

VIRGÍLIO FRIES MÜLLER

**TRANSMISSÃO DE IMAGENS E VÍDEO UTILIZANDO O
PADRÃO IEEE 802.15.4 / ZIGBEE**

Ilha Solteira - SP
2015

VIRGÍLIO FRIES MULLER

**TRANSMISSÃO DE IMAGENS E VÍDEO UTILIZANDO O
PADRÃO IEEE 802.15.4 / ZIGBEE**

Dissertação apresentada à Faculdade de Engenharia do Câmpus de Ilha Solteira - UNESP como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica.

Especialidade: Automação.

Prof. Dr. Nobuo Oki
Orientador

Ilha Solteira - SP
2015

FICHA CATALOGRÁFICA

Elaborada pela Seção Técnica de Aquisição e Tratamento da Informação
Serviço Técnico de Biblioteca e Documentação da UNESP - Ilha Solteira.

Müller, Virgílio Fries.

M946t Transmissão de imagens e vídeo utilizando o padrão IEEE 802.15.4/ZIGBEE
/ Virgílio Fries Fries Müller. – Ilha Solteira: [s.n.], 2015
89 f. : il.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de
Engenharia de Ilha Solteira. Área de conhecimento: Automação e Controle.
Engenharia Elétrica., 2015

Orientador: Nobuo Oki

Inclui bibliografia

1. Redes sem fio. 2. ZIGBEE. 3. Compressão de imagens. 4. WPAN. 5. MPEG-4



UNIVERSIDADE ESTADUAL PAULISTA
CAMPUS DE ILHA SOLTEIRA
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA



CERTIFICADO DE APROVAÇÃO

TÍTULO: Transmissão de Imagens e Vídeo utilizando o Padrão IEEE 802.15.4/ZIGBEE

AUTOR: VIRGÍLIO FRIES MÜLLER

ORIENTADOR: Prof. Dr. NOBUO OKI

Aprovado como parte das exigências para obtenção do Título de Mestre em Engenharia Elétrica ,
Área: AUTOMAÇÃO, pela Comissão Examinadora:

Prof. Dr. NOBUO OKI
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira

Prof. Dr. SERGIO AZEVEDO DE OLIVEIRA
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira

Prof. Dr. TÉRCIO ALBERTO DOS SANTOS FILHO
Departamento de Ciências da Computação / Instituto de Biotecnologia- Universidade Federal de Goiás

Data da realização: 30 de julho de 2015.

*À minha mãe, Marlise Fries, por todo amor incondicional e
minha namorada Jaqueline, as mulheres de minha vida!*

AGRADECIMENTOS

Neste texto a seguir gostaria de expressar meus mais sinceros agradecimentos a todos que tornaram este trabalho possível e ajudaram cada um à sua maneira.

A meu Deus, Jesus Cristo, pela força, confiança e amor em todos os dias de minha vida.

Ao meu orientador Nobuo Oki, por toda a competência, disposição e atenção a mim prestadas em todos os momentos do trabalho.

A minha mãe Marlise Fries, pelo incondicional apoio, não só neste, mas em todos os projetos da minha vida, e por todos os esforços que já realizou para que eu pudesse chegar onde estou.

A minha namorada Jaqueline, por todo apoio, companheirismo, incentivo neste trabalho e principalmente pelo amor, carinho e amizade.

Ao meu amigo Marcos Antônio Estremote o qual sempre arrumava uma maneira de colaborar e principalmente motivar.

Aos professores do departamento de pós-graduação da Faculdade de Engenharia - UNESP Ilha Solteira, em especial ao Sérgio Azevedo de Oliveira, Ailton Akira Shinoda, Suely C. Amaro Mantovani, Alexandre César Rodrigues da Silva e Ruben Augusto Romero Lázaro.

Aos funcionários da UNESP Ilha Solteira que, direta ou indiretamente, colaboraram para a realização deste trabalho, principalmente ao Deoclécio Mitsuiti Kosaka, secretaria acadêmica de pós-graduação e aos guardas do câmpus III que foram companheiros de estímulo em finais de semana e feriados.

Aos sempre amigos Ápio Carnielo Silva, Adriano Cardoso, Flavilene Souza, Marcos Vinicius Alves Pereira, Tiago Almeida, Wellington Luis Codinhoto Garcia, Daniel Gonçalves, Leonardo Dal Pont, Emerson Ravazzi, Rafael Marcelino de Jesus, Denis Rogério da Silva, Watson Rogério Azevedo, Leonardo Pereira, Estélio Amorin, Abílio Gama Neto, Wagner e pôr fim a amiga Andrea Proto que me apresentou o programa de Pós-Graduação em Engenharia Elétrica acendendo à ideia.

Aos velhos amigos dos quais tive que me afastar, mas não me afastei da amizade, que sempre vai me acompanhar, por onde eu andar e sempre estarão comigo.

Aos meus professores de graduação que desde o começo me motivaram como Augusto

Venâncio Neto, Bruno Pinto, Marcelo Pelágio e Fabian Correa Cardoso.

À banca examinadora pela análise, contribuições e sugestões do conteúdo deste trabalho.

Aos que acreditaram em mim e aos que não acreditaram, por que foi através desses, que tive força para seguir lutando.

Muitíssimo obrigado.

*“O fracasso jamais me surpreenderá,
se minha decisão de vencer for suficiente forte.”*

Og Mandino

RESUMO

As demandas por soluções de comunicações de redes sem fio (tecnologia *wireless*) estão integradas ao cotidiano e cada dia mais inseridas e valorizadas em nosso meio. Neste trabalho realizou-se o estudo da viabilidade de utilização da rede sem fio WPAN (padrão IEEE 802.15.4) na transmissão de imagens e vídeo utilizando o padrão ZigBee. Utilizou-se neste trabalho, linguagens de programação JAVA e C, o software livre *Atube Catcher* para compressão de vídeos, módulos de conexão, placas e transceptor da Atmel. A comunicação e configuração dos módulos para a transmissão de dados pelo padrão ZigBee são realizadas através da porta serial. Testes efetuados com imagens e vídeo MPEG-4 mostraram a viabilidade desta transmissão, tendo sido feitas medições com compressão de arquivos com imagens e vídeo, das perdas de pacotes com distância e da latência de transmissão dos dados.

Palavras-chave: Compressão de Imagens. MPEG-4. Redes sem Fio. WPAN. ZigBee.

ABSTRACT

The demand for wireless network communications solutions technology wireless are integrated into the everyday and each day more embedded and valued in our midst. In this work the study of the feasibility of using wireless network WPAN (IEEE 802.15.4) for video transmission with the Zigbee standard was done. For improve the capability of transmission of images and video compression techniques were used. For accomplish of this work the following tools are used: JAVA and C programming languages, aTube Catcher free software for video compression, connection modules, Atmel boards and transceiver. Tests were performed with images and MPEG-4 video and the results showed the feasibility of transmission. Besides, measurements with file compression with images and video, packet loss ratio with distance and data transmission latency were made.

Keywords: Wireless Networks ZigBee. Image compression. WPAN. MPEG-4.

LISTA DE FIGURAS

Figura 1	Visão humana	24
Figura 2	Exemplo de imagem vetorial.	25
Figura 3	Exemplo de imagem <i>bitmap</i>	25
Figura 4	Elementos de um sistema de processamento de imagens.	26
Figura 5	Representação de um histograma de imagem.	27
Figura 6	Exemplo de mistura de cores RGB.	29
Figura 7	Cubo RGB.	29
Figura 8	Exemplo de mistura de cores RGB.	30
Figura 9	Imagem em tons de cinza.	31
Figura 10	Camadas do padrão de uma rede sem fio Zigbee.	37
Figura 11	Topologia estrela no padrão ZigBee.	38
Figura 12	Topologia em malha (mesh) no padrão ZigBee.	38
Figura 13	Topologia árvore de uma rede sem fio ZigBee.	39
Figura 14	Transferência de dados para um <i>coordenador</i> do padrão IEEE 802.15.4: a) Com sinal b) sem sinal	42
Figura 15	Transferência de dados de um coordenador para um dispositivo. a) com sinal b) sem sinal	43
Figura 16	Organização do frame WPAN.	44
Figura 17	Interface da IDE NetBeans versão 8.	48
Figura 18	Interface IDE AVR Studio.	49
Figura 19	Interface AVR Studio com WinAvr.	50
Figura 20	Interface Atube Catch.	51
Figura 21	Interface Interface WireShark.	52
Figura 22	Interface STK 500.	53

