mais baixa de todo o sistema, é onde está compreendido o meio físico da transmissão de dados através do *RF* (Rádio Frequência). Ela também é responsável pela

comunicação com a camada superior *MAC*.

As principais atribuições da Camada Física são [7]:

- Ativação e Desativação do transceiver;
- *ED (Energy Detection)* ou Detecção de Energia - A medida da energia recebida é utilizada principalmente no algoritmo de seleção do canal a ser utilizado, sendo assim uma estimativa da potência do sinal recebido dentro do canal selecionado;
- *LQI (Link Quality Indication)* ou Indicação da Qualidade do Enlace - Todo pacote possui campos de qualidade do enlace, que é a caracterização da energia e qualidade do pacote recebido. Juntamente com o *ED*, uma estimativa da relação sinal ruído pode ser feita melhorando o resultado do *LQI* que é enviado as camadas superiores.
- *CCA (Clear Channel Assessment)* que é um mecanismo de verificação dos pacotes recebidos para saber se há algum tipo de sinal baixo. Isso implica diretamente no algoritmo implementado para evitar colisão de pacotes o chamado *CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance)*.
- Seleção do Canal de Transmissão;
- Transmissão e Recepção dos pacotes de dados.

Faixa de Operação

A faixa de operação definida para o padrão IEEE 802.15.4 é a *ISM (Industrial-Scientific Medical)*. Esta faixa é destinada a diversas aplicações e não requer licença de uso e é ilustrada na Fig. 3.3.

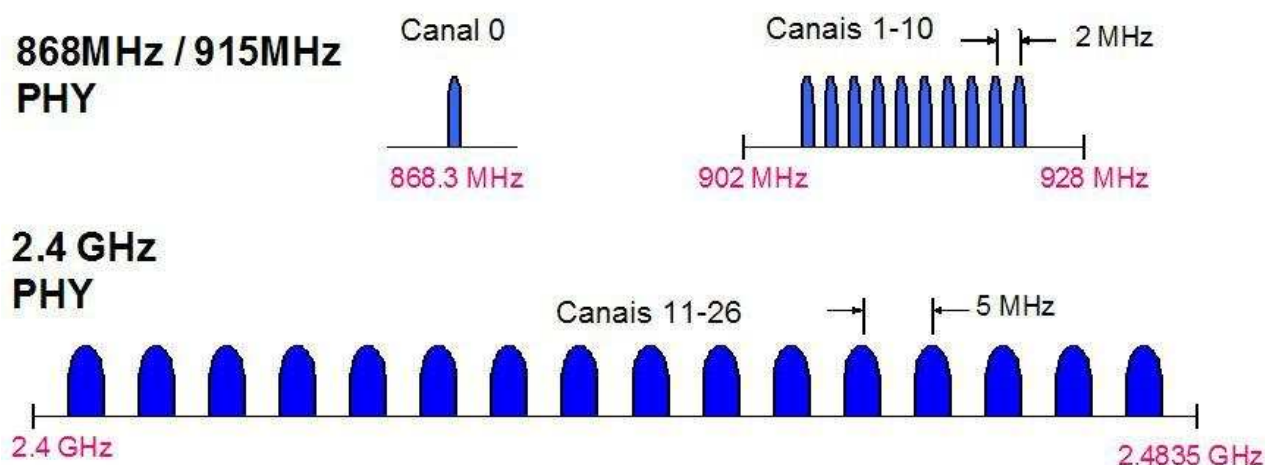


Fig. 3.3: Faixas de Frequência de Operação do Padrão IEEE 802.15.4

868MHz Faixa utilizada apenas na Europa e que possui uma taxa de transmissão de 20kbps, possui apenas um canal e sua modulação geralmente é a *BPSK (Binary Phase Shift Keying)*;

915MHz Faixa de operação de 902 a 928MHz disponível apenas na América do Norte e Austrália, sendo sua taxa de transmissão de 40kbps, 10 canais e geralmente a modulação *BPSK*;

2,4GHz Faixa *ISM* global cuja banda se estende de 2,4 a 2,4835GHz, taxa de transmissão de 250kbps, 16 canais e geralmente a modulação *O-PQSK* (*Offset-Quadrature Phase Shift Keying*)

Blocos da Camada Física

A camada Física possui basicamente três interfaces:

- PD-SAP
- PLME-SAP
- RF-SAP

As interfaces **PD-SAP** (*Physical Data - Service Access Point*) e **PLME-SAP** (*Physical PAN Information Base - Service Access Point*) são interfaces lógicas provendo dados e fazendo o gerenciamento de acesso aos serviços entre as camadas MAC e Física.

Já a interface **RF-SAP** (*Radio Frequency - Service Access Point*) é responsável pela comunicação com o meio físico (*hardware*) de *RF*.

Pode-se verificar a arquitetura da camada Física na Figura 3.4.

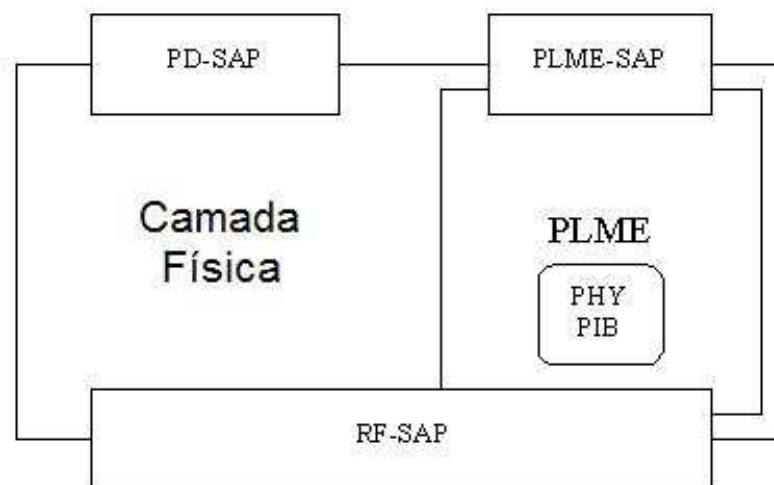


Fig. 3.4: Diagrama em Blocos da Camada Física [7].

O PLME é responsável pelo transporte de comandos de gerenciamento entre a entidade de gerenciamento da camada MAC e da camada Física e é onde se encontra o banco de dados da camada Física o chamado PHY PIB (*Physical PAN Information Base*).

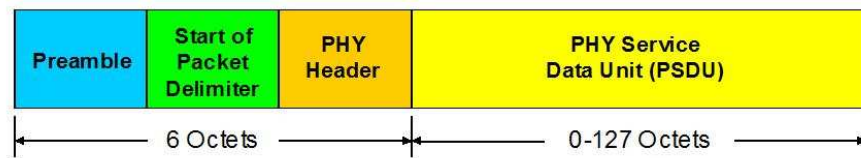


Fig. 3.5: Estrutura do Pacote da Camada Física [7].

A Estrutura do Pacote da Camada Física

A estrutura dos pacotes na camada Física é ilustrada na Fig. 3.5. Os campos são:

- *Preamble* (32 bits): Utilizado na sincronização;
- *Start of Packet Delimiter* (8 bits): Delimitador do começo do pacote;
- *PHY Header* (8 bits): Tamanho do PSDU;
- PSDU (0 to 1016 bits): Campo de dados.

3.2.2 A Camada MAC

A Camada MAC é a interface entre a camada superior (de Rede) e a inferior (Física), seu objetivo é de gerenciar e manipular o acesso ao canal físico de rádio. Suas principais atribuições são [7]:

- Gerar os *beacons* de rede se o dispositivo é do tipo coordenador;
- Sincronizar os *beacons* de rede;
- Suporte de associação e desassociação dos elementos de rede;
- Suporte da segurança dos dispositivos;
- Implemento do mecanismo de acesso aos canais de *RF* chamado de *CSMA-CA* (*Carrier Sense Multiple Access - Collision Avoidance*);
- Manipular e gerenciar o mecanismo de *GTS* (*Guaranteed Time Slot*);
- Prover e controlar o enlace entre duas entidades de dispositivos MAC.

Blocos da Camada MAC

As interfaces da camada MAC são:

- MCPS-SAP
- MLME-SAP

- PD-SAP
- PLME-SAP

A comunicação entre a camada superior, a de Rede, é através das entidades: *MCPS-SAP* (*MAC Common Part Layer - Service Access Point*) para serviços de gerenciamento e controle; e a *MLME-SAP* (*MAC Layer Management Entity - Service Access Point*) para serviços de transmissão de dados. A Fig. 3.6 ilustra o diagrama de blocos da Camada MAC.

Dessa forma, o *MLME* é responsável pelo gerenciamento de comandos, indicações, respostas e confirmações que são utilizadas no gerenciamento da *PAN*. Outra função do *MLME* é o banco de dados da camada MAC onde são encontradas as informações referentes aos dados de acesso da camada Física, estatísticas, etc. Esse banco é chamado de *MAC PIB* (*MAC PAN Information Base*).

O *MCPS* faz o tratamento dos comandos relacionados aos dados transmitidos [7].

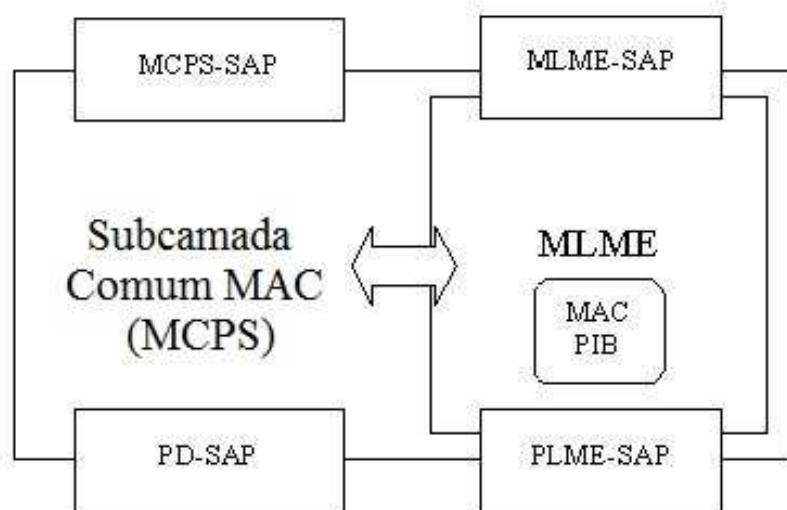


Fig. 3.6: Diagrama em Blocos da Camada MAC [7].

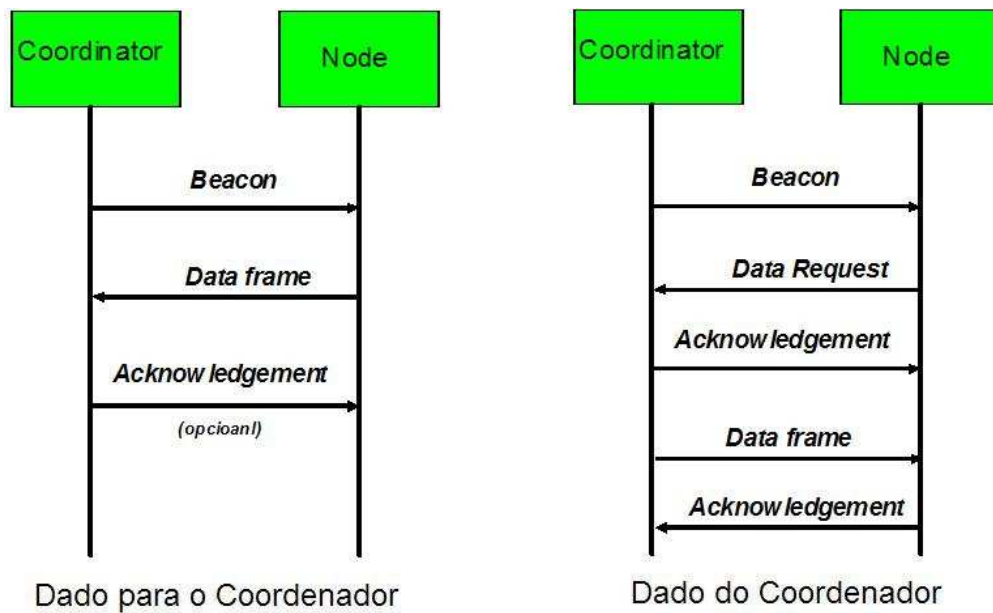
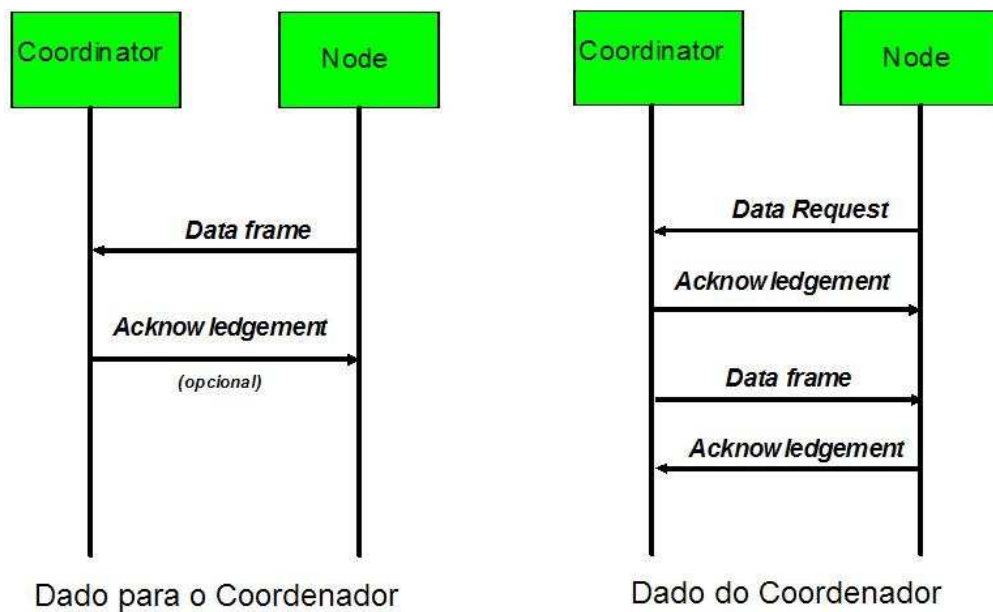
Rede Beacon e Não-Beacon

A configuração da rede do padrão 802.15.4 [7] pode ser de dois tipos, chamadas de *Beacon* e *Não-Beacon*. A Fig. 3.7 ilustra o funcionamento de uma rede *Beacon* e a Fig. 3.8 ilustra o funcionamento de uma rede *Não-Beacon*.

Na configuração *Não-Beacon*, depois dos dispositivos serem associados e reconhecidos pela rede, eles podem se comunicar um com o outro a qualquer momento sendo o acesso aos canais de comunicação e conexão gerenciados pelo mecanismo *CSMA-CA*.

Em uma rede com a configuração *Beacon* o coordenador fica transmitindo periodicamente um determinado pacote chamado *Beacon* que é utilizado pelos dispositivos de rede na sincronização do envio e recebimento de pacotes através dos canais de *RF*.

Pode-se resumir a rede *Não-Beacon* como sendo a transmissão de dados de qualquer dispositivo em qualquer momento desde que o canal de transmissão não esteja ocupado, já a rede *Beacon* a

Fig. 3.7: Troca de Mensagens numa Rede *Beacon* [7].Fig. 3.8: Troca de Mensagens numa Rede Não-*Beacon* [7].

transmissão acontece com a sincronização do coordenador, sendo então determinado o intervalo de tempo de cada dispositivo de rede.

O Superframe

O intervalo entre *Beacons* é chamado de *Superframe* que é dividido em duas partes, a *CAP* (*Contention Access Period*) e o *CFP* (*Contention Free Period*).

O *CAP* é designado para a transmissão de pacotes de dados de dispositivos que não possuem sincronização, como por exemplo alarmes ou controles onde há a concorrência de utilização do canal de rádio pois um ou mais dispositivos querem transmitir ao mesmo tempo. Neste caso, há a necessidade de utilização do mecanismo CSMA-CA 3.2.2.

O *CFP* é designado para a transmissão de dados com sincronização, dessa forma, dentro do intervalo *CFP* há fatias de tempo chamados de *GTS* que fazem o mapeamento dos dispositivos ou um grupo de dispositivos. Esta sincronização de pacotes é tratada pelo coordenador que tem a responsabilidade de gerenciar os pacotes *beacon*.

Um exemplo de um *Superframe* pode ser visto na Fig. 3.9, onde pode-se ver os sinais *beacon* no início e final de um *superframe*.

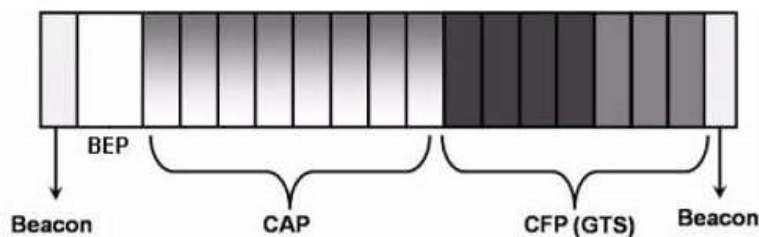


Fig. 3.9: Exemplo de um *Superframe* [7].

Há configurações de rede onde um *superframe* não contenha *CFP*, ou ainda redes que possuem um período de inatividade. Há uma grande vantagem em relação ao tempo de inatividade pois quanto maior for este tempo, maior será a economia da energia gasta pelos dispositivos de rede. Um exemplo de um *superframe* com inatividade pode ser visto na Fig. 3.10.