Figura 23	Módulo Atmel STK 501.	54
Figura 24	STK 501 conectado no STK 500.	54
Figura 25	Rádio Transceptor RZ502.	55
Figura 26	Módulo STK 500 + módulo STK 501 + rádio transceptor RZ502.	55
Figura 27	Módulo transceptor RZ RAVEN.	56
Figura 28	Fluxograma do algoritmo de transmissão de imagens e vídeo.	57
Figura 29	Tela de interface da comunicação ponto a ponto utilizando o padrão IEE 802.15.4.	57
Figura 30	Tela da interface Compressão.	58
Figura 31	Tela para seleção da imagem no navegador.	59
Figura 32	Tela da interface do software.	59
Figura 33	Tela da interface do software.	60
Figura 34	Fluxograma do software desenvolvido.	60
Figura 35	Tamanho do arquivo original selecionado.	61
Figura 36	Propriedades do Arquivo após a compressão	62
Figura 37	Comparativo de tamanho após compressão da imagem.	62
Figura 38	Apresentação de três imagens bitmap coloridas.	63
Figura 39	Apresentação de três imagens bitmap após ter sido compactadas pelo <i>software</i>	63
Figura 40	Comparativo de tamanho em arquivo.	63
Figura 41	<i>Software</i> Atube Catcher.	64
Figura 42	Demonstração da compressão da Tabela. Cenário 1.	65
Figura 43	Demonstração da compressão da Tabela 2. Cenário 2.	65
Figura 44	Tempo de envio de pacotes.	67
Figura 45	Comparação de imagem enviada por diferentes distâncias.	67
Figura 46	Estatística de transmissão coletadas pelo WireShark.	68
Figura 47	Dados coletados pelo WireShark.	68

Figura 48	Dados coletados pelo WireShark.	69
Figura 49	Dados coletados pelo WireShark.	69
Figura 50	Dados coletados pelo WireShark.	70
Figura 51	Dados coletados pelo WireShark.	70
Figura 52	Dados coletados pelo WireShark.	71
Figura 53	Comando para a criação do arquivo .hex para a gravação do MAC no EE- PROM do microcontrolador ATMEGA1281.	77
Figura 54	Execução do comando de criação do arquivo com o endereço MAC.	78
Figura 55	Formulário de seleção do dispositivo a ser gravado pelo AVR Studio.	78
Figura 56	Configuração do dispositivo conforme trabalho a ser realizado.	79
Figura 57	Definição do oscilador de frequência do dispositivo.	80
Figura 58	Configuração da Tensão do Dispositivo e Frequência do Clock.	80
Figura 59	Gravação do arquivo .hex na memória Flash do ATMEGA1281	81
Figura 60	Processo de gravação na EEPROM do ATMEGA1281.	82
Figura 61	Visão geral da aplicação utilizando o microcontrolador Atmega1281.	83
Figura 62	Tela de início da Aplicação.	84
Figura 63	Tela de configuração da COM.	84
Figura 64	Tela de configuração <i>Coordinator</i>	85
Figura 65	Tela configuração da COM.	85
Figura 66	Tela de interface da comunicação ponto a ponto utilizando o padrão IEE 802.15.4.	86
Figura 67	Tela da Interface da aplicação ponto a xonto enviando uma mensagem ao End Device.	86
Figura 68	Tela da Interface da aplicação ponto a ponto recebendo uma mensagem do Coordenador.	87
Figura 69	Tela da Interface da Aplicação End Device enviando uma mensagem para o Coordenador.	87

Figura 70	Tela da Interface da Aplicação - Coordenador recebe a mensagem do End Device.	88
-----------	---	----

LISTA DE TABELAS

Tabela 1	Comparativo dos tipos de arquivos a a compressão obtida. Cenário 1. . .	64
Tabela 2	Comparativo dos tipos de arquivos a a compressão obtida.	65

LISTA DE ABREVIACOES E SIGLAS

ACL	<i>Access Control List</i>
API's	<i>Application Program Interface.</i>
CMYK	<i>Cyan, Magent, Yellow, black.</i>
DCT	<i>Discrete Cosine Transform.</i>
DPI	<i>Dots per Inch.</i>
E/S	<i>Entrada/Saída.</i>
FFD	<i>Full Function Device.</i>
GIF	<i>Graphics Format.</i>
GNU	<i>General Public License</i>
GTS	<i>Guaranteed Time Slotsi</i>
IDE	<i>Integrated Development Environment.</i>
IEEE	<i>Institute of Electric and Electronic Engineers.</i>
ISP	<i>In-System Programming.</i>
JDK	<i>Java Development Kit.</i>
JEE	<i>Java Enterprise Edition.</i>
JME	<i>Java Micro Edition.</i>
JPEG	<i>Join Picture Expert Group.</i>
JRE	<i>Java Runtime Environment.</i>
JSE	<i>Java Standard Edition.</i>
JVM	<i>Java Virtual Machine.</i>
LED	<i>Light Emitter Diode.</i>
LZW	<i>Lempel-Ziv-Welch.</i>
MAC	<i>Medium Access Control.</i>
MHz	<i>Mega Hertz.</i>
MPDU	<i>Mac protocol Data Unit.</i>
MSDU	<i>Mac Service Data Unit.</i>
NWK	<i>NetWork.</i>
OSI	<i>Open Systems Interconnection.</i>
PC	<i>Personal Computer.</i>
PDA	<i>Personal Digital Assistant.</i>
PHY	<i>Physical Layer.</i>
PIXELS	<i>Picture Element.</i>
PAN	<i>Personal Area Networks.</i>

PPDU	<i>Physical Protocol Data Unit.</i>
RFD	<i>Reduced-Function Devices.</i>
RGB	<i>Red-Green-Blue.</i>
RS232	<i>Recommended Standard-232.</i>
Rx	<i>Receive.</i>
SDK	<i>Software Development Kit.</i>
SPI	<i>Serial Peripheral Interface.</i>
SRAM	Memória estática de acesso aleatório.
TIFF	<i>Tagged Image File Format.</i>
Tx	<i>Transmit.</i>
WLAN	<i>Wireless Local Area Network.</i>
WMAN	<i>Wireless Metropolitan Area Network.</i>
WPAN	<i>Wireless Personal Area Network.</i>
WWAN	<i>Wireless Wide Area Network.</i>
ZIF	<i>Zero Insertion Force.</i>
3GPP	<i>Third Generation Partnership Project.</i>
3G	Terceira Geração de Telefonia Móvel

SUMÁRIO

1	INTRODUÇÃO	19
1.1	OBJETIVOS	20
1.2	REVISÃO DE LITERATURA	20
1.3	ORGANIZAÇÃO DO TEXTO	21
2	IMAGENS	23
2.1	VISÃO HUMANA E VISÃO ARTIFICIAL	23
2.2	IMAGENS DIGITAIS	24
2.2.1	Imagens Vetoriais	24
2.2.2	Imagem de Bitmap	25
2.3	PROCESSAMENTO DE IMAGENS	25
2.4	HISTOGRAMA DE IMAGEM	26
2.5	RESOLUÇÃO DE UMA IMAGEM	27
2.5.1	Resolução Espacial	27
2.5.2	Resolução de Contraste	27
2.5.3	Resolução Temporal	28
2.6	USO DAS CORES E SUAS VARIAÇÕES NA MANIPULAÇÃO DAS IMAGENS	28
2.6.1	Imagem Colorida	28
2.6.2	Cores	28
2.6.3	Mistura	28
2.6.4	RGB	29
2.6.5	Adição e Subtração	29
2.6.6	Imagens Monocromáticas	30
2.6.7	Imagens em Tons de Cinza (<i>grayscale</i>)	30

2.7	<i>PIXELS</i>	31
3	COMPRESSÃO DE IMAGENS E VÍDEOS	32
3.1	CODIFICAÇÃO LZW	32
3.2	CARACTERÍSTICAS DE COMPRESSÃO E FORMATO DE ARQUIVOS DE IMAGENS E VÍDEOS	33
3.2.1	Características do JPEG	33
3.2.2	Características do BMP	33
3.2.3	Características do TIFF	34
3.2.4	Características do MPEG	34
3.2.5	Características do 3GPP	35
4	PADRÃO IEEE 802.15.4 / ZIGBEE	36
4.1	CARACTERÍSTICAS DO ZIGBEE	36
4.2	CAMADAS DO PADRÃO IEEE 802.15.4	36
4.2.1	Camada de Rede	37
<i>4.2.1.1</i>	<i>Tipos de Dispositivos</i>	<i>38</i>
<i>4.2.1.2</i>	<i>Topologias de Rede</i>	<i>38</i>
4.2.2	Camada Física	39
4.2.3	Banda utilizada no Brasil	40
4.3	CAMADA MAC	40
4.3.1	Camada de Controle de Acesso ao Meio	40
<i>4.3.1.1</i>	<i>Transferência de dados em redes com sinal ativado versus redes sem sinal</i>	<i>41</i>
<i>4.3.1.2</i>	<i>Métodos de transferência de dados</i>	<i>41</i>
<i>4.3.1.3</i>	<i>Transferência de dados para um Coordenador</i>	<i>41</i>
<i>4.3.1.4</i>	<i>Transferência de dados a partir de um coordenador</i>	<i>42</i>
4.3.2	Associação e dissociação	43

4.3.3	Segurança de uma rede ZigBee	43
4.4	ORGANIZAÇÃO DO <i>FRAME</i>	44
4.5	<i>GATEWAY</i> DE UMA REDE ZIGBEE	44
4.6	EXEMPLOS DE APLICAÇÕES UTILIZANDO O PADRÃO ZIGBEE	44
5	FERRAMENTAS UTILIZADAS E METODOLOGIA DE DESENVOLVIMENTO	46
5.1	LINGUAGEM JAVA	46
5.2	FERRAMENTA DE DESENVOLVIMENTO JAVA - NETBEANS	47
5.3	IDE E SIMULADOR AVR STUDIO	48
5.4	COMPILADOR LINGUAGEM DE PROGRAMAÇÃO C - WINAVR	49
5.5	COMPRESSÃO DE VÍDEO - <i>SOFTWARE ATUBE CATCH</i>	50
5.6	ANÁLISE DE TRÁFEGO WIRESHARK	51
5.7	MÓDULOS UTILIZADOS PARA O DESENVOLVIMENTO DA REDE ZIGBEE	52
5.7.1	MÓDULO STK500	52
5.7.2	Módulo STK501	53
5.7.3	Antena de comunicação RZ502	54
5.8	CONEXÃO DOS MÓDULOS UTILIZADOS	55
5.9	TRANSCEPTOR RZ RAVEN	56
5.10	COMUNICAÇÃO SEM FIO (IEEE 802.15.4) COM IMAGENS E VÍDEOS	56
5.11	DESENVOLVIMENTOS DE UM SOFTWARE COM INTERFACE GRÁFICA PARA COMPRESSÃO DE IMAGENS.	58
6	RESULTADOS E DISCUSSÕES	61
6.1	TESTES DE COMPRESSÃO DE IMAGENS BITMAP INTERFACE GRÁFICA	61
6.2	COMPRESSÃO DE VÍDEOS UTILIZANDO O SOFTWARE ATUBE CATCHER	64
6.3	COMUNICAÇÃO SEM FIO (IEEE 802.15.4) COM IMAGENS E VÍDEOS	66
7	CONCLUSÕES	72

REFERÊNCIAS	74
APÊNDICE A - Criação do Arquivo de Gravação do MAC na EEPROM do ATMEGA1281	77
A.1 CONFIGURAÇÃO DO AMBIENTE AVR STUDIO PARA A GRAVAÇÃO COM ARQUIVOS NO MICROCONTROLADOR	78
APÊNDICE B - CONFIGURAÇÃO DA COMUNICAÇÃO SEM FIO (IEEE 802.15.4) VIA TEXTO	83

1 INTRODUÇÃO

A rede sem fio refere-se a uma rede de comunicação sem a necessidade do uso de cabos e são classificadas de acordo com sua abrangência, ou seja, redes pessoais ou de curta distância, sendo WPAN (*Wireless Personal Area Network*), redes locais WLAN (*Wireless Local Area Network*), redes metropolitanas WMAN (*Wireless Metropolitan Area Network*) e redes geograficamente distribuídas ou de longa distância WWAN (*Wireless Wide Area Network*) (KUROSE et al., 2010).

As demandas por soluções de comunicações de redes sem fio (tecnologia *wireless*) estão integradas ao cotidiano e cada dia mais inseridas e valorizadas em nosso meio. As redes de sensores sem fio abrangem uma vasta gama de aplicações, tais como controle de dispositivos de localização, de periféricos de computadores, de sensores médicos, de brinquedos, de jogos, de sensor de umidade, de sensor de temperatura, de sensor velocidade do vento, de sensor de direção do vento, de sensor de pressão atmosférica, de iluminação, de aquecimento, de ventilação, de irrigação, de alarmes, de cancelas, de portas e portões bem como em aplicações de monitoramento remoto e em aplicações automotivas (GISLASON, 2008).

As redes sem fios possui as seguintes vantagens:

- **Mobilidade:** informação em tempo real em qualquer lugar da organização ou empresa para todo usuário de rede;
- **Facilidade de Instalação:** diminui a utilização de cabos elétricos nos prédios em que serão instalados;
- **Flexibilidade:** permite chegar onde o cabo não pode (lugares hostis);
- **Redução de Custos:** quando ocorrem mudanças frequentes ou o ambiente é mutável, o custo inicialmente alto da rede sem cabos torna-se mais baixo, além de ter maior tempo de duração e menor gasto na instalação;
- **Escalabilidade:** a mudança de topologia de rede é simples e afeta da mesma maneira pequenas e grandes redes, permitindo ampliar facilmente as mesmas.

O Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) lançou no ano de 2003 o padrão 802.15.4 para facilitar a comunicação sem fio, confiável, com baixo consumo de energia e baixas taxas de transmissão (HOWITT; GUTIERREZ et al., 2003). O padrão 802.15.4 é conhecido como ZigBee e sua área de abrangência corresponde as WPAN ou rede pessoal sem fio.

Normalmente utilizada para interligar dispositivos eletrônicos fisicamente próximos, sendo que a sua distância pode variar dependendo do padrão de comunicação utilizado (LABIOD; AFIFI; SANTIS, 2007). A rede é mais utilizada para eliminar a utilização de cabos para interligar equipamentos como, por exemplo, impressoras, mouses, teclados e outros.

O nome do padrão utilizado nesse trabalho surgiu devido ao voo em ziguezague das abelhas, que desta forma indicam aos membros da sua comunidade a distância, direção e localização dos alimentos descobertos, que surgiu por analogia, o nome deste padrão, como sendo ZigBee (FARAHANI, 2011).

Com grandes recursos computacionais disponíveis, processamento de imagens com altas resoluções demandam um elevado poder computacional, ocasionando assim o encarecimento de computadores para processarem este tipo de informação. Em algumas aplicações, como na WWW (*World Wide Web*), a utilização de imagens com elevadas taxas de resolução muitas vezes não é utilizada da forma correta, imagens de formato incorreto podem aumentar significativamente o tamanho do arquivo, tornando o sistema inviável para o tipo de aplicação de transmissão de imagens. É com estas considerações que são utilizados e constantemente criados e aperfeiçoados métodos de compressão de imagens para diminuir o tempo de transmissão, ou seja, diminuir o tráfego na rede, lembrando-se que a taxa de transmissão do padrão IEEE 802.15.4 é 250 kbps. A compressão de imagens e vídeo está se tornando métodos essenciais para o monitoramento de ambientes (plantações, residências, aplicações hospitalares, automóveis, aviões, etc.). Assim, este trabalho procura evidenciar o tempo de comunicação na transmissão de imagens e vídeos compactados verificando a viabilidade de implantação do sistema proposto.

1.1 OBJETIVOS

O presente trabalho tem como objetivo analisar e mensurar as limitações da transferência de imagens e vídeos através de uma rede sem fio com padrão IEEE 802.15.4. Pretende-se efetuar medições quanto à influência da distância na taxa de transmissão e nas perdas de pacotes transmitidos.

1.2 REVISÃO DE LITERATURA

Nessa seção, é apresentada a revisão bibliográfica sobre os trabalhos relacionados com o padrão IEEE 802.15.4 para transmissão de imagens e vídeo. No trabalho de Pekhteryev et al. (2005), apresentou-se uma proposta de testes de transmissão de imagens JPEG e JPEG-2000 através de uma rede ZigBee.

Segundo Zainaldin, Lambadaris e Nandy (2008) o padrão IEEE 802.15.4 (ZigBee) é pro-

jetado para baixa taxa de dados, baixo consumo de energia e baixo custo redes de sem fio. A transmissão de vídeo utilizando o padrão é considerado um problema pelo alto tráfego de dados. O trabalho dos autores apresenta um estudo sobre a melhoria de desempenho nas redes ZigBee e interferência de dispositivos que utilizam múltiplos canais para transmissão.