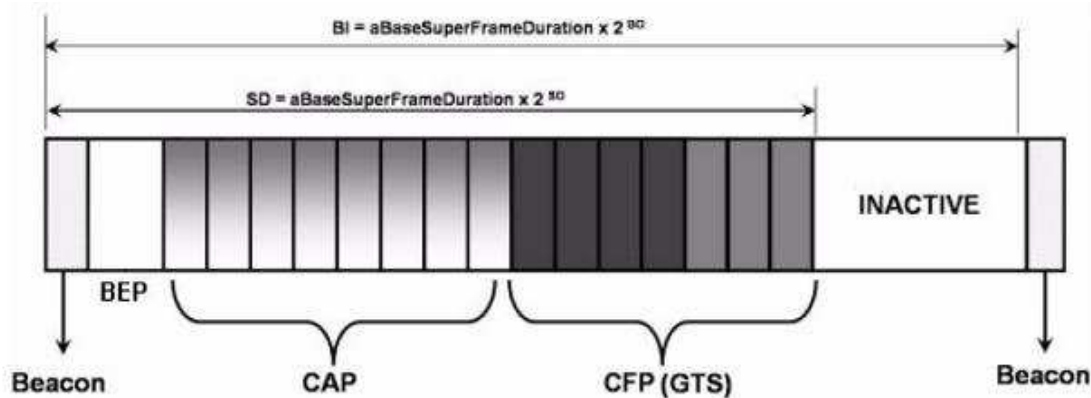


Fig. 3.10: *Superframe* com inatividade [7].

O algoritmo CSMA-CA

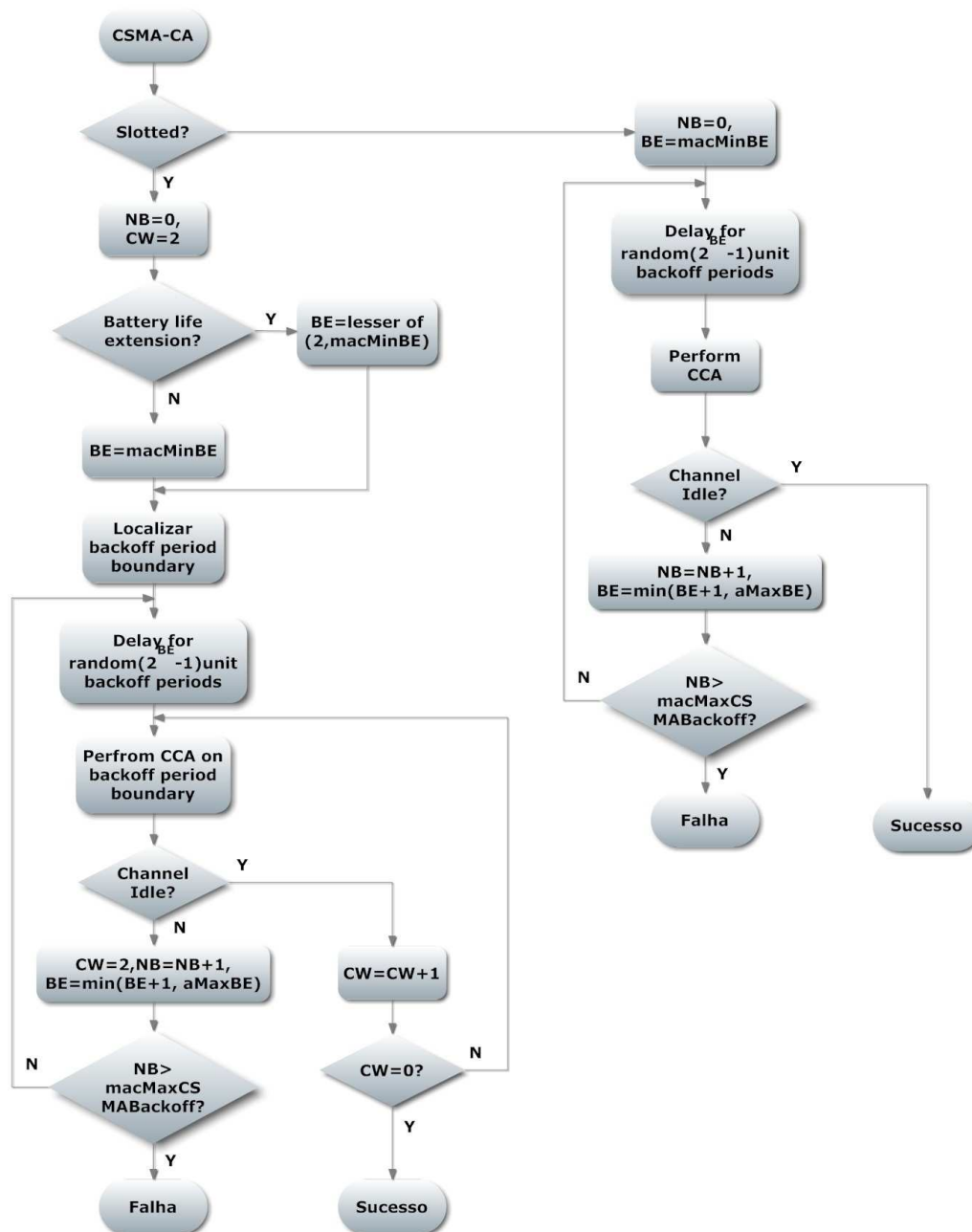


Fig. 3.11: Algoritmo CSMA-CA [7].

Uma rede onde se tem muitos pontos de rede não pode simplesmente enviar e receber dados sem nenhum tipo de controle relacionado à colisão dos pacotes. Portanto, há a necessidade de um mecanismo que resolva este tipo de problema, que é o chamado CSMA-CA.

Quando um *superframe* é usado em uma PAN, então o CSMA-CA *slotted* deve ser usado. Mas se *beacons* não estão sendo usados em uma PAN ou um *beacon* não pode ser localizado em uma rede

beacon, o algoritmo *CSMA-CA unslotted* é usado.

Este algoritmo é implementado usando unidades de tempo, que são chamados de *backoff periods*. Num canal onde o *CSMA-CA* está sendo usado, os períodos de tempo de cada dispositivo na *PAN* é alinhado com a fatia de tempo dos pacotes do coordenador da *PAN*. Assim, toda vez que um dispositivo que transmitir pacotes durante uma *CAP* (*Contention Access Period*) ele deve localizar o limite da próxima fatia de tempo. No caso de *CSMA-CA unslotted* as fatias de tempo de um dos dispositivos não precisam estar sincronizados com as fatias de tempo dos outros dispositivos.

Basicamente, cada dispositivo possui três variáveis: *NB*, *CW* e *BE*.

NB é o número de tempos que o algoritmo *CSMA-CA* foi usado durante a transmissão, sendo então inicializado com zero a todo início de transmissão.

CW é o tamanho da janela, definindo então o número de fatias de tempo que precisam estar inativos antes que a transmissão seja inicializada.

BE é o expoente das fatias, que é relacionado ao número de fatias que o dispositivo deve esperar antes de qualquer tentativa de acesso ao canal.

Essas três variáveis são utilizadas no algoritmo da Fig. 3.11 para a indicação de sucesso ou falha na utilização do canal.

3.2.3 A Camada de Rede

A Camada de Rede é a primeira a ser definida pela Aliança *ZigBee*. Basicamente seu objetivo é a conexão do início ao fim de um dispositivo à rede.

As suas principais atribuições são:

- Inicialização de uma rede: ser capaz de estabelecer uma nova rede;
- Associação e Desassociação: os nós de rede estão capacitados de se tornar membros de uma rede, bem como se desassociar da rede;
- Configuração: o nó de rede deve ser capaz de configurar sua pilha de operação de acordo com o tipo de rede;
- Endereçamento: o coordenador deve ser capaz de associar um endereço aos dispositivos que se tornaram membros da rede;
- Sincronização: os nós devem ser capazes de se sincronizar com outros nós por meio da recepção dos *beacons* ou *polling* de dados;
- Segurança: os pacotes devem chegar com integridade de quem enviou até o destino;
- Roteamento: os nós podem rotear pacotes para outros destinos.

Blocos da Camada de Rede

As interfaces da camada de Rede são:

- NLDE-SAP

- NLME-SAP
- MCPS-SAP
- MLME-SAP

Os serviços da camada de rede são providos através das entidades *NLDE* (*Network Layer Data Entity*) e a *NLME* (*Network Layer Management Entity*) que são ilustradas na Fig. 3.12.

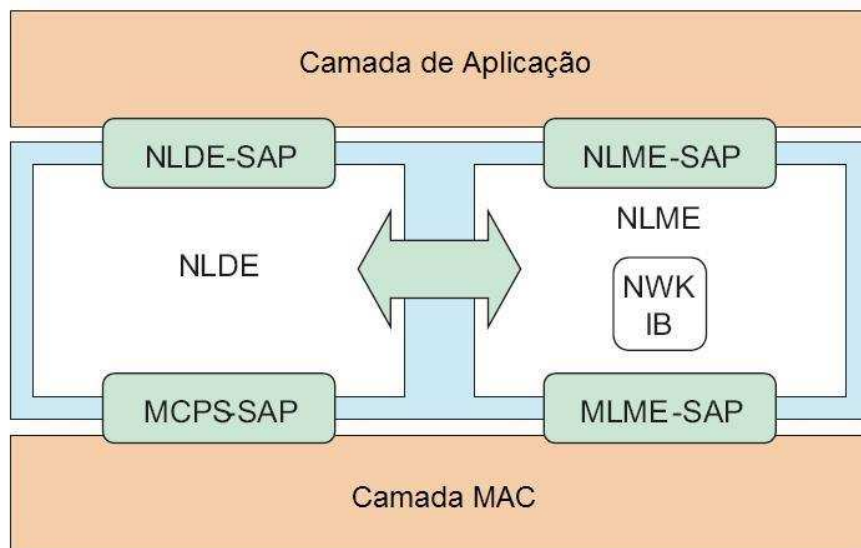


Fig. 3.12: Diagrama em Blocos da Camada de Rede [5].

O bloco *NLDE* é responsável pelo transporte de *APDU* (*Application Protocol Data Unit*) entre as entidades da camada superior de Aplicação e o *NLDE* via a primitiva de acesso *NLDE-SAP* (*Network Layer Data Entity - Service Access Point*). Suas principais atribuições são:

- Geração do nível de rede *NPDU* (*Network layer protocol data unit*): o *NLDE* deve ser capaz de gerar um *NPDU* de uma aplicação suportando as sub-camadas através da adição de um cabeçalho do protocolo apropriado.
- Roteamento específico da topologia: o *NLDE* deve ser capaz de transmitir um *NPDU* para os dispositivos apropriados, que é o destino final ou um passo para o destino final da comunicação.

O bloco *NLME* é utilizado para tratar os comandos de gerenciamento da camada de rede. Estes são enviados e recebidos através do *NLME-SAP* (*Network Layer Management Entity - Service Access Point*). Suas principais atribuições são:

- Configurar um Novo Dispositivo: deve ter a habilidade suficiente para configurar a pilha de operação requisitada, as opções de configuração e a entrada numa rede já existente;
- Iniciar uma rede: deve ser capaz de inicializar uma nova rede;

- Associação/Desassociação: o *NLME* deve ser capaz de se associar ou desassociar de uma rede;
- Endereçamento: o *NLME* como um dispositivo coordenador e roteador deve ter a capacidade de designar endereços aos dispositivos que se associaram a rede;
- Procura de Dispositivos Vizinhos: deve ser capaz de procurar, guardar e reportar as informações sobre os dispositivos vizinhos;
- Procura de Roteamento: esta sub-camada deve ser capaz de descobrir caminhos através da rede para que as mensagens sejam enviadas eficientemente aos seus destinos;
- Roteamento: o *NLME* deve ser capaz de usar diferentes mecanismos de roteamento para a mais eficiente transmissão de dados na rede.

A Fig. 3.12 ilustra uma conexão entre os blocos *NLDE* e *NLME*, isso se deve a necessidade de utilização das informações principalmente no banco de dados da camada de rede o *NWK IB* (*Network Information Base*) onde se encontram as informações da camada de Rede cuja responsabilidade de manter é do *NLME*.

3.2.4 A Camada de Aplicação

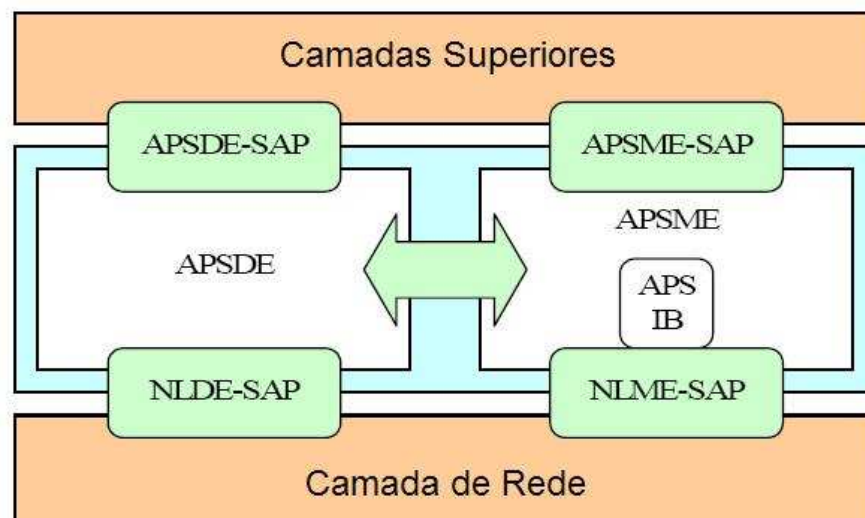


Fig. 3.13: Diagrama em Blocos da Camada de Aplicação [5].

A camada de Aplicação é o provedor de serviços entre a camada de rede e os objetos de aplicações. Nela temos as seguintes interfaces:

- *APSDE-SAP*
- *APSME-SAP*
- *NLDE-SAP*

- *NLME-SAP*

Elas são ilustradas na Fig. 3.13.

A camada de Aplicação na verdade é um conjunto de outras 3 subcamadas:

- **APS:** *Application Support Sub-Layer*
- **ZDO:** *ZigBee Device Object*
- **AF:** *Application Framework*

Elas são ilustradas na Fig. 3.14. Nela podemos verificar o funcionamento de toda a pilha *ZigBee* passando por todas as camadas e interfaces, verificar os blocos pertencentes a Aliança *ZigBee*, ao IEEE 802.15.4, etc.

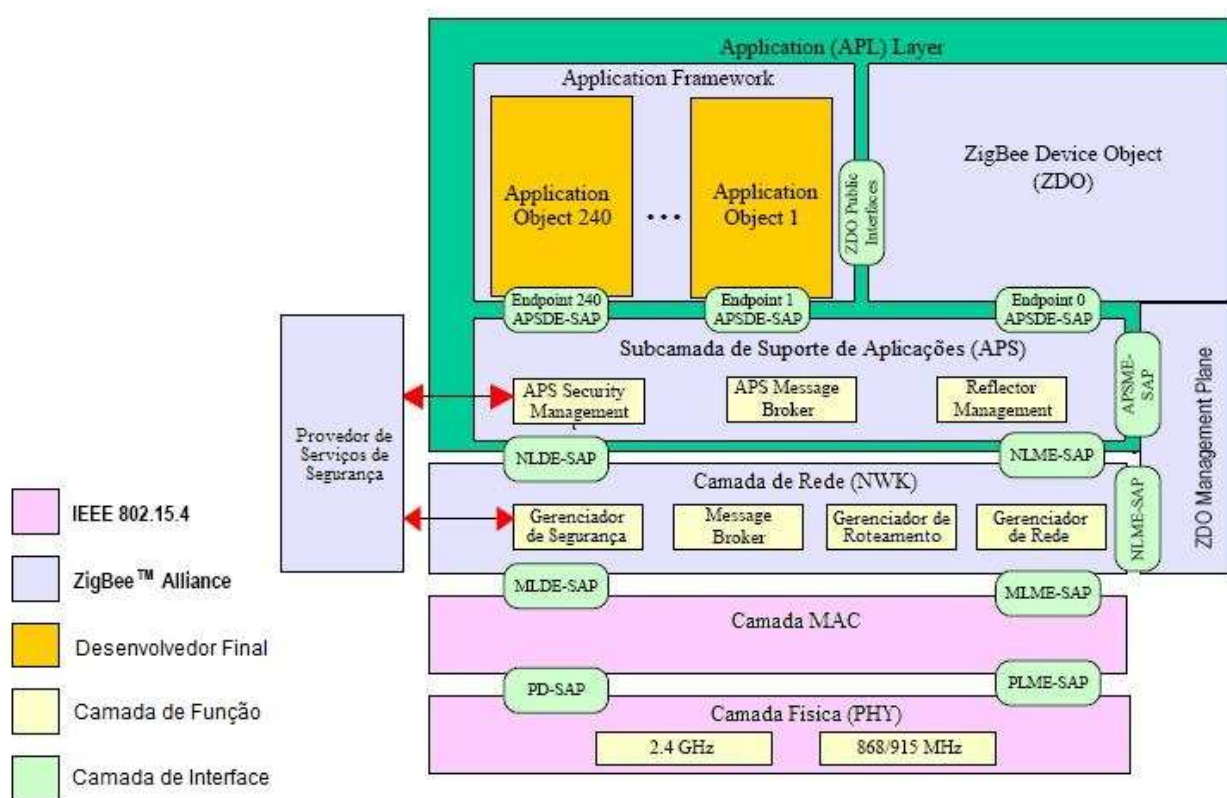


Fig. 3.14: Diagrama em Blocos da Camada do Sistema [5].

A Sub-Camada APS

A subcamada *APS* (*Application Support Sub-layer*) é responsável pela interface entre a camada de Rede e a camada de Aplicação através de serviços gerais que são usados pelo *ZDO* e são definidos pelos fabricantes. Estes serviços são divididos em duas entidades:

- *APSDE* (*Application Support Sub-layer Data Entity*)
- *APSME* (*Application Support Sub-layer Management Entity*)

A subcamada *APS* tem como principais atividades manter as tabelas de *binding* para garantir a conexão entre dois dispositivos que se baseiam em seus serviços, bem como enviando e recebendo mensagens entre um ou mais dispositivos para o gerenciamento da rede em que estes dispositivos estão localizados através do *APSME* e *APSDE*.

A transmissão dos *PDU*s é feita através do *APSDE* entre os dispositivos de rede de uma mesma rede, suportando fragmentação e mensagens com pacotes maiores dos suportados pelo *APSDU* (*Service Data Unit*) e também possui a responsabilidade da confiabilidade do transporte dos dados.