Sanchez e Haro (2010), apresentam um ambiente de simulação no NS-2 para analisar a viabilidade e comportamento da transmissão de vídeo MPEG-4 no padrão IEEE 802.15.4.

As redes de sensores sem fio destacam-se por possuírem baixo custo, serem confiáveis e possuírem taxas de transferência de dados aceitáveis para o controle de transdutores inteligentes. A quantidade de dados a serem transmitidos através da rede é mencionada como um grande problema pela sua limitação física (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000).

O estudo para a transferência e a compactação de imagens a cada dia se torna uma necessidade sendo utilizada por mais e mais usuários para conseguir armazenar uma quantidade maior de dados em pequenos espaços de armazenamento como, por exemplo, em dispositivos móveis. A utilização de algoritmos de compressão de áudio e vídeo pode ser utilizada no gerenciamento e controle de redes de sensores sem fio através do padrão IEEE 802.15.4. Apesar da limitação de baixas taxas de transferência de dados, com a utilização de técnicas de compressão de vídeos é possível executar uma transmissão com baixo custo e confiável. Elas enfatizam a necessidade de pesquisas sobre os problemas de comunicação multimídia em redes de sensores (GURSES; AKAN, 2005).

Irgan, Ünsalan e Baydere (2014) apresentaram um estudo sobre uma rede de sensoriamento de ambiente utilizando o padrão ZigBee, priorização de baixo custo de blocos de imagem em redes de sensores sem fio para a vigilância das fronteiras. Para estas aplicações, câmeras conectadas em diversos nós sensores são colocados em locais pré-selecionados na região de fronteira. No caso de um evento de intrusão, essas câmeras capturam imagens e transmitem aos nós da rede, no projeto foram utilizados arquivos JPEG com simulações no software MATLAB.

Mustafa et al. (2010), apresentaram um algoritmo para comparar os principais formatos de arquivos utilizados para transmitir recursos multimídia como o MPEG-4, 3GP, FLV no qual foi realizado um comparativo em diversos experimentos com o desempenho de cada formato em transmissões de redes sem fio. Em relação a este trabalho, esse artigo ajudou na escolha dos formatos de vídeos a serem testado no projeto.

1.3 ORGANIZAÇÃO DO TEXTO

O texto está estruturado em seis capítulos, além desta seção de introdução, organizados da seguinte forma:

- **Capítulo 2** - São apresentados conceitos de imagens e vídeos juntamente com suas características, conhecimentos necessários para adquirir e processar uma imagem / vídeo e conceitos de compressões de imagens e vídeo juntamente com suas aplicabilidades.
- **Capítulo 3** - São apresentados os conceitos de compressões de imagens, vídeos e formato de arquivos.
- **Capítulo 4** - Apresenta as principais características do padrão 802.15.4, conhecido como ZigBee, sua camada de rede, tipos de dispositivos, topologias de rede, camada física, tipos de comunicações utilizadas, segurança e vantagens.
- **Capítulo 5** - Descreva as linguagens utilizadas no desenvolvimento neste projeto: Java, C, IDE's (*Integrated Development Environment*) AVR Studio, NetBeans e o software livre *Atube Catcher* para compressão de vídeos, além das ferramentas da Atmel Stk 500, Stk 501 e rádio transceptor RZ 502.
- **Capítulo 6** - Relata a análise de testes e resultados, além da demonstração da interface gráfica adaptada para compressão de imagens. Descreve também o software livre *Atube Catcher* para compressão de vídeos.
- **Capítulo 7** - É apresentada a conclusão do trabalho e sugestões para trabalhos futuros.

2 IMAGENS

Neste capítulo serão definidos os paradigmas de como são tratadas as imagens pelo olho humano, além de ser estudados aspectos de qualidade das imagens, resolução, histograma e tipos de cores. O estudo realizado baseia-se nas referências (GONZALEZ; WOODS, 2000) e (FILHO; NETO, 1999).

2.1 VISÃO HUMANA E VISÃO ARTIFICIAL

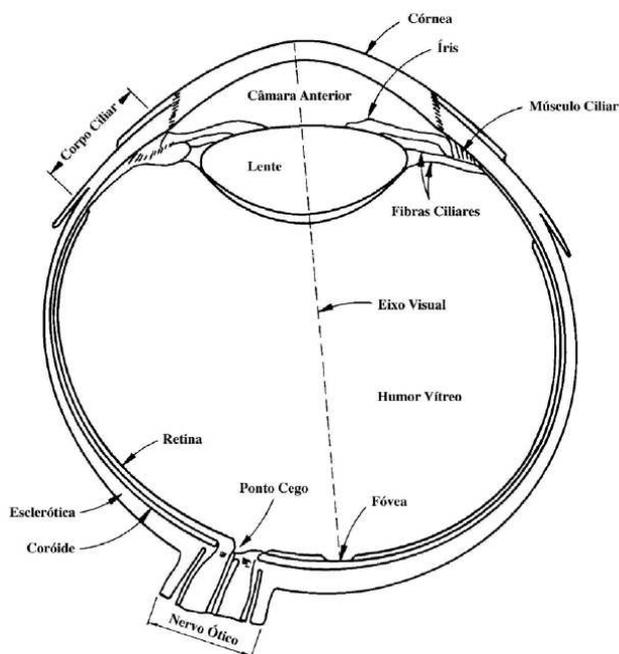
Um sistema geral de visão eficiente é o sistema visual humano. Este sistema é muito desenvolvido, apresentando grande complexidade. Mesmo com os avanços tecnológicos atuais, que permite aos computadores uma grande eficiência no que se refere, por exemplo, a realização de operações de cálculo, eles não realizam tarefas consideradas simples para seres humanos como, por exemplo, a capacidade de se adaptar a diferentes tarefas, condições de trabalho e de interpretação subjetiva de cores. Na Figura 1 representa-se uma vista em corte do olho humano, destacando o cristalino, a retina, o nervo óptico, a íris e a córnea.

Cristalino ou lente orgânica é uma parte do olho humano que se localiza do lado frontal, a qual contém informações de cor e de intensidade. Através do nervo óptico, milhões de células sensíveis a luz são estimuladas através da imagem e em seguida fazem a transmissão dos sinais nervosos diretamente ao cérebro, onde os sinais são traduzidos como sensações que incluem forma, cor e brilho.

Um sistema de visão artificial é como um sistema computadorizado capaz de adquirir, processar e interpretar imagens correspondentes a cenas reais.

Existem grandes diferenças entre o sistema visual humano e um sistema de visão artificial, conforme comparações entre espectro, flexibilidade, habilidade, cor, sensibilidade, tempo de resposta, 2D, 3D, e percepção.

Figura 1 - Visão humana



Fonte: Adaptado de (GONZALEZ; WOODS, 2000).

2.2 IMAGENS DIGITAIS

Cada tipo de imagem tem um valor, branco e preto, com valores de intensidade, e três valores associados: vermelho, verde e azul. Portanto, imagem é uma distribuição das propriedades físicas de pontos em um plano cartesiano.

Quando uma imagem é convertida para digital, ela pode ser alterada, armazenada e visualizada em um computador ou em equipamentos como celulares, televisores e *ipads*, podendo ser visualizadas a qualquer instante.

2.2.1 Imagens Vetoriais

As imagens vetoriais são utilizadas na área gráfica, como no desenvolvimento de logotipos de empresas, por desenhistas e nas engenharias civil e mecânica desenhando-se elementos gráficos por linhas, conforme apresentado na Figura 2. Embora muito adequada para representar um projeto arquitetônico, ela não é apropriada para representar uma foto digitalizada.

Figura 2 - Exemplo de imagem vetorial.



Fonte: (PAULISTA, 2014)

2.2.2 Imagem de Bitmap

Imagens de pintura, que são constituídas por pontos individuais são chamadas imagens *bitmap*. Cada pixel é colorido individualmente, sendo possível criar vários efeitos, mas suas partes não podem ser movidas individualmente.

Os *bitmaps* são conjunto de *pixels* da imagem digital, onde o valor de cada pixel representa as suas características de luminosidade e cor.

As imagens de *bitmap* são usadas nos aplicativos para a edição de fotografias, onde é possível se aperfeiçoar pequenos detalhes, fazer alterações radicais ou intensificar efeitos. Na Figura 3 exemplifica-se um tipo de imagem *bitmap*, onde é possível perceber a grande quantidade de cores presentes na imagem.