Já o *APSME* possui como principais atividades a segurança, a ligação de dispositivos, o estabelecimento de endereço de grupos e a manutenção do bando de dados dos objetos gerenciados, o chamado *AIB* (*APS Information Base*), que faz o mapeamento dos endereços entre os endereços *IEEE* de 64 bits para os endereços de rede *NWK* de 16 bits.

A Sub-Camada *ZDO*

O *ZDO* é basicamente um objeto de aplicação especial que faz a interação com as camadas inferiores da pilha *ZigBee* utilizando o *APSDE* para dados e *APSME* para comandos de gerenciamento. Esses comandos são:

- Inicialização da camada *APS*, da camada de Rede e da camada de segurança (*SSP*);
- Define a função principal do ponto de rede, sendo um coordenador, roteador ou *end device*;
- Responsabilidade pelas mensagens de *binding*;
- Estabelecimento de uma conexão segura na rede;
- Procurar e descobrir dispositivos na rede e determinar os serviços que estes dispositivos oferecem;
- Gerenciamento geral sobre itens de segurança, descoberta de outros dispositivos, da rede e conexão.

A Sub-Camada *Framework de Aplicação*

A sub-camada *Framework de Aplicação* é o ambiente no qual os objetos de aplicação são executados nos dispositivos *ZigBee*.

Pode-se ter até 240 objetos distintamente definidos onde cada um recebe um endereço de 1 a 240.

O endereço 0 é reservado para a interface de dados para o *ZDO* e o 255 é reservado para a interface de dados *broadcast* para todos os objetos de aplicação. Já os endereços 241 a 254 são reservados para um futuro uso.

A Sub-Camada de Segurança SSP

O *SSP* (*Security Service Provider*) é a camada responsável por fornecer os serviços de segurança, fazendo o estabelecimento troca das chaves de segurança entre as camadas de rede.

Ele é responsável por prover estes serviços as camadas de Rede e Aplicação. O *SSP* pode ser visto na Fig. 3.14.

3.3 Os Elementos de Rede

Os elementos de uma rede *ZigBee* são basicamente divididos em duas categorias:

- **FFD** (*Full-Function Device*)
- **RFD** (*Reduced-Function Device*)

O *FFD* é um dispositivo que pode operar em três modos em uma *PAN*: coordenador, roteador ou *end device*. Sua comunicação pode ser realizada com qualquer *RFD* ou outros *FFDs*. Como possui todas as funcionalidades da rede, este tipo de dispositivo também possui uma complexidade maior, sendo geralmente localizado em um local onde possua fonte de alimentação, pois seu consumo de energia acaba sendo maior. Dessa forma, este tipo de dispositivo acaba sendo indicado geralmente para ser um coordenador ou roteador.

Já o *RFD* são dispositivos com funções reduzidas, dessa forma sua comunicação pode ser feita somente através de dispositivos *FFD* e, obviamente, só podem se associar a uma rede através dos *FFDs*. Este tipo de dispositivo é voltado para os pontos finais de rede, ou seja, os elementos que farão o sensoriamento ou monitoramento dos sinais ou aplicações implementadas, dessa forma são tipicamente ligados à baterias e ficam grandes períodos de tempo inativos para conseguir a maior economia de energia possível, ficando assim o controle da rede para os dispositivos *FFD*.

Os dispositivos da rede *ZigBee* podem ser designados para as seguintes funções:

- **Coordenador:** é a raiz de uma rede *ZigBee* e pode ser uma bridge para outras redes. Cada rede possui necessariamente ao menos um coordenador e pode ser usado como repositório de informações da rede incluindo ações e segurança. São dispositivos *FFD* e tem função principal o gerenciamento global da rede.
- **Roteador:** são dispositivos que fazem o papel de extensores da rede, e podem ser tanto um *FFD* quanto um *RFD*. Dessa forma podem fazer o papel de um *End Device* fazendo medidas, monitoramento e atuação, bem como o controle de vários pontos de rede.
- **End Device:** como característica fundamental ele apenas fala com seu nó pai que pode ser um coordenador ou um roteador, não interferindo na conversa de outros dispositivos. Por ser um dispositivo *RFD*, é o que menos precisa de memória sendo o mais barato de todos e é destinado ao monitoramento ou atuação como função principal.

3.4 Topologias de Rede

O padrão de rede *ZigBee* pode operar em várias topologias: *Peer To Peer*, Estrela, Árvore e *Mesh*, que podem ser vistas na Sessão 2.3.

A topologia *Peer to Peer* é a mais simples, onde se tem um coordenador e um *end device*. Não é muito usual pois uma rede *ZigBee* normalmente possui inúmeros pontos ao invés de apenas um.

Analizando a configuração Estrela, pode-se ver que ela é composta por um elemento coordenador, sendo o responsável pelo gerenciamento de toda a rede e dos *End Devices*. Dessa forma, os *end device* podem enviar mensagens ao coordenador, mas se quiserem enviar uma mensagem para qualquer outro dispositivo da rede, esta necessariamente deve passar pelo coordenador antes de chegar ao seu destino final.

A topologia estrela é um exemplo de rede *single-hop* onde a transmissão de dados de cada ponto possui apenas um caminho.

A Configuração árvore é uma mistura da topologia estrela com roteadores, onde os *end devices* podem se associar a rede através do coordenador e roteadores tornando possível então a conexão de *end devices* fora da área de cobertura do coordenador. Dessa forma a transmissão de dados segue uma hierarquia de roteamento.

Outra topologia muito comum é a *mesh*, cuja principal característica é a possibilidade de *end devices* se comunicarem diretamente entre si, sem a necessidade das mensagens passarem pelo coordenador ou por roteadores. Para que isso ocorra, os *end devices* precisam ser configurados como *FFD*. Isso é um problema pois implica no aumento do consumo de energia e deve ser utilizado com o uso de alimentação diretamente na rede elétrica ao invés de baterias.

3.5 Inicialização dos Dispositivos

O procedimento de inicialização dos dispositivos segue aos padrões descritos pelos órgãos padronizadores IEEE 802.15.4 [7] e da Aliança *ZigBee* [5].

3.5.1 Inicialização de uma nova PAN

1. **Inicialização das camadas Física e MAC:** primeiramente deve-se inicializar as camadas Física e MAC. Dessa forma os serviços *MCPS* e *MLME* se tornam acessíveis sendo possível então ao primeiro elemento da rede, o coordenador, ser inicializado.

Nesta etapa, não apenas se faz a inicialização das duas camadas mas também se faz o *Reset* de todos os módulos associados ao microcontrolador;

2. **Varredura de detecção de Energia:** (*Energy Detection Scan*), logo após a inicialização das camadas Física e MAC, o coordenador da PAN precisa obter a frequência de rádio que será utilizada como canal lógico da PAN. Dessa forma, uma varredura de canais acontece para a escolha do canal com menor nível de energia e que não esteja sendo utilizado por outra unidade de rede;
3. **Escolha de Endereço Curto:** Todo dispositivo possui um endereço pré-fixado chamado de Endereço Longo constituído por 8 bytes. Porém, todo elemento de rede recebe um Endereço

Curto constituído por 2 *bytes*. Como o coordenador é o primeiro dispositivo a ser inicializado, este deve escolher seu próprio endereço curto podendo ser qualquer valor exceto os reservados;

4. **Escolha do PAN ID:** O *PAN ID* é um identificador utilizado na transmissão dos pacotes de rede, ele indica o número do conjunto de elementos de uma rede. Dessa forma, pode-se ter vários grupos de redes diferentes em um mesmo local com *PAN IDs* diferentes, fazendo com que o canal lógico seja compartilhado entre as redes.
5. **Iniciando uma PAN:** Depois de toda a inicialização da pilha *ZigBee*, é necessária a inicialização da *PAN* através do *MLME* para os valores desejados de configuração da *PAN*, como as configurações de *beacon*, etc.

3.5.2 Principais iterações de uma PAN

Depois de inicializada, a *PAN* possui várias possíveis iterações, que podem ser descritas abaixo:

- **Associando um elemento a uma PAN:** Após a criação da *PAN*, os dispositivos distribuídos devem se associar ao coordenador de rede. Isso é feito através da Varredura Ativa (*Active Scan*) onde o dispositivo procura por um coordenador que aceite a sua requisição de entrada na rede.
- **Transferência de Dados:** Depois da Associação do elemento à rede, informações de medidas, etc. podem ser enviadas e recebidas pelos elementos associados cuja transmissão pode seguir os modos *beacon* e não *beacon* como descrito na Sessão 3.2.2

3.6 Modulação do Sistema ZigBee

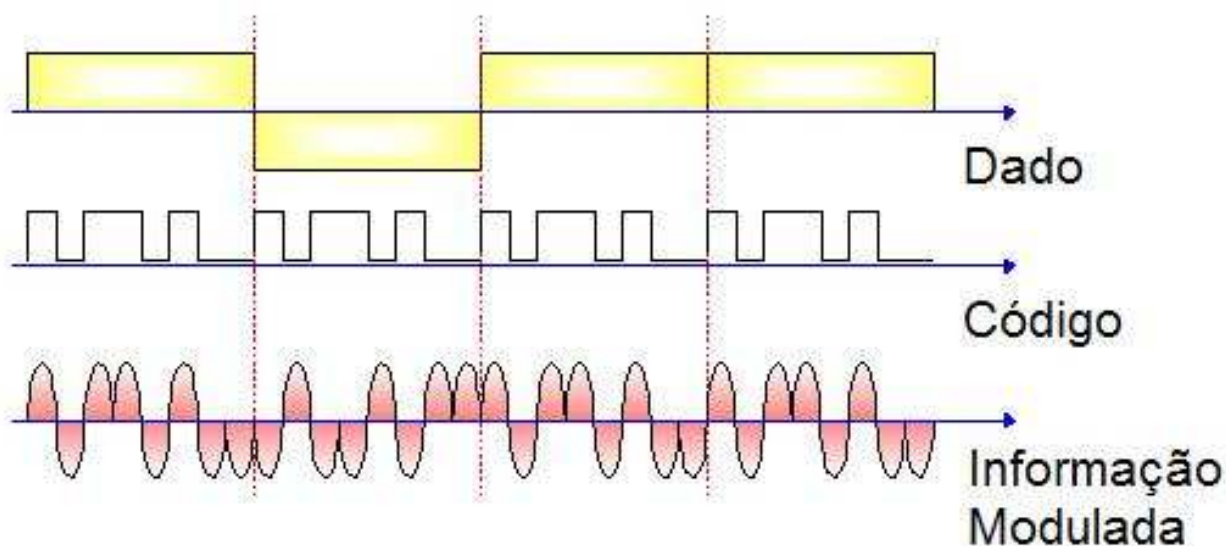


Fig. 3.15: Modulação DSSS

O sistema de transmissão *ZigBee* utiliza como modulação o *DSSS (Direct Sequence Spread Spectrum)*. Este tipo de modulação basea-se no envio de sinal codificado por uma sequência pseudo-randomica chamada de *chip sequence*.

Dessa forma, o sinal codificador é combinado com um circuito digital "OU Exclusivo". Este sinal então faz a alteração da fase da portadora do sinal dependendo da saída da combinação que pode ser visto na Fig. 3.15

A principal vantagem deste tipo de modulação, é de que vários transmissores podem utilizar o mesmo canal, apenas é necessária a mudança do código a ser combinado com a informação.

Capítulo 4

Desenvolvimento do Projeto

4.1 Introdução

O projeto pode ser desmembrado em duas partes: *hardware* e *software*. Ambas as partes foram desenvolvidas buscando atender os requisitos do projeto com a melhor relação custo-benefício. Não apenas o valor teórico das medidas foi levado em conta, mas também como o projeto pode ser empregado na prática em uma residência.

Dessa forma, este capítulo descreve como foi o desenvolvimento de todas as partes do projeto e como foram feitas as escolhas dos componentes utilizados e as configurações de *hardware*.

4.2 Hardware

O desenvolvimento do *hardware* foi baseado no *kit* de desenvolvimento ATAVRRZRAVEN 2.4 GHz Evaluation and Starter Kit [9] da empresa Atmel [10]. A Fig. 4.1 ilustra os componentes do *kit*.



Fig. 4.1: *Kit* de desenvolvimento ATAVRRZRAVEN [9]

As principais características deste *kit* são:

- 2 placas *LCD* com interfaces digitais e analógicas contendo:
 - *Transceiver* AT86RF230 [11];
 - 2 microcontroladores ATmega3290P [12] e ATmega1284P [13];
 - *Speaker* e microfone;
 - Interface de entrada e saída com *GPIOs* digitais e analógicos, *USART* e *relay driver*;
 - Alimentação via 2 baterias de 1,5V ou fonte externa (5 a 12V);
 - Comunicação *ISP* e *JTAG*.
- 1 Placa *USB*
 - *Transceiver* AT86RF230 [11];
 - Microcontrolador com interface *USB Full Speed* AT90USB1287 [14];
 - Comunicação *JTAG*.

Cada uma das placas é utilizada para um determinado objetivo, sendo uma para a rede de sensores e outro para a coleta de dados da central:

- A placa *LCD* será utilizada em cada ponto da rede, servindo então como interface de rede e controle de cada um dos medidores, sendo responsável pela comunicação com os sensores que farão a medida do consumo, pela retenção desta medida e pelo envio das medidas registradas para a central.

A central ou coordenador é responsável por armazenar todas as medidas que chegam dos sensores, sendo então responsável pela comunicação e gerenciamento da rede e pelo armazenamento das medidas de cada ponto.

Desta forma, o *kit* possui muito mais recursos que o necessário, pois apenas um microcontrolador pode fazer a interface com os sensores e com o *transceiver* de comunicação *ZigBee*. O diagrama em blocos pode ser visto na Figura 4.2.

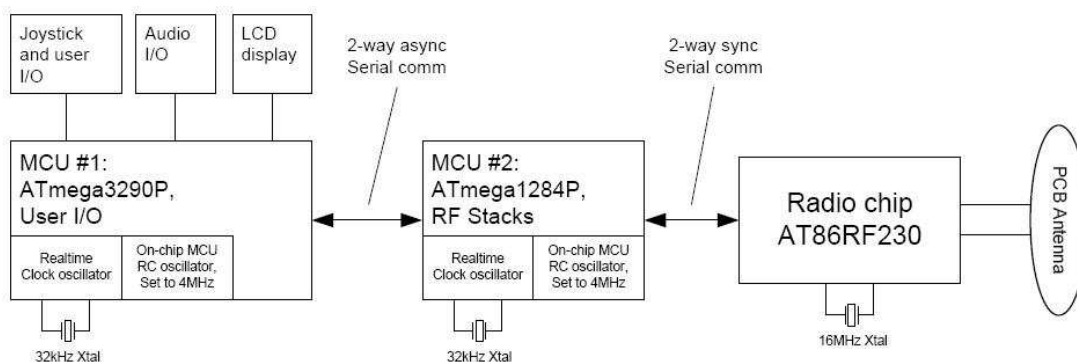


Fig. 4.2: Blocos do *Kit* de desenvolvimento ATAVRRZRAVEN [9]

Portanto, todos os periféricos como *LCD*, *speaker*, microfone, o próprio microcontrolador ATmega3290P, etc. serão desligados a fim de diminuir ao máximo o consumo de cada ponto de rede e fazer com que o projeto se torne mais simples. O diagrama em blocos final pode ser visto na Fig. 4.3.

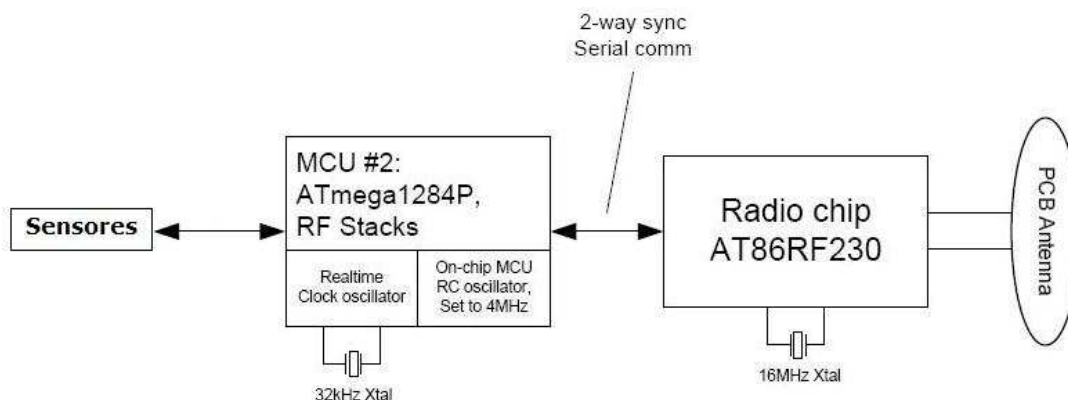


Fig. 4.3: Blocos do Projeto

O bloco "Sensores" é dividido em três tipos de placas, uma para cada tipo de medida:

- Medidas em tomadas e aparelhos eletrônicos;
- Medidas em Lâmpadas;
- Medidas no Chuveiro.

Essas divisões foram efetuadas para que seja possível a medida do consumo em cada ponto de consumo, inicialmente a idéia era utilizar o componente AD71056 [6] mas nem todos os pontos de consumo possui uma instalação simples, dessa forma outras soluções foram buscadas.

- A placa *USB* é utilizada para a coleta de dados das medidas através da própria rede. O diagrama em blocos é ilustrado na Fig. 4.4.

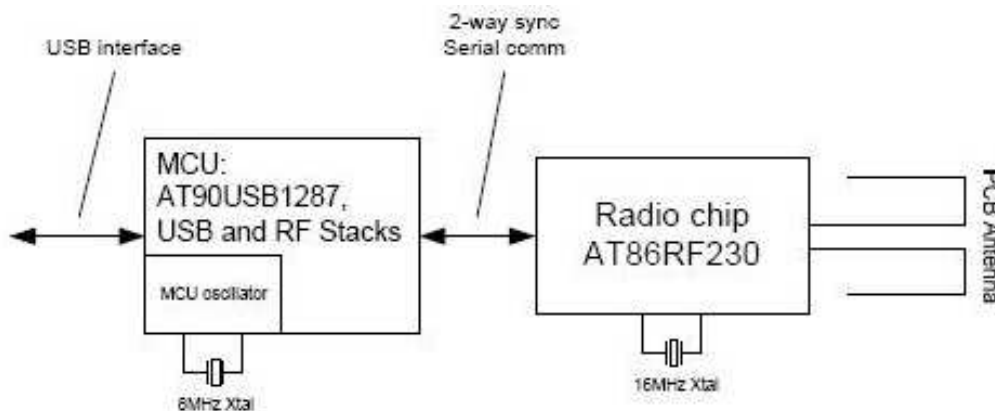


Fig. 4.4: Blocos do *Kit* de desenvolvimento ATAVRRZRAVEN [9] *USB Stick*

4.2.1 Medidores em Tomadas e Aparelhos Eletrônicos

Primeiramente, o AD71056 [6] se aplica a situações onde se o equipamento vai ser ligado a rede através de uma tomada comum como uma televisão, geladeira, etc.

Nestes casos um medidor de consumo será acoplado entre a tomada do aparelho e a rede, fazendo com que a corrente elétrica consumida pelo aparelho residencial seja amostrada pelo medidor. Isso pode ser ilustrado na Fig. 4.5.

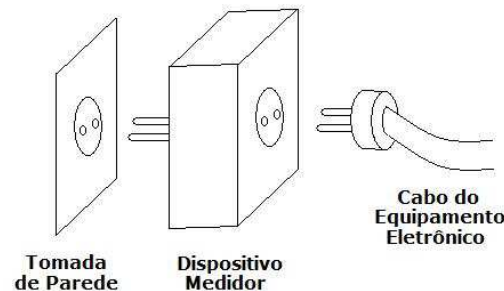


Fig. 4.5: Medidor de Tomada

Dessa forma, fica simples para que a instalação do ponto de medida seja feita, basta apenas desconectar o equipamento residencial da rede e adicionar o medidor entre o cabo e a tomada.

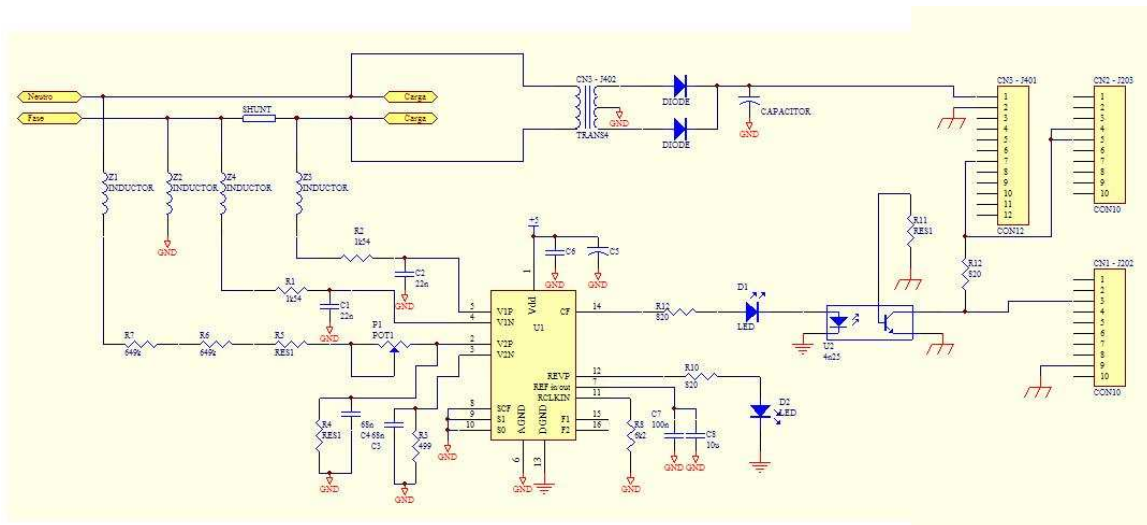


Fig. 4.6: Diagrama Esquemático do Circuito Medidor de Tomada

Portanto, a instalação deste tipo de medidor é um dos mais importantes pois deve ser rápida e fácil pois o número de equipamentos elétricos e eletrônicos em uma residência pode ser muito grande.

Outro ponto importante é de que este tipo de medida utiliza a rede, isso faz com que o circuito do medidor possa ser também alimentado pela rede ao invés da utilização de baterias.

O circuito sugerido pode ser visto na Fig. 4.6.

Ele é composto de uma fonte de 5V, dada pelo circuito abaixo que é um simples retificador de onda completa. Esta fonte é utilizada para a alimentação do circuito medidor e também do *kit Raven*.

Pode-se verificar que não há resistores e nem potenciômetro de calibração, pois este é feito através do *software* no microcontrolador.

Há também a presença de um acoplador óptico, onde a saída dos pulsos são isoladamente enviadas através do conector J202.

Pulsos proporcionais ao consumo de energia é a principal característica do componente AD71056 [6]. E que portanto necessita de um "contador" de pulsos para a obtenção das medidas.

4.2.2 Medidas em Lâmpadas

Normalmente, a instalação elétrica para lâmpadas é composto por interruptores que ficam embutidos nas paredes e as lâmpadas são instaladas em lustres tornando o acesso aos fios da instalação difícil.

Além do acesso, há ainda a instalação do medidor que poderia ser feita num interruptor ou ainda dentro do lustre da lâmpada. Ambos são de difícil instalação, pois há a necessidade de "mexer" na instalação elétrica da casa.

Dessa forma, optou-se pela utilização de um circuito que apenas detecta a presença ou não da luz, dizendo se a lâmpada está acesa ou apagada ilustrado na Fig. 4.7. Isso torna a instalação simples e rápida pois com apenas um adesivo, este dispositivo pode ser colocado bem ao lado de cada luminária ou lâmpada, devendo apenas verificar a potência nominal de cada uma das lâmpadas para a configuração do sistema.

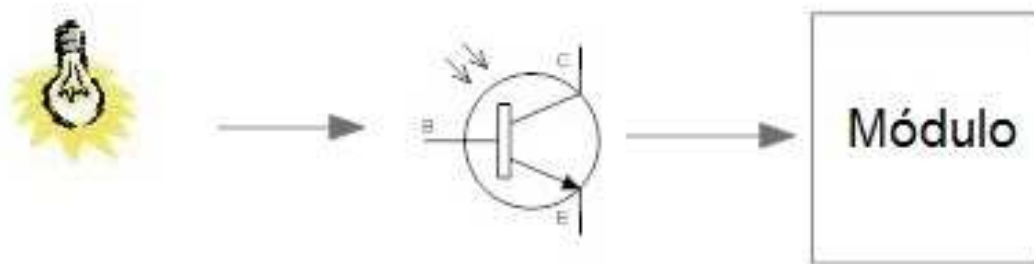


Fig. 4.7: Diagrama em Blocos do Circuito Medidor de Lâmpadas

A medida da potência consumida da lâmpada é dada pelo tempo em que a lâmpada fica acesa, este tempo é multiplicado pela potência nominal, obtendo-se então a potência consumida pela lâmpada.

4.2.3 Medidas no Chuveiro

O chuveiro é outro tipo de medida que deve ser feita de outra maneira, pois como não se deve alterar a instalação elétrica da casa e não há uma conexão como uma tomada para o chuveiro, a solução é a instalação de um circuito que funcione como um alicate amperímetro.

Dessa forma o instalador ao chegar no chuveiro faz o acoplamento do circuito medidor através de uma "garra" em um dos fios do chuveiro. Essa "garra" tem a finalidade de monitorar a corrente elétrica consumida pelo chuveiro. Esta medida é então multiplicada com o tempo de uso e pela tensão do chuveiro que pode ser de 127V e 220V.

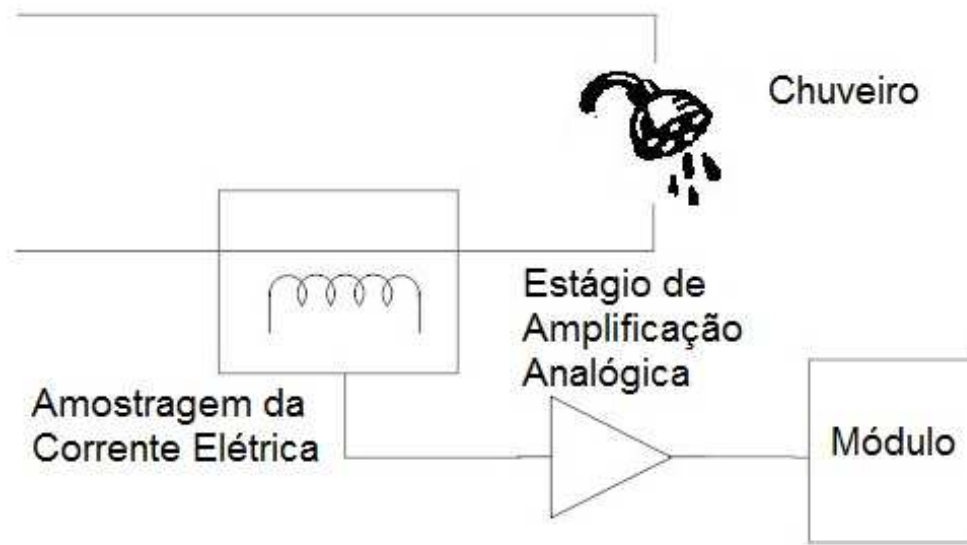


Fig. 4.8: Diagrama em Blocos do Circuito do Chuveiro

A Fig. 4.8 ilustra o funcionamento do circuito de amostragem de corrente elétrica no chuveiro.

4.2.4 Interfaces

O projeto possui três tipos de placas que serão acopladas com o *kit Raven* dependendo do tipo de aplicação, se tomada, lâmpada ou chuveiro como visto na Sessão 4.2.

Para se ter uma implementação de maior facilidade de montagem e instalação, optou-se por desenvolver uma forma de configuração, fazendo com que o microcontrolador saiba qual a placa que esta sendo acoplada.

Dessa forma, foram definidas as seguintes interfaces na Tab. 4.1.

Tab. 4.1: Configuração das Placas de Monitoramento

Configuração	Pino J203-4	Pino J203-5
Tomada	0	0
Lâmpada	0	1
Chuveiro	1	0

A pinagem utilizada para o sinal de medida acaba sendo também diferenciada para o circuito do chuveiro, sendo que sua medida é analógica esta deve ser introduzida no conversor Analógico-Digital do microcontrolador. Já para os outros dois tipos de placas, os pulsos ou níveis são enviados através de *GPIOs* digitais como ilustrado na Fig. 4.9.

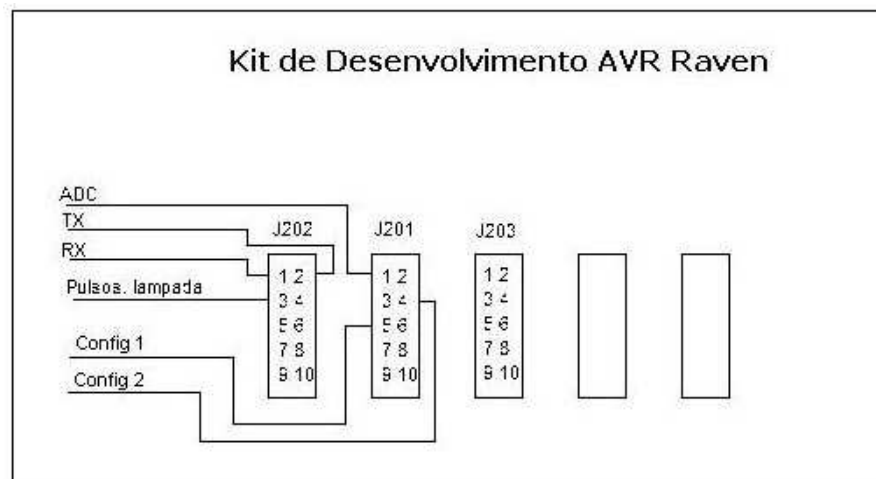


Fig. 4.9: Interfaces

A configuração dos pinos utilizados para a amostragem do sinal medido pode ser visto na Tab. 4.2.

Tab. 4.2: Pinagem utilizada para cada tipo de placa

Configuração	Tipo de Pino	Pinos
Tomada	GPIO Digital	J202-3
Lâmpada	GPIO Digital	J202-3
Chuveiro	ADC	J201-1
Gateway PLC	USART	J202-1 e J202-2

O *Gateway PLC* é uma extensão que permite que o módulo seja acoplado também a uma rede *PLC* e que não será abordado neste documento.

4.3 *Firmware*

O *Software* desenvolvido neste projeto é de vital importância, pois engloba não apenas conceitos de programação mas também toda a organização da rede *ZigBee*, a manipulação de pinos, amostragem e armazenamento.

Assim, o *software* pode ser dividido em várias partes, desde a sua plataforma de desenvolvimento até o envio e recebimento de mensagens na rede.

4.3.1 Plataforma

A Atmel [10] normalmente fornece uma plataforma de desenvolvimento para os seus *kits*. No caso do *AVR Raven* isso não é diferente, é a chamada *BitCloud*.

BitCloud é uma plataforma para *software* embarcado, ela provê a pilha de desenvolvimento necessária para o desenvolvimento de aplicações sem fio em alguns *hardware*. Esta plataforma pode ser utilizada em automações residenciais, comerciais, monitoramento, ou ainda industriais.

É compatível com os padrões *ZigBee* para monitoramento e controle provendo um conjunto de módulos e *APIs* para a manutenção e conformidade com as normas e padrões. Ela também oferece funcionalidades que podem ser extendidas pelos desenvolvedores, tornando-as customizadas para cada tipo de aplicação.

Com uma plataforma definida, o tempo de desenvolvimento acaba sendo menor, pois uma vez desenvolvida uma interface com algum tipo de dispositivo, esta não precisa ser novamente desenvolvida. Por exemplo o caso de um conversor Analógico-Digital, uma vez desenvolvida a interface de utilização deste componente, basta usá-la.

As principais vantagens da plataforma *BitCloud* são:

- Conformidade com *ZigBee PRO* e *ZigBee*;
- *APIs* em "C";
- Otimizada para aplicações de baixo consumo de energia;
- Possível extensão das *APIs* de segurança;
- Suporte para *upgrade* de *software*;

Arquitetura

A arquitetura interna da plataforma *BitCloud* segue as camadas de implementação *ZigBee* vistas na Sessão 3.2, onde há a separação da pilha em camadas lógicas.

Além da pilha principal *ZigBee* contendo a implementação do protocolo, a plataforma contém camadas adicionais com a implementação de serviços compartilhados (como o gerenciador de dispositivos, segurança, ou *task manager*), abstração de *hardware* chamada de (HAL) (*Hardware Abstraction Layer*) e *BSP* (Board Support Package).

Estas *APIs* estão na verdade fora da pilha principal, entretanto isso acaba por diminuir a complexidade das aplicações e simplifica a integração. A pilha da plataforma pode ser vista na Fig. 4.10.

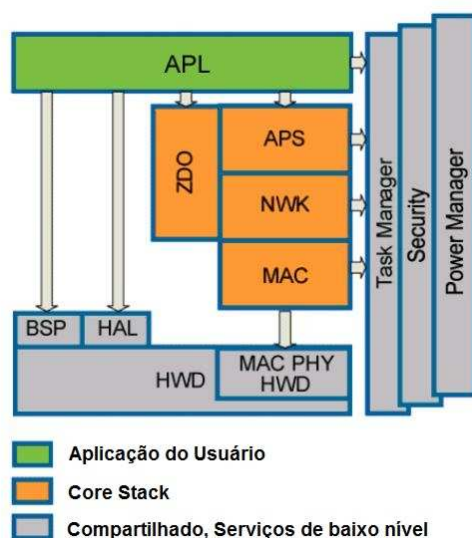
De forma a seguir os padrões de uma rede *ZigBee*, a camada mais alta é a *APS* que provê o mais alto nível das *APIs* visíveis para a aplicação.

A camada *ZDO* provê a conformidade com o padrão *ZDO* visto na Sessão 3.2 que habilita e faz o gerenciamento das funcionalidades de rede (inicialização, formação, associação, etc.). Ela também define os tipos de descritores, os dispositivos e os comandos de procura para ser implementados na pilha.

Há três serviços de componentes verticais que são: o gerenciador de tarefas, segurança e o gerenciador de recursos. Todos esses serviços podem ser utilizados pela aplicação ou por camadas inferiores da plataforma.

Os serviços principais que foram utilizados neste projeto são:

- **Gerenciador de Tarefas (*Task Manager*):** é o escalonador de pilhas que gerencia o uso da *MCU* pelas pilhas internas de componentes e da aplicação. Ele implementa a prioridade das

Fig. 4.10: Pilha da Plataforma *Bitcloud* [15]

tarefas que é baseada no compartilhamento de uma mesma unidade de processamento para várias tarefas. Cada tarefa recebe uma determinada prioridade de execução, onde há uma concorrência na utilização de *MCU*.

- **Gerenciador de Recursos** (*Power Manager*): é o responsável pelo desligamento de toda a pilha e componentes, salvando todo o estado do sistema, preparação para entrada de modo *sleep* e restauração de todo o sistema quando este é acordado.
- **HAL**: este componente inclui um conjunto completo de *APIs* para o uso dos recursos de *hardware* como a *EEPROM*, *timers* e proporciona a rápida integração de periféricos externos como *IRQ*, *USART*, etc.
- **BSP**: é um conjunto de *drivers* que fazem o tratamento de recursos presentes na placa de desenvolvimento, como botões, sensores, etc.

Modo de Operação

Normalmente, sistemas embarcados possuem alguns tipos de paradigmas de programação para a máxima economia de memória já que é um recurso muito pequeno quando comparado a computadores pessoais.

Dessa forma, o conceito de programação orientada a eventos é muito difundida em sistemas embarcados. Ele é uma referência a um estilo de programação e organização de arquitetura que possui uma paridade entre uma *API* e uma função de notificação assíncrona que seria o resultado da operação. Assim, quando é solicitada uma operação, uma função de retorno ou *callback* é também associada a esta operação. Como exemplo de funções síncronas e assíncronas temos a Fig. 4.11.

Todas as aplicações baseadas no plataforma *BitCloud* são programadas com orientação a eventos. Cada camada define um número de *callbacks* para que camadas mais baixas possam invocar e retornar nas funções de *callback* definidas nas camadas superiores.