Figura 3 - Exemplo de imagem *bitmap*.



Fonte: (PMISA, 2013)

2.3 PROCESSAMENTO DE IMAGENS

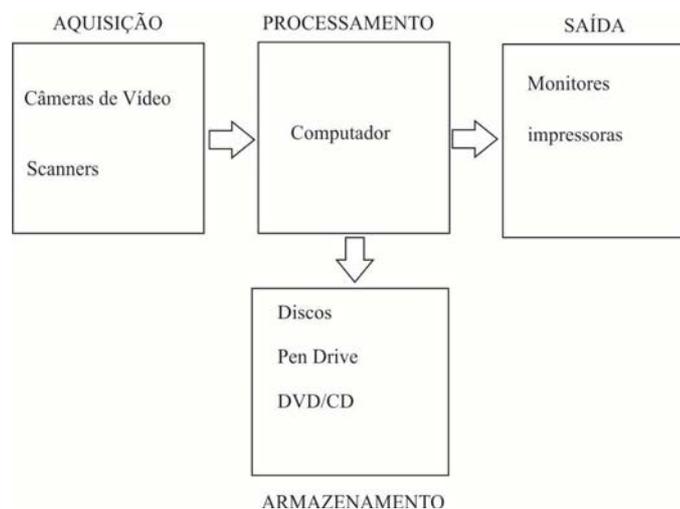
Uma imagem digital pode ser considerada como sendo uma matriz cujos índices de linhas e de colunas identificam um ponto na imagem, e o correspondente valor do elemento da matriz identifica o nível de cor naquele ponto. Processamento de imagens refere-se à manipulação de

uma imagem realizada por um computador, onde a entrada e a saída do processo resultam em imagens.

A área de processamento de imagens possui destaque em várias áreas de pesquisa, como por exemplo, a transmissão de sinais de televisão, análise de imagens, análise de solos através de imagens fornecidas por satélites.

As imagens podem ser adquiridas através de câmeras de vídeo, *scanners* de mesa, tomógrafos e satélites. Estas imagens podem ser armazenadas em discos rígidos ou em outros dispositivos de armazenamento, conforme é ilustrado na Figura 4. A imagem armazenada pode ser processada para cumprir os mais variados objetivos: compressão, melhoramento e reconhecimento de padrões. A imagem adquirida ou processada pode ser visualizada em diversos dispositivos de saída gráfica: filmes fotográficos, impressoras ou monitores de vídeo.

Figura 4 - Elementos de um sistema de processamento de imagens.



Fonte: (FILHO; NETO, 1999)

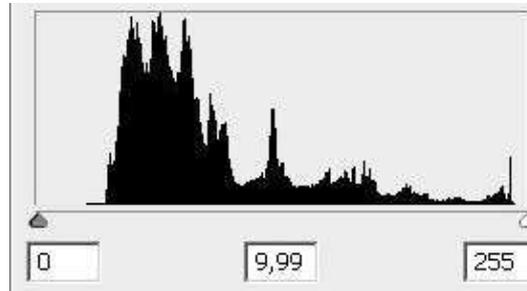
2.4 HISTOGRAMA DE IMAGEM

O histograma de uma imagem é simplesmente um conjunto de números indicando o percentual de *pixels* naquela imagem que apresentam um determinado nível de cinza. Estes valores são representados em um gráfico que informa para cada nível de cinza a quantidade de *pixels* presentes na imagem.

A visualização do histograma de uma imagem permite perceber a qualidade referente ao nível de contraste e brilho médio (caso a imagem seja predominantemente clara ou escura). Além disso, o histograma de uma imagem, distribui os níveis de cinza que cada imagem possui, cujos valores estão compreendidos entre 0 (preto) e 255 (branco). Na Figura 5 apresenta-se um histograma em que o eixo X corresponde aos níveis de cinza e o eixo Y corresponde à

frequência da imagem.

Figura 5 - Representação de um histograma de imagem.



Fonte: (ADOBE, 2014)

O histograma realiza a distribuição de maneira estatística dos níveis de cinza, representados pela quantidade de *pixels* em cada nível e esta distribuição também pode ser realizada através do percentual da quantidade total de *pixels* da imagem.

2.5 RESOLUÇÃO DE UMA IMAGEM

Uma imagem é caracterizada através da utilização de parâmetros de qualidade de imagem. Os parâmetros mais úteis na qualidade de imagem são a resolução espacial, a resolução de contraste e a resolução temporal.

2.5.1 Resolução Espacial

A resolução espacial está relacionada à aspereza da imagem, ou seja, a capacidade de exibir separação física entre dois *pixels* presentes em uma imagem.

Em uma imagem digital, a resolução espacial é definida pelo número de *pixels* por área de imagem.

2.5.2 Resolução de Contraste

Resolução de contraste refere-se à distinção de diferenças referentes à intensidade, ou seja, a quantidade de níveis de cinza, como por exemplo a atenuação de raios X. Em imagens digitais, a quantidade de bits por pixel define a resolução de contraste da imagem.

2.5.3 Resolução Temporal

A resolução temporal refere-se ao tempo gasto para a formação de uma imagem, ou seja a frequência de quadros por segundo. Aplicações em tempo real, por exemplo, devem ter um processamento de imagens que gere, no mínimo, 30 imagens por segundo.

2.6 USO DAS CORES E SUAS VARIAÇÕES NA MANIPULAÇÃO DAS IMAGENS

2.6.1 Imagem Colorida

A imagem colorida é constituída por três cores primárias: vermelho, verde e azul. Estas três cores possuem intensidade variando, geralmente, do 0 (zero) ao 255, permitindo a codificação de aproximadamente 17 milhões de cores diferentes.

A partir do momento em que tem-se sensores para a luz (os olhos), é natural que os processos internos tratem essas imagens, resultantes dessa percepção. Já os computadores possuem alguns métodos usados para variar e manipular as imagens e suas cores.

2.6.2 Cores

Cores são as diferenças no comprimento de ondas do sinal eletromagnético de alta frequência. A luz ou ondas eletromagnéticas são conhecidas como fótons (fluxo de partículas energéticas sem massa), refratadas em diversos comprimentos, que variam pelo encontro com outras partículas, oxigênio, gás carbônico entre outros.

De acordo com o comprimento da onda, as sensações de cores são produzidas pela luz. Escolhendo três cores básicas, azul, verde e vermelho, nota-se, que a propriedade mais importante que possuem é a de serem misturadas, para produzirem novas cores.

2.6.3 Mistura

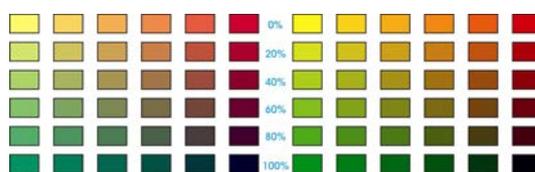
Existe uma teoria, sobre a percepção cromática, que baseia-se em uma hipótese formulada por Young em 1801 (GONZALEZ; WOODS, 2000). Para Young (1801) "células fotossensíveis que compõem a retina, juntamente com os bastonetes, se subdividem em três classes, existindo apenas diferença de sensibilidade em tom de vermelho".

Portanto, para a cor ser vista pelo olho humano, é necessário que o olho, seja atingido por uma energia eletromagnética. Então, a cor que se vê no objeto é verificada através da luz refletida por este objeto.

Na Figura 6 apresenta-se de forma resumida a especificação de mistura de cores RGB,

apresentando o percentual de cores RGB, o que resulta em várias cores secundárias. Na seção 2.6.4 apresenta-se de forma resumida a especificação da mistura de cores RGB.

Figura 6 - Exemplo de mistura de cores RGB.

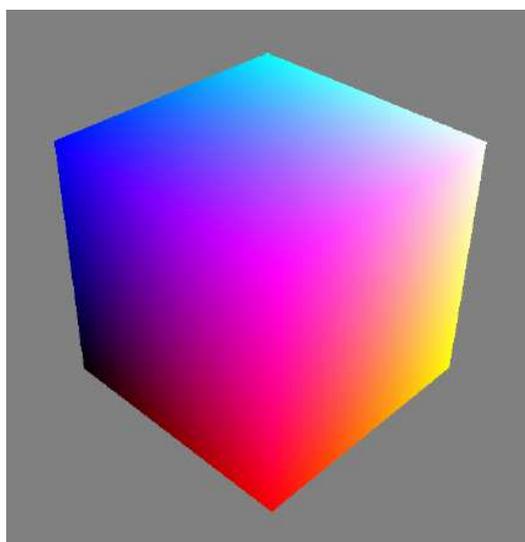


Fonte: Elaborado pelo autor.