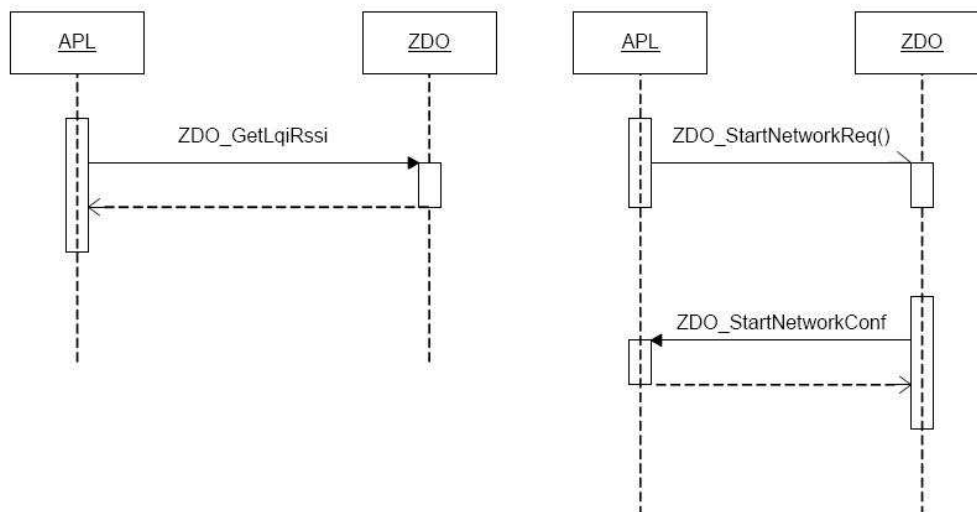


Fig. 4.11: Operação Síncrona e Assíncrona [15]

O mecanismo de requisição e confirmação é apenas uma instância da programação orientada a eventos. Simplesmente colocar uma requisição é uma chamada assíncrona em outra pilha para que alguma ação seja executada. A confirmação é quando a função de *callback* é executada, significando que a ação foi completada e o resultado da operação está pronto.

Para que isto ocorra, o principal ponto no desenvolvimento da aplicação é o controle de fluxo para garantir que as diferentes partes da aplicação não interfiram umas nas outras.

Normalmente, em sistemas não embarcados, o controle de acesso aos recursos de sistema é feito pelo sistema operacional, seja para a coordenação ou gerenciamento dos recursos compartilhados. Este tipo de sistema pode ser chamado de multitarefa.

A chamada das funções de *callbacks* estão normalmente no escopo de uma outra pilha, não a pilha da aplicação. Dessa forma, pode haver problemas se uma função de retorno demore muito tempo a ser executada podendo-se bloquear o sistema até que o retorno seja completo.

A recomendação é de que funções de retorno não demorem mais que 10ms para serem executadas [15].

4.3.2 Inicialização do Sistema

A inicialização do sistema *ZigBee* e melhor detalhada Sessão 3.5.

Ao mesmo tempo, há a verificação de qual tipo de placa está acoplada a placa do microcontrolador (AVR *Raven*). Isto pode ser visto no fluxograma da Fig. 4.12.

Após a identificação do tipo de placa acoplada, diferentes comportamentos acabam sendo executados dependendo do tipo de placa acoplada.

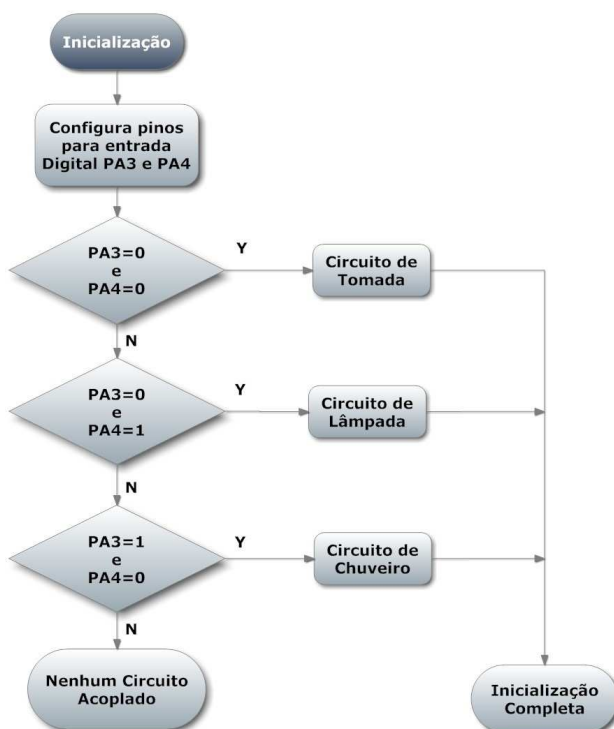


Fig. 4.12: Algoritmo de Inicialização do Sistema

4.3.3 Configuração e Funcionamento do Sistema de Medida em Tomadas

Como visto na Sessão 4.2.1, a saída do AD71056 [6] são pulsos, cuja frequência é proporcional ao consumo de energia.

Dessa forma, quando a placa acoplada ao *kit* for uma placa de Tomada, a configuração de medida do lado do microcontrolador é basicamente um contador de pulsos.

Este contador na verdade é atualizado por interrupções, toda vez que ele recebe um pulso. Para isso, o pino que recebe o sinal do medidor deve ser configurado para informar ao *software* que houve um pulso.

O microcontrolador ATmega1284P possui alguns pinos que podem ser configurados como interrupções. Interrupções externas podem ser de vários tipos, estes tipos são configurados no registrador EICRA do microcontrolador. A Fig. 4.13 ilustra o registrador EICRA.

Bit	7	6	5	4	3	2	1	0	
(0x69)	—	—	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	EICRA
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Fig. 4.13: Registrador EICRA [13]

O pino utilizado para isso é o PD2, que é conectado a interface J202-2 (ilustrada na Fig. 4.9). Os tipos de configuração de interrupções demonstrados na Tab. 4.3.

Tab. 4.3: Tipos de Interrupções

ISCn1	ISCn0	Configuração
0	0	Nível Baixo gera uma Interrupção
0	1	Qualquer Borda (Subida ou Descida) gera uma Interrupção
1	0	Borda de Descida gera uma Interrupção
1	1	Borda de Subida gera uma Interrupção

A configuração que mais se aplica ao tipo de sinal é a de Interrupção em apenas uma borda, podendo ser tanto a borda de subida quanto a de descida. Isto porque o sinal proveniente do medidor é um pulso que pode ser tratado tanto em sua borda de subida quanto de descida.

Neste projeto, a Interrupção foi configurada como borda de Descida, que pode ser visualizada na Tab. 4.4.

Tab. 4.4: Configuração da Interrupção para Interface com o Medidor de Tomada

ISCn1	ISCn0	Configuração
1	0	Borda de Descida gera uma Interrupção

Pode-se notar que não foi utilizada a *API* de interrupção da plataforma, isto porque apesar da informação dada pelo fabricante de que os recursos de *hardware* são abstraídos pelas *APIs* da camada *HAL*, não há uma documentação nem exemplos de utilização para o *kit Raven*. Dessa forma foi configurada através das primitivas básicas do microcontrolador. Obviamente estas primitivas também podem ser utilizadas mesmo em um plataforma.

O tratamento da interrupção gerada também não pode demorar muito a ser executada, e no caso apenas uma variável global faz o armazenamento dos pulsos sendo incrementada a cada interrupção gerada.

4.3.4 Configuração e Funcionamento do Sistema de Medida em Lâmpadas

As medidas de consumo para o caso de lâmpadas são provenientes da medida do tempo em que a lâmpada fica acesa.

Dessa forma, a medida a ser efetuada é o início do tempo em que a lâmpada é ligada até que esta seja desligada. A potência é dada pela seguinte equação

$$P = P_n \cdot t \quad (4.1)$$

Onde:

- P - Potência Consumida [W/h];
- P_n - Potência Nominal da Lâmpada [W];
- t - Tempo da Lâmpada ligada [h];

Tab. 4.5: Configuração da Interrupção para Interface com o Medidor de Lâmpada

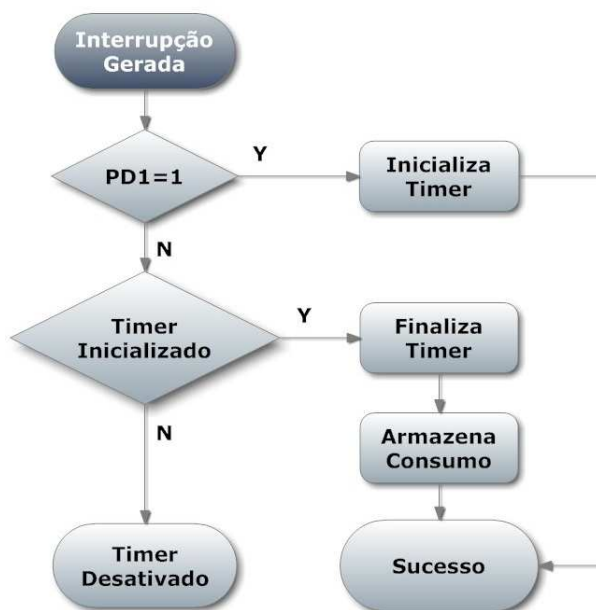
ISCn1	ISCn0	Configuração
0	1	Qualquer Borda (Subida ou Descida) gera uma Interrupção

Dessa forma, a configuração da interface pode ser similar à configuração de Tomada na Sessão 4.3.3, apenas alterando a configuração da interrupção para borda de subida e de descida que pode ser vista na Tab. 4.5.

Quando uma interrupção é gerada, o microcontrolador verifica o estado do pino, se este estiver em nível alto um *timer* é inicializado. Quando uma nova interrupção é gerada, novamente o estado do pino é verificado e estando em nível baixo o *timer* é parado.

O valor do tempo da lâmpada ligada é utilizada no cálculo da Equação 4.1 para que a potência consumida seja enviada ao coordenador do sistema.

O pino utilizado para isso é o mesmo PD2, que é conectado a interface J202-2 (ilustrada na Fig. 4.9).

Fig. 4.14: Algoritmo de Funcionamento do *software* para o Consumo em Lâmpada

O algoritmo pode ser visto na Fig. 4.14.

4.3.5 Configuração e Funcionamento do Sistema de Medida em Chuveiros

A medida de potência consumida do chuveiro é dada através da corrente elétrica medida nos fios que levam energia elétrica até o chuveiro por meio de uma "garra" como visto na Sessão 4.2.3.

Dessa forma, a medida realizada é de forma analógica, onde o valor medido é referente a intensidade da corrente elétrica no chuveiro. Para o cálculo da potência consumida é necessário o valor da tensão da rede que chega ao chuveiro. Assim, temos a seguinte equação:

$$P = V_{RMS} \cdot I_{RMS} \cdot t \quad (4.2)$$

Onde:

- P - Potência Consumida [W/h];
- V_{RMS} - Tensão da Rede [V];
- I_{RMS} - Corrente Medida [A];
- t - Tempo da Lâmpada ligada [h];

Para a visualização da medida, um conversor Analógico/Digital deve ser configurado e utilizado pelo microcontrolador. Assim, o pino PA0 através da interface de pino J203-1 (ilustrada na Fig. 4.9) é o caminho para que o sinal analógico medido chegue ao microcontrolador.

Neste caso, pôde-se utilizar do componente *HAL* para o tratamento conversor Analógico/Digital. A configuração do conversor é detalhada na Tab. 4.6.

Tab. 4.6: Configuração do Conversor Analógico/Digital

Configuração	Valor
Taxa de Amostragem	4800SPS
Resolução	8 bits
Referência	Interna

No caso do chuveiro não há nenhum tipo de interrupção, dessa forma o microcontrolador faz constantes medidas no *ADC* a fim de encontrar algum valor que considere o chuveiro como ligado, isto é, diferente de zero. O algoritmo é ilustrado na Fig. 4.15.

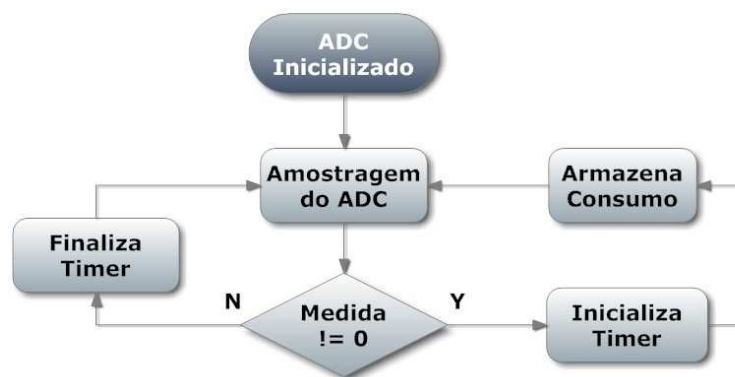


Fig. 4.15: Algoritmo de Funcionamento do *software* para o Consumo em Chuveiro

Pode-se verificar que há um *polling* no sinal analógico, onde a medida é efetuada constantemente.

4.3.6 Armazenamento dos Dados

As medidas de potência devem ser sempre guardadas em um local do microcontrolador onde não há problemas de falta de energia ou qualquer eventual problema que ocorra. Dessa forma, as informações podem ser armazenadas em memória *flash* ou na *EEPROM*.

A escrita na memória *flash* pode ser feita muitas vezes, em torno de 10.000. Mas possui um problema de tempo e tamanho da escrita. Neste microcontrolador a escrita em *flash* deve ser feita em uma página inteira de memória, mesmo sendo apenas um *byte*.

Já a escrita em uma *EEPROM* pode ser feita mais vezes (100.000) e pode ser feita em apenas um *byte*.

O microcontrolador ATmega1284P [13] possui 4K *bytes* de memória *EEPROM*. Assim, há três tipos de informação a ser armazenada:

- **Potência Consumida:** Esta potência consumida nada mais é que a potência medida em cada dispositivo. Se houver algum tipo de falha na comunicação o dispositivo continuará fazendo as medidas e fazendo seu armazenamento próprio. Durante a inicialização, o microcontrolador vai à posição da *EEPROM* onde o valor anterior foi guardado e assim a medida passa a ser constantemente atualizada;
- **Calibração:** A calibração vale apenas para os medidores de tomada. Todo medidor precisa ser calibrado e o valor do fator de calibração é guardado também na *EEPROM*. Quando o dispositivo detecta que a placa acoplada é uma placa de tomada, o microcontrolador vai à posição da *EEPROM* onde o valor da calibração foi guardada e o utiliza para o envio da mensagem contendo o valor do consumo já recalculado para o coordenador;
- **Potência Consumida Geral:** Esta Potência é o conjunto de todas as potências consumidas, válida apenas para o coordenador. Assim, cada dispositivo possui um determinado *ID* que é sua posição no mapa de memória contendo entradas para todas os valores consumidos de todos os dispositivos. Toda vez que uma mensagem é recebida contendo o valor de consumo, esta medida é comparada com a pré-existente no coordenador e então se for diferente é atualizada no banco de dados do coordenador.

Mapa de Memória *EEPROM*

O mapeamento de memória foi definido dado o tamanho da informação e quais são as informações a serem armazenadas. Obviamente que novos campos de informação podem ser adicionados num futuro trabalho, neste trabalho apenas é necessário informar ao usuário do sistema uma identificação do ponto de rede em que a medida está sendo efetuada, e quanto vale esta medida.

A Fig. 4.16 ilustra uma parcela da memória *EEPROM* limpa, cujos valores apenas possuem 0x00.

A forma de armazenagem como dito anteriormente na Sessão 4.3.6 é diferente dos pontos de rede que efetuam as medidas e o coordenador onde ficam armazenadas todas as informações dos sensores.

- **Sensores:** A informação a ser armazenada dinamicamente é a potência consumida por aquele ponto de rede. Isto é feito através da utilização de uma variável do tipo *UINT64* cujo tamanho é de 8 *bytes*, pode-se alcançar uma informação de número máximo 18.446.744.073.709.551.615.

[illegible]

Fig. 4.16: Mapa de Memória *EEPROM* limpa dos endereços 0x00 ao 0x90

O mapa de memória apresentado na Fig. 4.17, compreende a posição 0 da *EEPROM* onde se localiza o registro de medidas efetuadas. Neste exemplo, temos o valor de 0x223D pulsos medidos que é apresentado de forma invertida na memória. Outra informação também armazenada na *EEPROM* que é o valor da calibração, armazenado na posição de memória 0x256.

```
:1000000003D220000000000000000000000000000091
```

Fig. 4.17: Mapa de Memória *EEPROM* da medida do sensor armazenada

- **Coordenador:** Todas as medidas chegam ao coordenador através das mensagens de rede. O corpo da mensagem possui inúmeras informações referentes ao ponto de rede, mas as informações principais são a identificação do sensor e a medida de potência realizada pelo sensor até o momento. Dessa forma, a memória é dividida em unidades de 8 *bytes* para o armazenamento de cada sensor, sendo o índice desta medida. O funcionamento é ilustrado na Fig. 4.18.

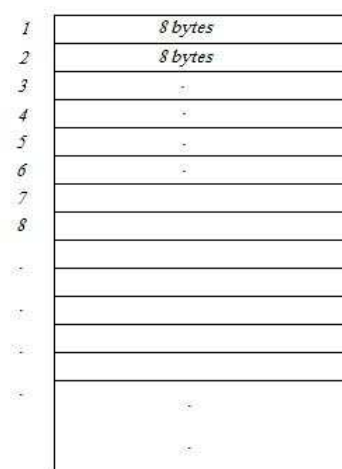


Fig. 4.18: Mapa de Memória *EEPROM* Coordenador

4.4 USB Stick e Coleta de Dados

Os dados das medidas estão armazenados na Central ou Coordenador. Para que estes dados cheguem de uma forma simples ao usuário, um novo ponto de rede precisa ser inserido, este fará a coleta de dados do Coordenador, enviando-os para um Computador.

Esta operação é feita por um dos módulos presentes no kit de desenvolvimento apresentado na Sessão 4.2. Este dispositivo possui uma interface *USB* que é utilizada para o envio dos dados requisitados para o computador onde está acoplado.