2.6.4 RGB

Na digitalização, em 32 bits, os canais: ciano, magenta, amarelo e preto, fazem a leitura original, para que a cor na hora da impressão seja sem compactação. Na Figura 7, demonstra-se a aplicação de cores RGB, onde a mesma está em forma de um cubo. Os *scanners* conseguem capturar imagens de 24 bits, sendo 8 bits para cada cor, com 256 níveis de brilho pra expressar a presença relativa de cor, possibilitando a representação de 16,7 milhões de cores diferentes em cada imagem.

Figura 7 - Cubo RGB.



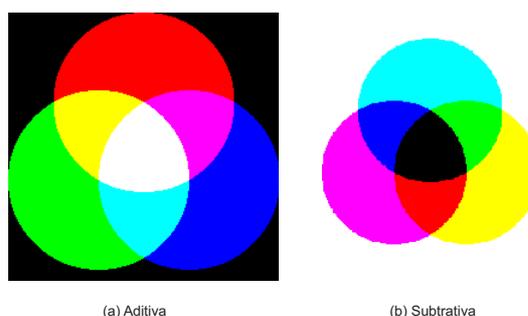
Fonte: Elaborado pelo autor.

2.6.5 Adição e Subtração

As cores primárias (vermelho, azul e verde), são aditivas, pois, pode-se obter outras a partir da combinação de qualquer uma, formando as cores secundárias. As misturas das três cores

primárias, ou de uma secundária, com sua cor primária contrária, produz "luz branca". Estas combinações, são chamadas de subtrativas, pois subtrai-se parte da luz branca, refletindo apenas a cor que corresponde ao pigmento. Na Figura 8 apresenta-se dois exemplos de mistura de cores RGB.

Figura 8 - Exemplo de mistura de cores RGB.



Fonte: Elaborado pelo autor.

2.6.6 Imagens Monocromáticas

A principal característica de imagens monocromáticas é o fato delas não apresentarem cores em sua composição, mas apresentarem tons de cinza. E os valores considerados intermediários, referem-se aos tons de cinza.

2.6.7 Imagens em Tons de Cinza (*grayscale*)

Nas imagens em tons de cinza, os *pixels* apresentam valores compreendidos na escala de cinza. É adotada uma escala contendo 256 níveis de cinza, sendo que o valor 0 (zero) indica a cor preta, enquanto que o valor 255 refere-se à cor branca. Outros tons na cor cinza, estão compreendidos nesses dois valores.

As imagens em tons de cinza, também são denominadas imagens *grayscale* (escala de cor cinza) conforme mostrado na Figura 9. E pelo tipo de armazenamento no computador, elas também podem ser conhecidas por imagens com 8 bits, cujo exemplo prático pode-se citar as radiografias.

Figura 9 - Imagem em tons de cinza.



Fonte: (PMISA, 2013).

2.7 PIXELS

Pixels são elementos de figuras que possuem dispositivos coloridos de maneiras diferentes, para formar um padrão. É um elemento com dimensões finitas, possui uma forma retangular ou quadrada. O número de *pixels* em uma imagem vai variar, de acordo com o tamanho e resolução das imagens.

3 COMPRESSÃO DE IMAGENS E VÍDEOS

Neste capítulo, apresentam-se diferentes métodos de compressões de imagens e vídeo, dentro da compressão sem perda de qualidade e com perda de qualidade, baseando-se nos autores Filho e Neto (1999).

A compressão de imagens e vídeo tem vantagens importantes. Torna possível a utilização de vídeo digital em transmissão e armazenamento em ambientes realmente onde dificilmente trafegaria um vídeo.

Na compressão de imagens digitais, três redundâncias básicas de dados podem ser identificadas e exploradas: redundância de codificação, redundância interpixel, e redundância psicovisual. A compressão de dados é efetivamente obtida quando uma ou mais dessas redundâncias são reduzidas ou eliminadas.

Redundância de codificação utiliza o melhoramento da qualidade da imagem através de modificações de seu histograma.

Na redundância interpixel a maior parte da contribuição visual de um simples pixel para uma imagem é redundante; ela poderia ter sido predita com base dos valores de seus vizinhos.

A redundância psicovisual comprova que o olho humano não consegue captar todas as informações visual o qual recebe.

Embora existam alguns trabalhos na literatura dedicados a compressão de imagem para a utilização de banda em redes de sensores sem fio, estas são ainda em seus estágios iniciais como o trabalho de Lecuire, Duran-Faundez e Krommenacker (2007).

3.1 CODIFICAÇÃO LZW

O método de codificação LZW (*Lempel-Ziv-Welch*) baseia-se na construção progressiva de uma tabela (dicionário) contendo as *strings* de símbolos de vários comprimentos encontradas na sequência de símbolos da fonte. Como este dicionário é criado a partir da sequência a ser codificada, ele reflete com precisão as estatísticas daquela fonte de informação sendo codificada (NISHAD; CHEZIAN, 2014).

Entre os métodos de compressão de imagens mais importantes como por exemplo PNG, TIFF, destacam-se, o JPEG (*Joint Photographic Experts Group*), cuja origem é da composição de várias técnicas. É muito usado, pela maioria dos profissionais, da área de design gráfico e

web devido a compressão da imagem.

3.2 CARACTERÍSTICAS DE COMPRESSÃO E FORMATO DE ARQUIVOS DE IMAGENS E VÍDEOS

Na compressão com perdas é importante conhecer o sistema de visão, para descartar-se as informações não relevantes, o qual será especificado abaixo.

3.2.1 Características do JPEG

JPEG é uma técnica de compressão com perdas, baseado na DCT (*Discrete Cosine Transform*). O algoritmo trabalha sobre o limite do olho humano, principalmente as variações das cores menos perceptíveis com as variações de brilho. O JPEG tem quatro modos de operação:

- Da esquerda para a direita, de alto a baixo, a imagem é codificada em uma única etapa;
- Progressiva: a qualidade e resolução aumentam a cada etapa em que a imagem é codificada;
- Hierárquica: em várias resoluções a imagem é multiplicada e codificada;
- Sem perda: varia de acordo com a compactação da imagem.

Nota-se que as técnicas progressiva e hierárquica permitem ao usuário selecionar um nível de qualidade variável para a imagem. Utiliza-se uma imagem de menos qualidade quando a taxa de transmissão é lenta.

3.2.2 Características do BMP

Tem propriedades de compressão sem perdas. É um formato de arquivo criado pela Microsoft, baseado no mapa de bits presente no Windows. É o formato padrão de aplicações, presentes no Windows, responsáveis pela manipulação de imagens.

Projetado para ser utilizado em sistemas operacionais que sejam compatíveis com a plataforma Windows, o formato BMP não suporta outras arquiteturas. Deste modo, arquivos BMP não estão presentes em aplicações para internet, pelo fato de não ser um padrão aberto, universal e devido ao tamanho do arquivo.

3.2.3 Características do TIFF

Criado no ano de 1986 pelas empresas Aldus e pela Microsoft, o formato TIFF é direcionado para arquivos de maior tamanho e que apresentem alta qualidade, onde é possível ter-se uma imagem de qualidade alta ou baixa, cujo propósito é gerar arquivos com ausência de compressão. Esta ausência permite evitar problemas de compatibilidade nos sistemas.

Um arquivo de extensão TIFF pode ser de 8 ou de 16 bits por cada canal de cor RGB, permitindo ao usuário um ganho de 48 bits para cada pixel em relação a uma quantidade de bilhões de cores.

Utilizar 48 bits gera uma desvantagem, ou seja, o arquivo gerado pode chegar a dobrar seu tamanho. Quando uma imagem é processada várias vezes, é indicado utilizar o modo 16 bits para evitar perdas. Quando uma imagem em formato TIFF é convertida para JPG, a sua qualidade cai de 16 para 8 bits, sendo impossível reverter esse processo de perda de qualidade.

3.2.4 Características do MPEG

O comitê MPEG (*Moving Pictures Experts Group*) foi estabelecido pelo ISO e o IEC em 1988 com o objetivo de desenvolver padrões de codificação de vídeo e áudio associado para armazenamento em mídia digital. Desenvolvido para comprimir mais ou menos 30 minutos de áudio e vídeo em um CD. Com resolução de 320 *pixels* por 240 *pixels*.