4.4.1 USB Driver e Emulação de uma Porta Serial

A comunicação de um dispositivo *USB* é dada através de um *driver*, cuja função é mapear características específicas de um determinado *Hardware* para que este seja entendido pelo Sistema Operacional. Para isto, dois arquivos de sistema são necessários: o *.inf* e o *.sys*.

- **.inf**: O propósito deste arquivo é a identificação do dispositivo conectado feito através do *Hardware ID*, a cópia dos arquivos binários para o sistema e adicionar as entradas no gerenciador de registros.
- **.sys**: É um arquivo de sistema que possui as informações necessárias para carregar e configurar um dispositivo, bem como as funções utilizadas pelo sistema operacional para a interação com o dispositivo acoplado.

Neste projeto, o *.inf* básico é provido pelo fabricante do componente e o *.sys* é utilizado diretamente do sistema operacional, dado que o dispositivo *USB* acoplado é mapeado como uma porta serial.

Para que isso aconteça, o *.inf* deve indicar o tipo de dispositivo a ser acoplado, no caso um conversor *USB* para Serial. Dessa forma, utiliza-se a comunicação *default* do *Windows* para porta serial: *usbser.sys* e *MsPorts.dll*.

Pode-se verificar a detecção e o funcionamento da porta serial pelo gerenciador de dispositivos, ilustrado na Fig. 4.18.

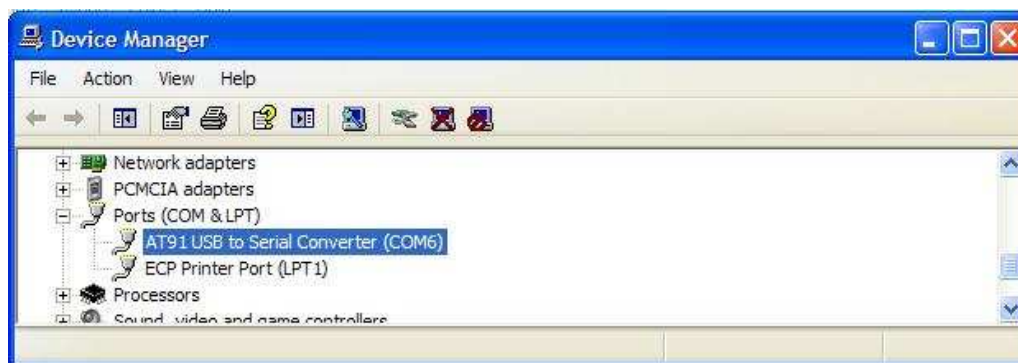


Fig. 4.19: Porta Serial no Gerenciador de Dispositivos

A configuração da porta serial (velocidade, número de *bits*, paridade, *bits* de parada e controle de fluxo) são ilustradas na Fig. 4.20.



Fig. 4.20: Configuração da Porta Serial

4.4.2 *Software de Comunicação entre o computador e o USB Stick*

A estrutura da Fig. 4.21, possui todos os campos da comunicação entre o computador e o *USB Stick*.

```
typedef struct
{
    unsigned _int8 cmd;
    unsigned _int8 id;
    unsigned _int8 type;
    unsigned _int8 measure[8];
} doutUARTMessage_t;
```

Fig. 4.21: Estrutura de Comunicação da Porta Serial

Tem-se então a seguinte descrição dos campos da estrutura:

- **cmd**: é o comando para a troca de mensagens
- **id**: é a identificação do dispositivo, em caso de envio de dados da central para o computador
- **type**: é o tipo do dispositivo conectado (Tomada, Lâmpada ou Chuveiro)
- **measure[8]**: é o valor medido para cada dispositivo num total de 8 *bytes*

Esta estrutura está presente tanto no lado do computador, como no lado do *USB Stick* para que a comunicação possa ocorrer de forma síncrona.

Dessa forma, foi desenvolvido um programa no computador em C++ para a aquisição de dados da Central para o Computador.

Seu funcionamento se dá pela requisição dos dados através de um comando enviado pelo computador para o *USB Stick*, o qual ao reconhecer este comando, envia um novo comando para a central

pedindo as medidas de todos os sensores. Esta então retorna com todos os dados que são então roteados para a porta serial. No Programa do computador, conforme os dados são recebidos, estes são escritos num arquivo. Este procedimento é ilustrado na Fig. 4.22.

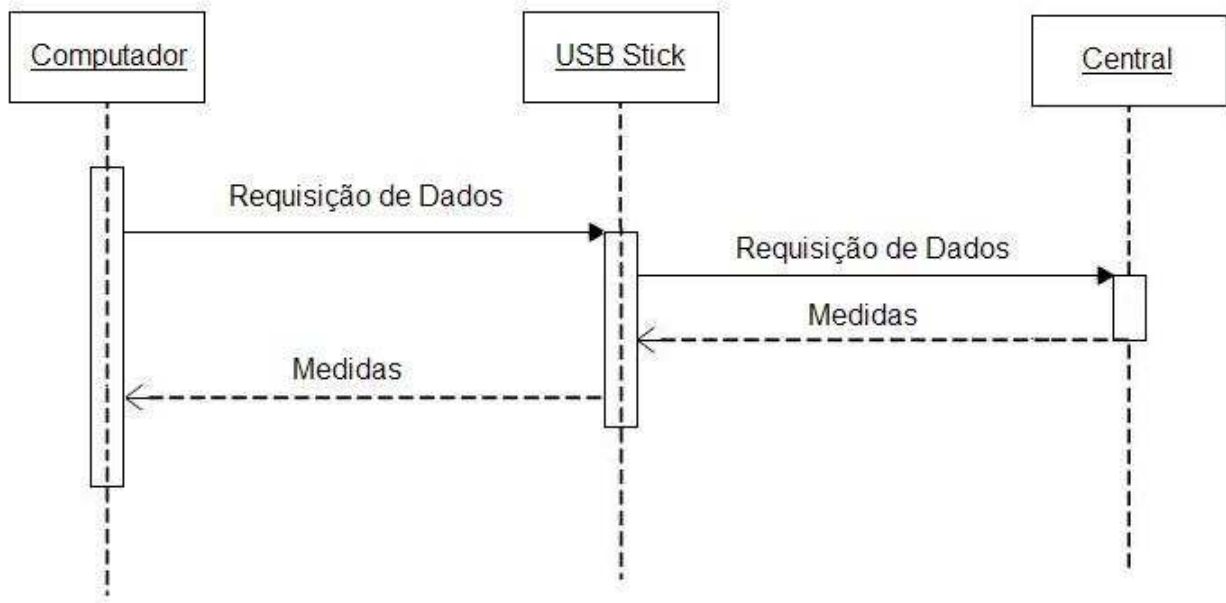


Fig. 4.22: Requisição e Aquisição das Medidas do Computador para a Central

O arquivo de saída possui as seguintes informações em hexadecimal:

- **id**: número da identificação do dispositivo
- **measure[8]**: valor da medida do dispositivo

Ilustrado na Fig. 4.23.

```

1,00000000047
2,00006adc792
3,00003fc53ca
  
```

Fig. 4.23: Arquivo de Saída com os Valores Medidos de cada Sensor

4.5 Software de Interface Gráfica com o Usuário

Os dados de todos os sensores e medidas que estão armazenados num arquivo em formato texto serão apresentados ao usuário. Foi então desenvolvido um *software* servindo como uma interface gráfica de fácil visualização e manipulação das medidas efetuadas.

Os requisitos para a interface com o usuário são:

- Tabela para entrada das configurações de cada sensor, contendo o **ID**, **Nome** e **Tipo**.
- Relatório de saída com as medidas de cada sensor.

Na Fig. 4.24 é ilustrada a tabela de configuração dos sensores.

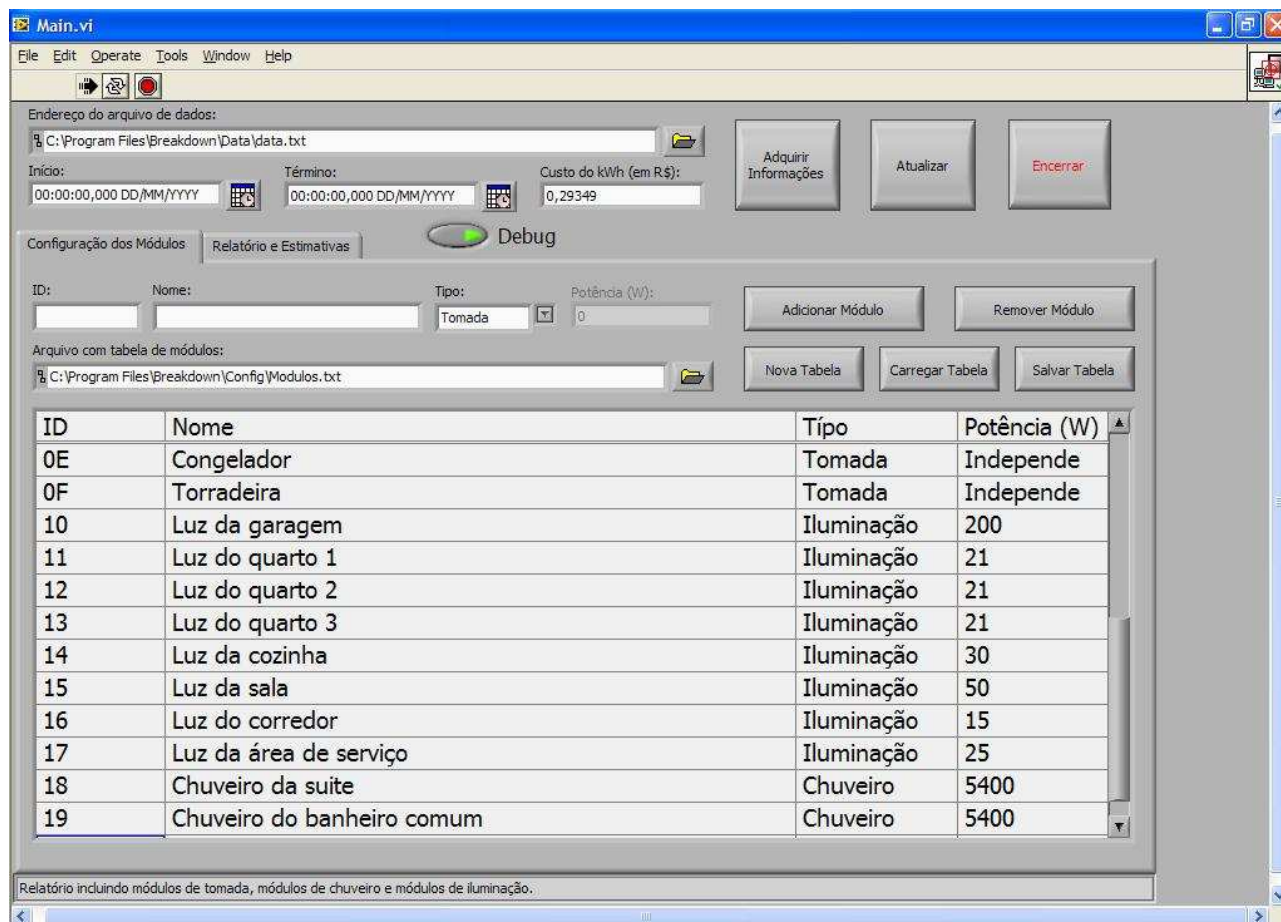


Fig. 4.24: Tabela de Entrada das Configurações de Cada Sensor

Como exemplo foram adicionadas entradas de diferentes tipos, sendo tomada, lâmpada e chuveiro.

O *software* permite a criação de tabelas novas, ou mesmo a utilização de tabelas antigas que podem ser referência para diferentes residências a serem medidas através dos botões "Nova Tabela", "Carregar Tabela" e "Salvar Tabela".

A adição e remoção de campos é feita através dos botões "Adicionar Módulo" e "Remover Módulo". No caso de adição, basta configurar o campo com seu ID, Nome e Tipo. Em caso de Chuveiro e Lâmpada, deve-se configurar também a potência nominal ilustrado na Fig. 4.25.

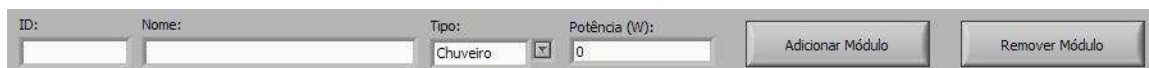


Fig. 4.25: Campos de Configuração de Cada Sensor

A aquisição de dados é feita pelo botão "Adquirir Informações", atualizando o relatório de saída pelo botão "Atualizar". O seu funcionamento básico é ilustrado pela Fig. 4.26.

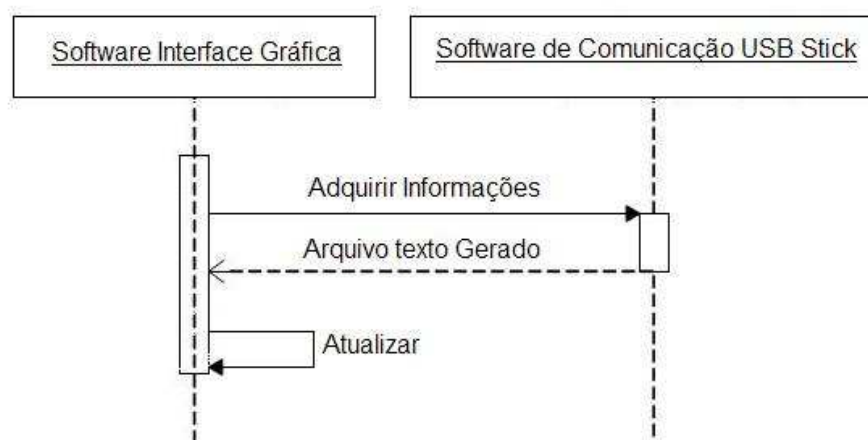


Fig. 4.26: Comunicação entre os *Softwares* de Interface Gráfica e Interface *USB-Stick*

A Fig. 4.27 ilustra o relatório de saída depois da aquisição dos dados e da atualização.

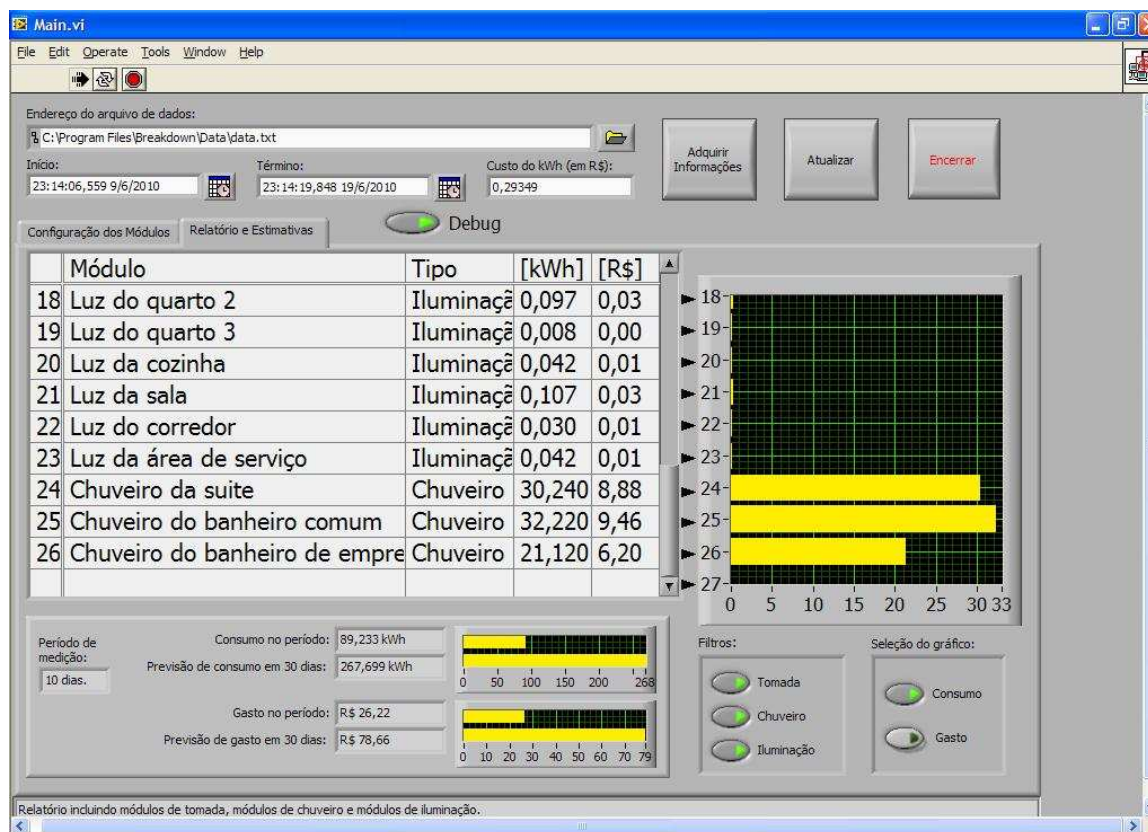


Fig. 4.27: Relatório de Saída

O relatório de saída possui basicamente o valor medido por cada sensor, cujo resultado é dado na unidade de [kWh]. Outro campo é o valor gasto pelo usuário já em unidade financeira dependendo do valor inserido no campo "Custo do kWh em R\$", obtendo-se portanto o custo de cada módulo.

O total do kWh para cada módulo é feito de maneira diferente para cada tipo de módulo:

- **Tomada:** Valor Recebido multiplicado por 1000 - Valor padrão de saída para o AD71056 [6];
- **Chuveiro:** Valor Recebido multiplicado pelo valor nominal configurado no *software* de Interface com o Usuário.
- **Lâmpada:** Valor Recebido multiplicado pelo valor nominal configurado no *software* de Interface com o Usuário.

O impacto de cada módulo no consumo total de uma residência também é demonstrado graficamente no *software*. Na Fig. 4.28 temos como exemplo o "Chuveiro do banheiro comum" que é o maior consumo quando comparado com outros sensores.

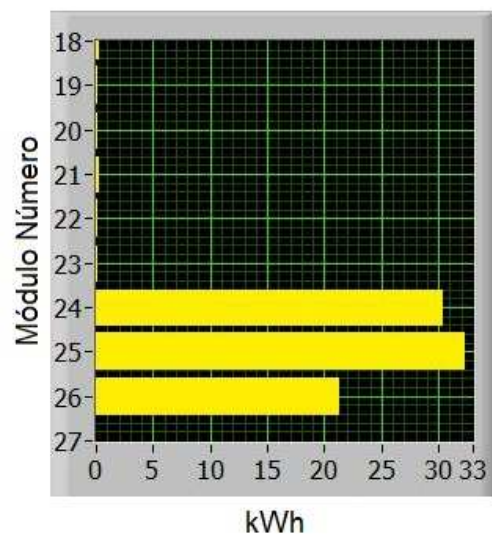


Fig. 4.28: Impacto Cada Módulo visto de Forma Gráfica

Também foi introduzido o fator tempo, neste caso o usuário insere a data de início e término da medição. Este tempo normalmente deve ser de 10 dias, dessa forma uma projeção para o consumo no total de 30 dias pode ser previsto e é ilustrado na Fig. 4.29. Obviamente que quanto maior a amostragem de tempo maior é a confiabilidade do valor amostrado e consequentemente do seu relatório final, por isso uma amostragem de 10 dias seria uma amostragem razoável para a determinação do consumo mensal.

Outras funcionalidades foram desenvolvidas para facilitar a visualização dos módulos, como por exemplo filtros onde qualquer tipo de módulo pode ser mostrado ou removido do relatório. Isso facilita principalmente quando se quer buscar informações relativas a apenas módulos do tipo Chuveiro como exemplo por geralmente se tratar da maior parte do consumo.

O gráfico ilustrado pela Fig. 4.28 além de representar o consumo gráfico de cada módulo, pode ilustrar diretamente o custo tornando sua escala em unidade financeira.

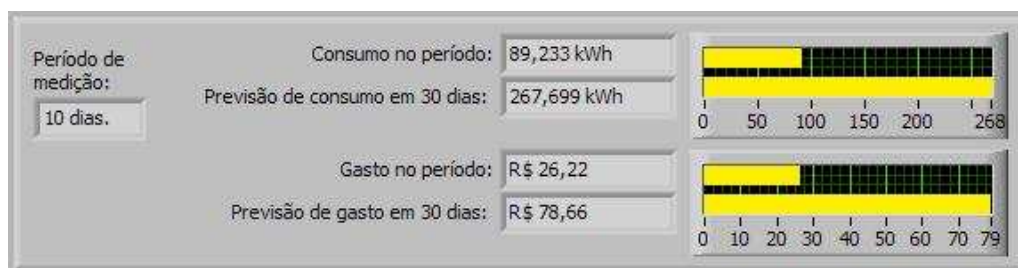


Fig. 4.29: Previsão de Consumo e Custo para o Total do Mês

4.6 Otimização do Consumo dos Módulos

Um ponto muito importante no projeto é o consumo de cada módulo, especialmente para os que são alimentados por bateria. Dessa forma, um estudo de como se comporta a corrente elétrica nos módulos é importante, bem como encontrar a melhor forma de otimizá-la.

Utilizando um resistor *shunt* pode-se verificar o consumo elétrico de forma gráfica. O circuito da Fig. 4.30 demonstra como a forma de onda da corrente consumida é medida.

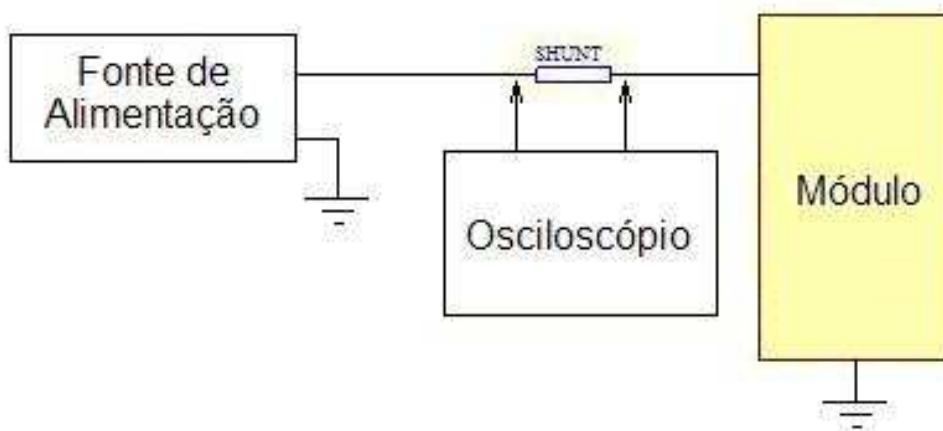


Fig. 4.30: Circuito para o Monitoramento do Consumo do Módulo

O Resistor *shunt* utilizado tem o valor de 68 Ohms gerando a forma de onda ilustrada pela Fig. 4.31.

A configuração de rede para este teste possui um *sleep time* de 5s, que é o tempo entre os pulsos de transmissão.

Este é o primeiro ponto a ser otimizado, sendo que a transmissão de uma medida pode ser feita depois de um tempo maior. Quanto maior o *sleep time* maior será a economia de energia já que o intervalo entre os pulsos de transmissão também aumenta.

Outro ponto importante é a corrente base. A Fig. 4.31 mostra que esta corrente é de 2mA. Isto significa que periféricos da placa do *kit* de desenvolvimento estão consumindo corrente mesmo com o módulo em modo *sleep*.

Pode-se verificar também um consumo médio de 15mA para a transmissão de dados.

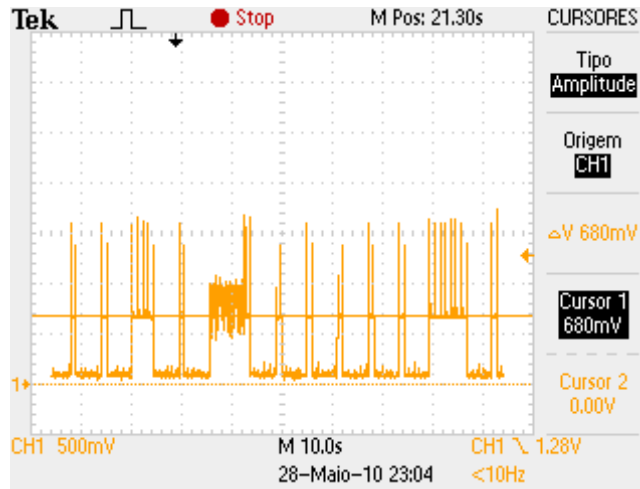


Fig. 4.31: Gráfico de Consumo de um Módulo

Verificando este consumo para uma bateria de 1200mA e considerando apenas o consumo da corrente base, temos a previsão de tempo de vida desta bateria dada pela Equação 4.3.

$$T = \frac{CB}{CM.NH} \quad (4.3)$$

Onde:

- CB - Carga da Bateria [mA/H];
- CM - Consumo do Módulo [mA];
- NH - Número de Horas em um Dia [H];
- T - Tempo total [Dias];

O resultado é um tempo de 25 dias, relativamente pouco considerando que apenas o consumo básico representa menos do período de 30 dias.

A primeira medida tomada foi a remoção do microcontrolador ATMEGA3209P [12], que é responsável pela alimentação dos periféricos da placa como o *display*, microfone, auto-falante, etc. A nova corrente em *sleep* é ilustrada na Fig. 4.32.

Dessa forma o consumo diminuiu para $140\mu A$ o que torna o resultado da Equação 4.3 igual a 357 dias.

Como a corrente de $140\mu A$ é muito baixa para ser visualizada no osciloscópio, um Multímetro com fundo de escala $0,1\mu A$ foi utilizado para visualização desta medida. Esta medida e o *setup* da bancada para a realização do teste pode ser observada na Fig. 4.33.

Mesmo com esta diminuição, uma corrente de $140\mu A$ ainda não é totalmente satisfatória pois foi apenas levada em consideração a corrente de *sleep*, pois há vários momentos em que o circuito precisa enviar os dados para o Coordenador onde o consumo do processador e da interface de rádio são relativamente altos por volta de 10mA.

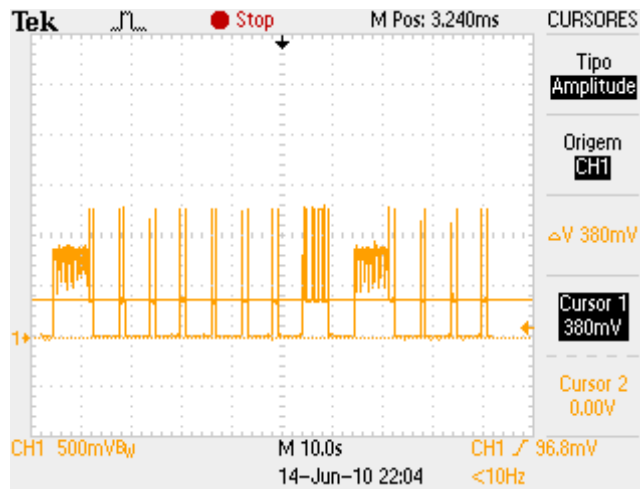


Fig. 4.32: Gráfico de Consumo de um Módulo sem o ATMEGA3209P [12]

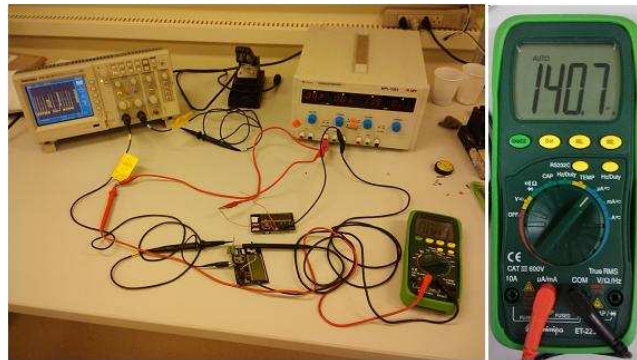


Fig. 4.33: Setup de Bancada e Medida de Corrente de Entrada do Módulo com Multímetro

Por outro lado, a corrente de *sleep* é o consumo predominante, estado em que o módulo deve ficar em quase 100

A estimativa de consumo total leva em conta também os pulsos, analisando cada um dos pulsos ilustrados pela Fig. 4.32:

1. 6 pulsos de 1s e amplitude 5,59mA
2. 12 pulsos de 50ms e amplitude 17,6mA
3. 1 pulso de 8s e amplitude média 11,03mA
4. 1 pulso de 500ms e amplitude média 11,03mA

Dado que o intervalo de tempo entre cada pulso é de 5s, o consumo total é dado pela Equação 4.4 para o período de 1 hora.

$$T = \frac{(6 \cdot 5,59) + (12 \cdot 0,05 \cdot 17,6) + (8 \cdot 11,03) + (0,5 \cdot 11,03)}{60} \quad (4.4)$$

O resultado é T igual a 2,3mA que é o consumo médio do sistema. Este valor é somado com o consumo base de 140 μ A para o restante do período (45s) onde os pulsos maiores não ocorrem resultando num valor de 2,4mA.

Considerando que o intervalo entre pulsos é de 5s, e será alterado para 10 minutos, a estimativa de consumo é demonstrada na Equação 4.5.

$$T = \frac{(6 \cdot 5,59) + (12 \cdot 0,05 \cdot 17,6) + (8 \cdot 11,03) + (0,5 \cdot 11,03)}{60 \cdot 120} \quad (4.5)$$

O valor de T agora é de 19 μ A que é então somado ao outros 140 μ A do restante do tempo, resultando num total de 159 μ A.

O fator 120 é adicionado, pois a alteração de 5 segundos para 10 minutos implica numa relação de 120 vezes.

Dessa forma, a estimativa para uma bateria de 1200mA seguindo a Equação 4.3 é de 314 dias.

Capítulo 5

Resultados Experimentais

5.1 Introdução

Este Capítulo está organizado de maneira a apresentar os resultados obtidos durante a elaboração deste trabalho, apresentando uma simulação de pontos de consumo e a obtenção destes dados até a demonstração destes dados no *Software* de Monitoramento para o Usuário.

5.2 Metodologia do Teste

A metodologia de teste teve foco principal a utilização da rede de sensores para comprovar seu funcionamento. Neste sentido, as principais variáveis foram testadas, tais como os tipos de sensores e a configuração de rede. O tempo de duração total do teste foi de aproximadamente 8 horas.

5.2.1 A Rede

O teste realizado emula todos os tipos de sensores possíveis na rede. Neste sentido, 4 pontos de rede foram configurados:

- **Coordenador.**
- **Sensor 1:** Tomada - *End Device*
- **Sensor 2:** Chuveiro - Roteador
- **Sensor 3:** Lâmpada - *End Device*

A disposição de cada um dos quatro pontos de rede é ilustrada na Figura 5.1.

O Sensor 3 não tem potência de sinal suficiente para se conectar ao Coordenador diretamente. Dessa forma, o Sensor 2 foi configurado como Roteador.

Esta configuração de rede força uma transmissão diretamente ligada ao Coordenador através dos Sensores 1 e 2, mas também através de um roteador feita no Sensor 2 para que o sinal chegue ao Sensor 3.

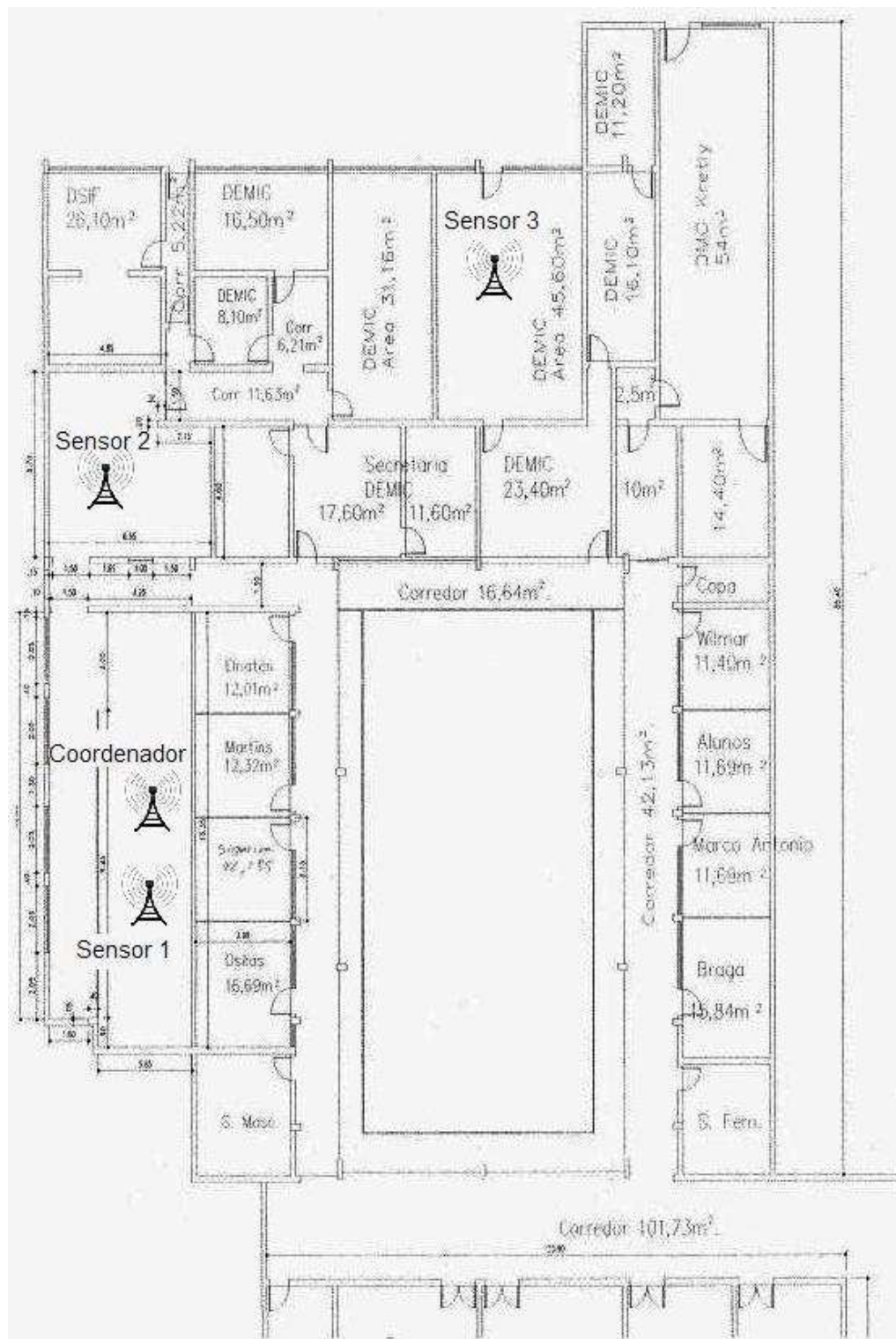


Fig. 5.1: Disposição dos Sensores

5.2.2 Os Sensores

A forma com que os sensores foram estimulados foi através de geradores de função. Dessa forma o consumo de energia foi emulado utilizando valores que são normalmente utilizados em uma residência para o chuveiro, lâmpada e tomada.

- **Sensor 1 - Tomada:** Como visto na Sessão 4.3.3, um módulo configurado para Tomada registra um pulso de consumo pela borda de descida. Portando um gerador de sinal de amplitude de 0 a 3V e frequência 1Hz foi utilizado para a emulação de um determinado equipamento eletrônico ligado e sendo monitorado.
- **Sensor 2 - Chuveiro:** Como visto na Sessão 4.3.5, um módulo configurado para Chuveiro registra o valor de corrente medido multiplicado pelo tempo que este foi usado e que portanto não funciona através de interrupções. Neste caso, um gerador de sinal de amplitude de 0 a 1V e frequência 0,07Hz foi utilizado para a emulação de um chuveiro ligando e desligando.
- **Sensor 3 - Lâmpada:** Como visto na Sessão 4.3.4, um módulo configurado para Lâmpada registra o tempo entre a borda de subida e de descida de um pulso. Portanto, um gerador de sinal de amplitude de 0 a 3V e frequência 0,05Hz foi utilizado para a emulação de uma lâmpada sendo ligada e desligada.