O formato MPEG-2 foi publicado como padrão em 1994 e foi criado justamente para suprir algumas necessidades que o MPEG-1 não atendia no desenvolvimento de vídeo digital. Com resolução de 720 *pixels* por 480 *pixels*.

O formato MPEG-4 foi finalizado em 1998 e surgiu por inúmeras necessidades distintas e complexas. Formatos como o MPEG-4 determina codificar e decodificar sons e imagens, que em seu formato original ocupariam uma enorme quantidade de espaço e recursos para transmissão pela internet.

O MPEG-4 juntamente com o H.264 são padrões para a codificação e representação da informação visual. Cada norma visa assegurar que os codificadores e decodificadores compatível com êxito podem interagir um com o outro, permitindo simultaneamente aos fabricantes a liberdade de desenvolver produtos competitivos e inovadores.

O formato MPEG-7 está em estudos ainda e é também chamado de *Multimedia Content Description Interface*. O padrão especificará uma descrição padronizada dos vários tipos de informação multimídia (imagens estáticas, gráficos, áudio, vídeo, e informação sobre como estes elementos são combinados numa apresentação). Esta descrição será associada ao conteúdo multimídia para permitir acesso rápido e eficiente ao material e poderá ser feita em vários níveis

semânticos.

3.2.5 Características do 3GPP

É um projeto de colaboração entre os grupos de associações de telecomunicações definida pela *Third Generation Partnership Project* (THIRD... , 2015), para desenvolver um formato de arquivo de vídeo de terceira geração (3G) para sistemas de telefonia celular o qual possuem uma capacidade limitada de memória (MUSTAFA et al., 2010). Embora o formato seja derivado do MPEG-4, o 3GP não possui a mesma qualidade de vídeo, várias marcas de celulares mundiais estão definindo o formato como padrão para *smartphones* com câmeras.

4 PADRÃO IEEE 802.15.4 / ZIGBEE

Neste capítulo, apresentam-se conceitos do padrão IEEE 802.15.4, suas características, incluindo suas topologias de rede, camadas, critérios de segurança e utilização.

4.1 CARACTERÍSTICAS DO ZIGBEE

O termo ZigBee é um padrão que define um conjunto de comunicação sem fio com baixa taxa de dados e com curto alcance. Este padrão é baseado em dispositivos com comunicação sem fio que operam em bandas de 868 MHz, 915 MHz utilizados em Países da Europa e a banda utilizada no Brasil de 2,4 GHz. A taxa máxima de transferência de dados e comunicação sem fio deste padrão é de 250 kbits por segundo. A utilização do padrão IEEE 802.15.4 é recomendada para aplicações que não necessitam de alto desempenho na comunicação sem fio (taxa de dados) e que necessitam de otimização do tempo de duração das baterias, umas das suas principais vantagens. WPAN's são redes que focalizam ao redor de um indivíduo (SWAMINATHA; ELDEN, 2002). Nas Redes WPAN existentes pode-se utilizar o padrão ZigBee IEEE 802.15.4 (MESSIAS, 2009). Em muitos casos de aplicações, o tempo total do dispositivo é limitado, ele passa a maior parte do tempo em modo de espera economizando energia, também conhecido como hibernação (FARAHANI, 2011).

O padrão ZigBee foi desenvolvido pela ZigBee Alliance, que tem centenas de empresas associadas, desde a indústria de semicondutores, desenvolvedores de software, fabricantes de equipamentos e instaladores. A ZigBee Alliance foi constituída em 2002 como uma organização sem fins lucrativos, aberta a todas empresas que queiram participar do projeto (ALLIANCE, 2013).

De acordo com Jucá, Pereira e Vasconcelos (2012), nenhuma das tecnologias *wireless* anteriores ao ZigBee foi dedicada a operar com taxas de bits típicas para dispositivos simples como sensores, lâmpadas e outros equipamentos para automação industrial e residencial. Em suas pesquisas descrevem o sistema de atualização sem fio do *firmware* dos microcontroladores utilizando o padrão ZigBee ao qual pode ser utilizado em monitoramento remoto com rede de sensores sem fio como, por exemplo, em sistemas descentralizados de geração de energia, em controle de acesso de laboratórios ou em robótica educacional.

4.2 CAMADAS DO PADRÃO IEEE 802.15.4

As camadas do padrão IEEE 802.15.4 são divididas em camada MAC e camada física (FARAHANI, 2008).

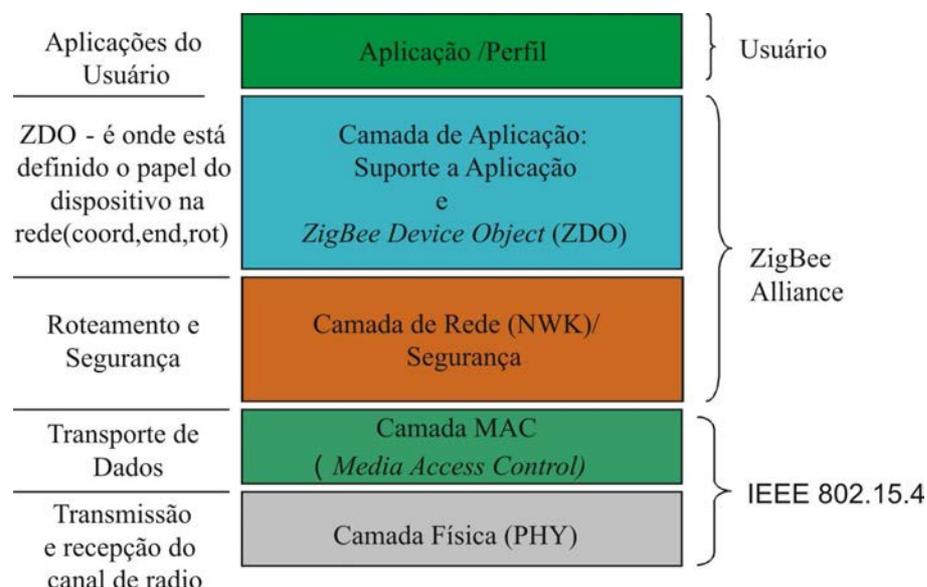
4.2.1 Camada de Rede

Uma das formas mais comuns de estabelecer uma rede de comunicação (com ou sem fio) é utilizar o conceito de camadas de rede. Cada camada é responsável por determinadas funções na rede. As camadas de padrão do IEEE 802.15.4 ZigBee são baseados no modelo de referência OSI (*Open System Interconnect*) (FARAHANI, 2011).

Dividir um padrão de rede em camadas tem uma série de vantagens. Por exemplo, se o padrão sofrer mudanças ao longo do tempo, é mais fácil de substituir ou modificar a camada que é afetada, em vez de substituir todo o padrão. Além disso, no desenvolvimento de um aplicativo, as camadas inferiores do padrão são independentes da aplicação (FARAHANI, 2011).

Como mostrado na Figura 10, na parte inferior, duas camadas de rede são definidas pelo padrão IEEE 802.15.4 (GREBENNIKOV, 2005). Esta norma é desenvolvida pelo comitê de padrões IEEE 802 e foi inicialmente lançado em 2003. O padrão ZigBee define apenas a rede, aplicações e camadas de segurança e adota o padrão IEEE 802.15.4 para camadas PHY e MAC (*Medium Access Control*). Portanto, qualquer dispositivo compatível com ZigBee tem conformidade com IEEE 802.15.4. (FARAHANI, 2011).

Figura 10 - Camadas do padrão de uma rede sem fio Zigbee.



Fonte: (ALLIANCE, 2013).

As características do nível físico da rede são determinadas pela camada PHY, portanto, parâmetros como a frequência de operação, a taxa de dados e os tipos de dispositivos são especificados no padrão IEEE 802.15.4 (FARAHANI, 2011).

4.2.1.1 Tipos de Dispositivos

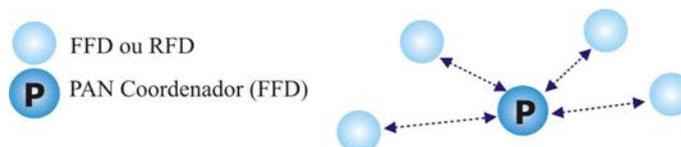
Existem dois tipos de nós que podem se comunicar: Dispositivo de Função Completa - *Full Function Device* (FFD) e Dispositivo de Função Reduzida - *Reduced Function Device* (RFD). Um FFD pode funcionar como um coordenador PAN (*Personal Area Network*) ou como um dispositivo, e comunicar com outros RFDs e FFDs. Um RFD só pode "comunicar-se" com um FFD e se destina a ser utilizado em um ponto terminal, tal como um sensor ou atuador. O dispositivo adaptável permite a construção de softwares pequenos e menos complexos, resultando em menor consumo de energia e de custos reduzidos nas variantes (FARAHANI, 2011; KUBAN, 2006).