5.2.3 Configuração do *Software* de Interface com o Usuário

Os sensores são configurados dependendo de algumas variáveis, tais como que tipo de medida será executada até seu número de identificação *ID*. Estas configurações precisam estar de acordo tanto no módulo quanto no *software* de Interface Gráfica com o Usuário.

A configuração de cada sensor é descrita pela Tab. ??.

Tab. 5.1: Configuração dos Módulos

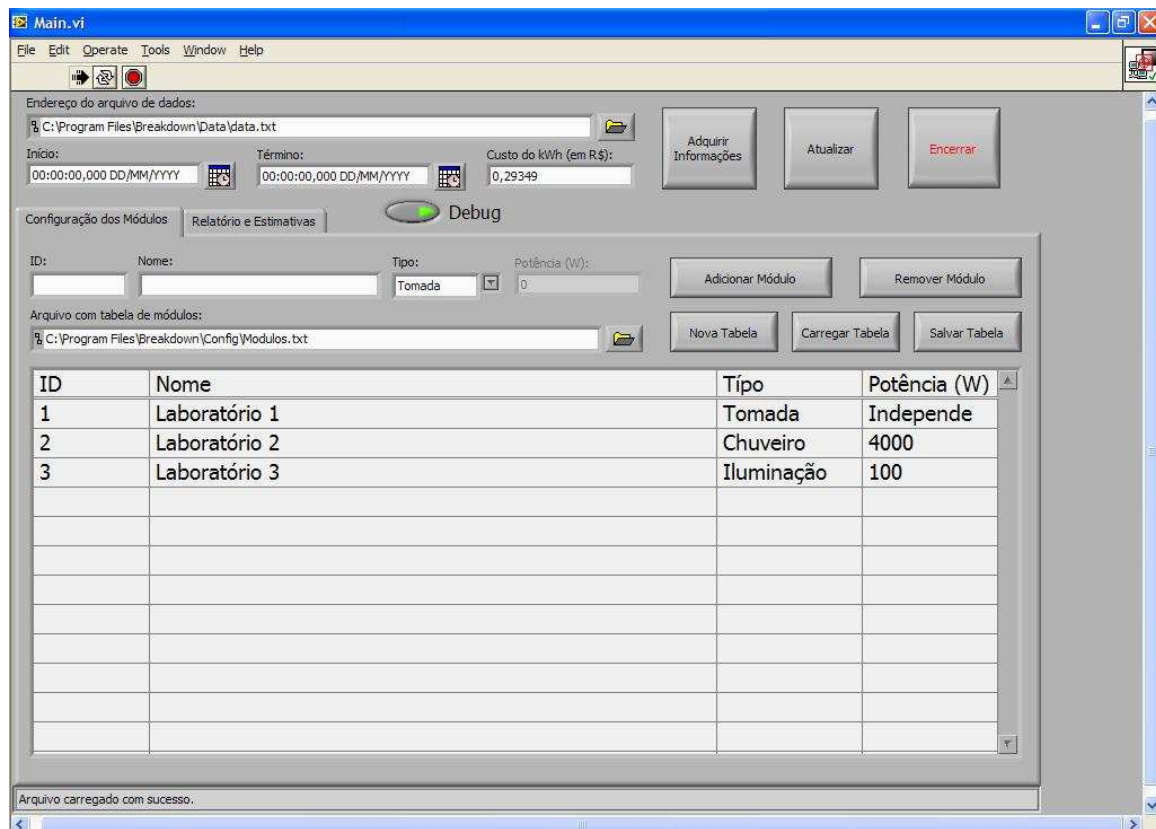
Módulo	<i>ID</i>	Nome	Tipo	Potência	Tipo do Módulo
Sensor 1	0x01	Laboratório 1	Tomada	Independente	<i>End Device</i>
Sensor 2	0x02	Laboratório 2	Chuveiro	4000W	Roteador
Sensor 3	0x03	Laboratório 3	Lâmpada	100W	<i>End Device</i>

O valor utilizado para a simulação de um chuveiro é a potência nominal de um chuveiro convencional em sua máxima potência. Já o valor utilizado para a simulação da lâmpada é a potência nominal de uma lâmpada incandescente.

Dessa forma, a configuração da Interface com o Usuário é ilustrada na Fig. 5.2.

Outros pontos a serem considerados:

- Não há datas de Início e Término configuradas pois o teste realizado foi de horas, e não de dias.
- O preço do kWh utilizado foi configurado apenas como valor de referência.
- Os sensores foram desligados algumas vezes, simulando o desligamento total dos medidores. Sendo que as medidas devem ser mantidas nas bases de dados de cada dispositivo.

Fig. 5.2: Configuração do *Software* de Interface com o Usuário

5.3 Resultados Esperados

Todos os cálculos desta Sessão são detalhados na Sessão 4.5.

- **Tomada:** Para o circuito da tomada, cada pulso recebido já é o valor da potência consumida. Considerando então que cada pulso medido seja 1kWh, num teste de 8 horas sem interrupções a medida deve ficar em aproximadamente 28KWh dado pela Equação 5.1.

$$P = \frac{VM}{T} \quad (5.1)$$

Onde:

- VM - Valor Medido [W/H];
 - T - Tempo total do teste [horas];
 - P - Potência [KWh];
- **Lâmpada:** O valor recebido da lâmpada é o tempo em que está ficou ligada. Dessa forma, o valor esperado seria de aproximadamente 24kWh que é dado pela Equação 5.2.

$$P = \frac{VM.VN}{60} \quad (5.2)$$

Onde:

- VM - Valor Medido [W/H];
 - VN - Valor Nominal da Lâmpada [W];
 - P - Potência [KWh];
- **Chuveiro:** O valor recebido do chuveiro é o tempo em que este ficou ligado dada a medida de conversão feita pelo A/D. Dessa forma, o valor de medida sem interrupções seria de aproximadamente 960kWh que é dado pela Equação 5.3.

$$P = \frac{VM.VN}{60} \quad (5.3)$$

Onde:

- VM - Valor Medido [W/H];
- VN - Valor Nominal do Chuveiro [W];
- P - Potência [KWh];

5.4 Resultados

Após 8 horas de testes, cada dispositivo enviou seu consumo para a central, que os envia para o computador. A primeira interface de dados como visto na Sessão 4.4.2. Os pulsos recebidos pelo *USB-Stick* e enviados para o computador são ilustrados na Fig. 5.3.

```
1,00000000000005c15
2,000000000000030c3
3,000000000000035B5
```

Fig. 5.3: Arquivo de Saída

Após o processamento dos dados pelo *Software* de Interface com o Usuário como visto na Sessão 4.5, o Relatório final das medidas é ilustrado pela Fig. 5.4.

Os dados mostram que o consumo do sensor 2 foi muito maior que dos outros dois. Isto acontece pela potência do chuveiro que é muito maior dos outros sensores de lâmpada e tomada. Consequentemente, o chuveiro teve um impacto financeiro muito maior que os outros dois sensores.

Este impacto pode ser melhor visualizado pelo gráfico dos valores da tabela ilustrado pela Fig. 5.5.

Os dados são apenas ilustrativos e são de uma simulação, porém tentou-se encontrar valores de simulação próximos a uma residência real onde um dos maiores consumidores de energia é o chuveiro.

Dos resultados obtidos, pode-se chegar nas seguintes conclusões:

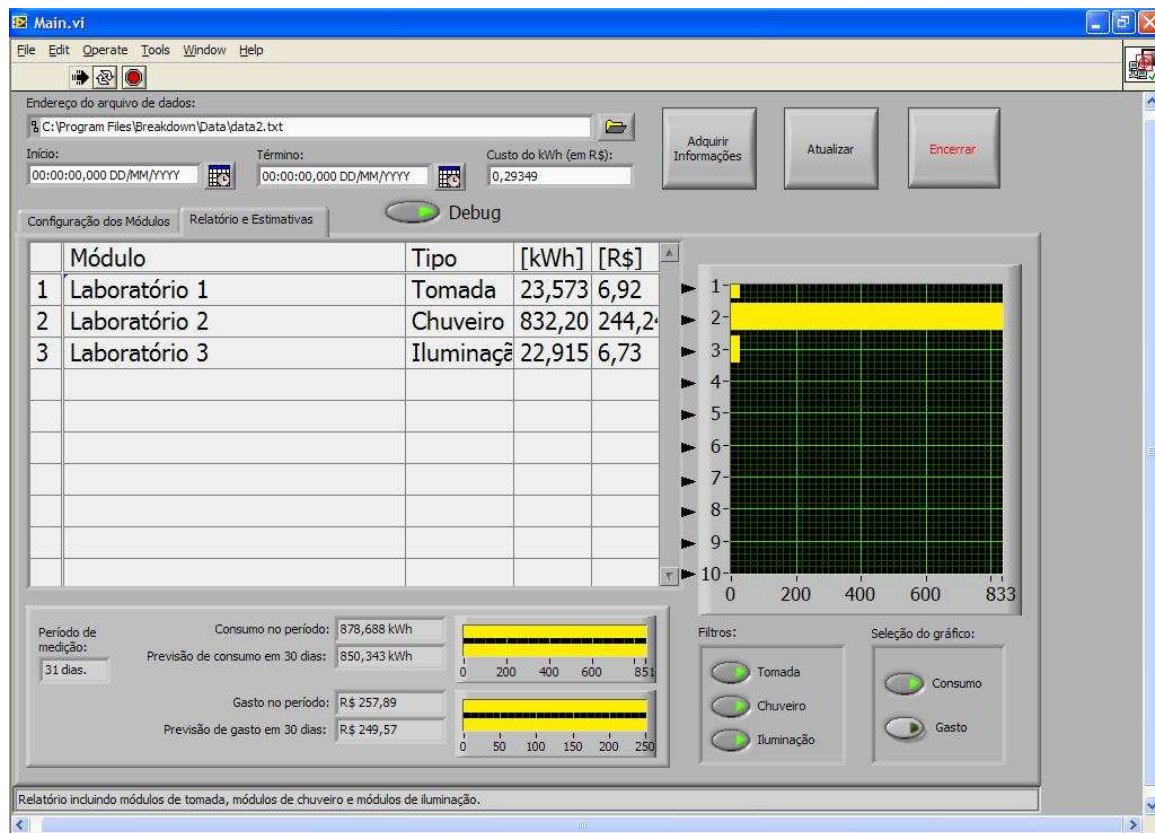


Fig. 5.4: Relatório Final do Teste

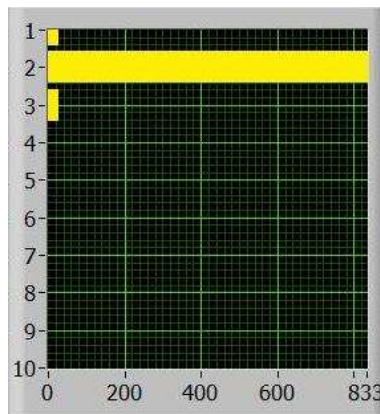


Fig. 5.5: Gráfico do Teste Final

- Os resultados obtidos demonstram os diferentes dispositivos de monitoramento do consumo de energia de uma forma funcional, desde a sua geração pelos medidores de energia até a geração do relatório final para o usuário com todos os dados de cada ponto da rede. De acordo com os resultados, a topologia adotada pela rede é ilustrada pela Fig. 5.6.
- Todos os tipos possíveis de consumo de energia elétrica numa residência foram cobertos. O



Fig. 5.6: Rede Final

sistema se portou bem em todos os tipos, desde os que operam em maiores níveis de medidas até os que de menor intensidade.

- Considerando os diferentes tipos de aquisição de dados, possíveis problemas em relação à estabilidade poderiam ocorrer mas o sistema se mostrou robusto o que pôde ser comprovado em um teste de 8 horas seguidas ininterruptas.
- A Tab. 5.2 ilustra a diferença dos valores estimados como máximos num teste de 8 horas sem o desligamento dos sensores e os valores medidos com interrupções para verificar seu correto funcionamento mesmo após serem desligados.

Tab. 5.2: Comparação entre os Valores Medidos e os Estimados

Sensor	Valor Estimado	Valor Medido
Tomada	Máximo de 28KWh	23.573KWh
Chuveiro	Máximo de 960KWh	832.20KWh
Lâmpada	Máximo de 24KWh	22.9KWh

Os resultados demonstram que o sistema se comporta muito bem mesmo sendo desligado, o que resulta em robustez pois mesmo que ocorra qualquer tipo de falha os valores medidos estão sempre armazenados com segurança.

Capítulo 6

Considerações Finais e Trabalhos Futuros

Este trabalho foi focado na implementação e operação de um sistema sem fio para o monitoramento do consumo em cada ponto de uma residência.

Conforme apresentado na Sessão 1.4, todos os objetivos do projeto foram alcançados:

1. **Tipos de medidas:** A implementação do *firmware* leva em conta os três possíveis tipos de sensores: lâmpadas, chuveiro e tomadas. Cada tipo de sensor possui um tipo diferente de medida, seja ela pulsos, tempo de acionamento ou uma leitura analógica.
2. **Rede de Sensores:** A primeira parte do projeto foi a implementação de uma rede de sensores que pudesse atender as exigências para a transmissão de cada ponto de consumo de energia em uma residência. Uma das exigências do projeto é de que esta rede seja sem fio, fazendo com que a instalação dos dispositivos seja a mais simples possível.
3. **Topologia de Rede:** A rede de sensores *ZigBee* pode ser configurada de várias formas quanto a sua topologia, essa característica faz com que a rede implementada possa atender a todas as necessidades da residência a ser monitorada, pois todos os pontos de consumo podem ser alcançados.
4. **Armazenamento dos dados:** Os dados transmitidos de cada sensor são enviados para uma Central, que é o Coordenador da rede *ZigBee*.
5. **Coleta de Dados:** Para a geração de um relatório com os dados do consumo de cada ponto da rede, um *USB Stick* é conectado a um computador. Este dispositivo também é um módulo de comunicação *ZigBee* e é o responsável por receber os dados armazenados no Coordenador e enviar para o computador através da porta *USB*.
6. **Geração do Relatório de Consumo:** O relatório de Consumo é a interface gráfica dos dados recebidos com o usuário, nele o usuário encontra cada sensor e seu respectivo consumo.

O projeto teve um foco direto na simplicidade de operação e instalação do sistema, dessa forma não foi levado em conta nenhum tipo mais específico de operação ou mesmo de configuração para que este não dependa de um grande conhecimento da área. Dessa forma o projeto se torna uma ferramenta que pode ser utilizada por um usuário simples de computador.

6.1 Trabalhos Futuros

Como sugestões para melhorias e trabalhos futuros, temos:

1. Otimização do consumo:
 - (a) O consumo dos nós de rede que pode passar de $140\mu A$ para pouco mais de $100\mu A$.
 - (b) O consumo das larguras dos pulsos para a transmissão podem ser menores.
 - (c) O *payload* dos *bursts* de transmissão podem ser menores.
2. A escrita dos valores medidos pode ser melhor dimensionada, já que um sistema é alimentado via bateria, esta nunca irá ser desligada podendo chamar rotinas de escrita em *eeeprom*, que possuem um consumo de energia maior, de uma forma mais otimizada.
3. Disponibilização dos dados obtidos na Internet, para que um usuário possa verificar qual o consumo de sua residência de uma forma rápida e em qualquer lugar que possua acesso à Internet.

Referências Bibliográficas

- [1] ZIGBEE. ZigBee Alliance. Página na internet, Acesso em 09/08/2009. "<http://www.zigbee.org/>".
- [2] IEEE. Institute of Electrical and Electronics Engineers. Página na internet, Acesso em 20/11/2009. "<http://www.ieee.org/>".
- [3] Atmel. ATAVRRZRAVEN 2.4 GHz Evaluation and Starter Kit. Página na internet, Acesso em 10/11/2009. "http://www.atmel.com/dyn/Products/tools_card.asp?tool_id=4291".
- [4] Atmel. BitCloud. Página na internet, Acesso em 21/05/2009. "http://www.atmel.com/dyn/products/tools_card.asp?tool_id=4495".
- [5] Atmel. ATmega1284P. Página na internet, Acesso em 10/11/2009. "http://www.atmel.com/dyn/products/product_card.asp?part_id=4331".
- [6] Atmel. ATmega3290P. Página na internet, Acesso em 10/11/2009. "http://www.atmel.com/dyn/products/Product_card.asp?part_id=3550".
- [7] COPEL. Companhia Paranaense de Energia Elétrica. Página na internet, Acesso em 20/11/2009. "<http://www.copel.com/hpcopel/root/nivel2.jsp?endereco=%2Fhpcopel%2Fresidencial%2Fpagcopel2.nsf%2Fdocs%2F9C83B5131AF54B1B032573EC005D8B0D>".
- [8] AES Eletropaulo. AES Eletropaulo. Página na internet, Acesso em 18/11/2009. "<http://www.eletropaulo.com.br>".
- [9] ARSESP. Agência Reguladora de Saneamento e Energia do Estado de São Paulo. Página na internet, Acesso em 18/11/2009. "<http://www.arsesp.sp.gov.br>".
- [10] National Instruments. O que é uma Rede de Sensores Sem Fio? Página na internet, Acesso em 10/11/2009. "<http://digital.ni.com/worldwide/brazil.nsf/web/all/263C7D0112470D22862575FA0046DD11>".
- [11] Analog Devices. AD71056. Datasheet, Analog Devices, Inc., Acesso em 14/10/2009. "http://www.analog.com/static/imported-files/data_sheets/AD71056.pdf".

- [12] Ana Teruko Yokomizo Watanabe. Monitoramento controle e parametrização de eletrodomésticos numa rede sem fio via interface virtual remota. Master's thesis, Universidade do Estado de Santa Catarina - UDESC, Maio 2008.
- [13] Atmel. Atmel Corporation. Página na internet, 2009. "<http://www.atmel.com>".
- [14] Atmel. AT86RF230. Página na internet, Acesso em 10/11/2009. "http://www.atmel.com/dyn/products/product_card.asp?part_id=3941".
- [15] Atmel. AT90USB1287. Página na internet, Acesso em 10/11/2009. "http://www.atmel.com/dyn/products/product_card.asp?part_id=3875".