4.2.1.2 Topologias de Rede

A topologia de rede é administrada pela camada de rede ZigBee. A rede deve ser de duas topologias especificadas no IEEE 802.15.4: a topologia estrela e ponto a ponto (FARAHANI, 2011).

Na topologia estrela, mostrado na Figura 12, cada dispositivo na rede pode se comunicar apenas com o coordenador do PAN. O dispositivo FFD, programado para ser um coordenador do PAN, é ativado e começa a definir a sua rede. A primeira coisa que este coordenador PAN faz é garantir que o identificador do PAN não seja usado por qualquer outra rede nas proximidades (FARAHANI, 2011).

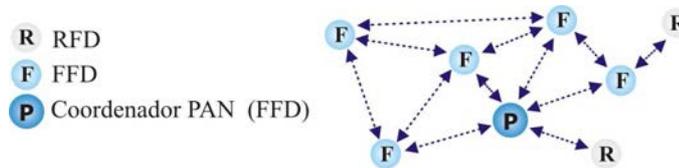
Figura 11 - Topologia estrela no padrão ZigBee.



Fonte: (ALLIANCE, 2013)

A topologia ponto a ponto, pode assumir diferentes formas, definindo as restrições sobre os dispositivos que podem se comunicar uns com os outros, conforme demonstrado na Figura 12. Se não houver restrição, esta é conhecida como uma topologia em malha (*mesh*).

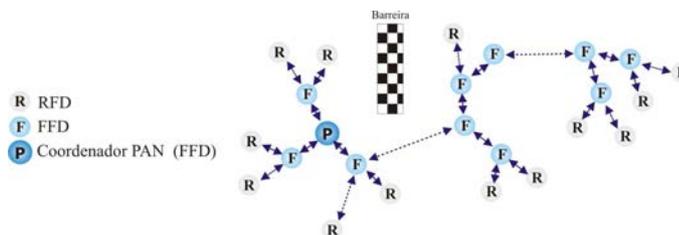
Figura 12 - Topologia em malha (mesh) no padrão ZigBee.



Fonte: (ALLIANCE, 2013)

Neste caso, o coordenador PAN das redes sem fio ZigBee inicia a rede. Roteadores ZigBee formam os galhos e retransmitem as mensagens. Os dispositivos *end Device's* irão agir como as folhas da árvore e não participam no encaminhamento da mensagem. Os roteadores ZigBee podem expandir a rede para além da rede inicial estabelecido pelo coordenador ZigBee (PAN), conforme demonstrado na Figura 13. Na Figura 14 mostra-se um exemplo de como enviar uma mensagem na topologia árvore das redes sem fio ZigBee, auxilia na expansão da rede e resolve problemas de obstáculos. Por exemplo, o dispositivo precisa enviar uma mensagem para outro dispositivo, mas há uma barreira entre eles que é difícil para o sinal penetrar, a topologia de árvore ajuda a mensagem "dar uma volta" na barreira roteando o pacote para outros dispositivos próximos os quais desviam a barreira para chegar ao dispositivo requerido (FARAHANI, 2011).

Figura 13 - Topologia árvore de uma rede sem fio ZigBee.



Fonte: (ALLIANCE, 2013)

O padrão IEEE 802.15.4, independentemente da sua topologia, sempre é criado por um dispositivo PAN coordenador. O coordenador do PAN controla a rede e executa as seguintes funções mínimas (FARAHANI, 2011):

- Atribuir um endereço exclusivo (16 bits ou 64 bits) para cada dispositivo na rede;
- Iniciar, encerrar e encaminhar as mensagens através da rede;
- Selecionar um identificador exclusivo para a rede PAN.

Existe apenas um coordenador PAN em toda a rede. Um coordenador PAN pode precisar ter longos períodos ativado (ligado em funcionamento) e, portanto, geralmente é ligado a uma

fonte principal, em vez de uma bateria. Todos os outros dispositivos da rede são normalmente alimentados a bateria (FARAHANI, 2011).

4.2.2 Camada Física

O padrão IEEE 802.15.4 adotar como protocolos a camada física (PHY) e a *Medium Access Control* (MAC). Portanto, um dispositivo compatível com ZigBee é compatível também com o padrão IEEE 802.15.4. (FARAHANI, 2011).

O número total de canais é 27, desses, 16 canais ficam na faixa de 2.450 MHz, 10 canais a 915 MHz e um em 868 MHz (LABIOD; AFIFI; SANTIS, 2007).

Nas redes sem fio ZigBee encontra-se a camada mais baixa do padrão IEEE 802.15.4 que é a camada física, ou PHY. Esta camada é a camada mais próxima ao hardware e o controla e se comunica com o rádio transceptor. A camada PHY é responsável pela ativação do rádio que transmite ou recebe pacotes. O PHY também seleciona a frequência do canal e certifica-se se este não está sendo utilizado por qualquer outro dispositivo da rede (FARAHANI, 2011).

4.2.3 Banda utilizada no Brasil

A camada física com frequência de 2.450 MHz possui uma taxa de transferência de 250 kbps. Os dados são mapeados primeiros em símbolos, 4 bits por 4 bits. Em seguida, os símbolos são mapeados em quadros de 32 bits de comprimento e, em seguida, modulados. Nota-se que na banda 2.402 a 2.480 MHz, podem-se ter vários periféricos, telefones sem fio, fornos de micro-ondas e outros equipamentos, gerando possíveis interferências (LABIOD; AFIFI; SANTIS, 2007).

4.3 CAMADA MAC

A subcamada MAC fornece dois serviços: o serviço de dados do MAC e o serviço de gerência do MAC. As características da subcamada MAC são: gerência de sinal, acesso do canal, gerência de GTS (*Guaranteed Time Slots*), envio de confirmação, associação e dissociação.

4.3.1 Camada de Controle de Acesso ao Meio

O padrão de redes sem fio IEEE 802.15.4 implementa um método para permitir que vários dispositivos possam usar o mesmo canal de frequência para o seu meio de comunicação. O mecanismo de acesso ao canal utilizado é a Camada de Controle de Acesso ao Meio (CSMA-CA). No CSMA-CA, quando um dispositivo deseja transmitir, ele primeiro faz uma avaliação para confirmar se o canal não está sendo utilizado para garantir que o canal não está em uso por

qualquer outro dispositivo. Em seguida, o aparelho começa a transmitir seu sinal próprio. A decisão de declarar um canal aberto ou não, pode ser baseado na medição da energia espectral no canal de frequência de interesse do dispositivo ou na detecção do tipo do sinal de ocupação do dispositivo (FARAHANI, 2011).

Quando um dispositivo irá transmitir um sinal, primeiro ele vai para o modo de recepção para detectar e estimar o nível de energia do sinal no canal desejado. Esta tarefa é conhecida como a detecção de energia (ED - *Energy Detection*). Em ED, o receptor não tenta decodificar o sinal e apenas o nível de energia é estimado. Se houver um sinal já na banda de interesse é aguardado até que o canal se torne livre. ED não determina se este é um sinal do padrão IEEE 802.15.4. Se o canal não está liberado, o dispositivo aguarda por um período de tempo aleatório e tenta novamente. Após um tempo de espera, o ED tenta novamente, até que o canal se torne claro ou o dispositivo atinja o número máximo de tentativas definidas pelo usuário (FARAHANI, 2011).

4.3.1.1 *Transferência de dados em redes com sinal ativado versus redes sem sinal*

Existem dois métodos de acesso ao canal: contenção ou baseado no argumento livre. Em contenção de acesso baseado em canais, todos os dispositivos que desejam transmitir no mesmo canal de frequência usam o mecanismo CSMA-CA, e o primeiro que encontra o canal começa a transmitir. No método de disputa de canal livre, o coordenador dedica ao PAN um horário específico para um dispositivo transmitir. Isto é chamado um tempo de intervalo garantido (GTS) (FARAHANI, 2011).

Uma rede em que o coordenador do PAN não transmite sinais é conhecido como uma rede sem sinal. Uma rede sem sinal não pode ter, portanto, GTS livres, porque os dispositivos não podem ser sincronizados aos *clocks* dos nós da rede. A duração da bateria em uma rede sem sinal pode ser visivelmente melhor do que em uma rede com sinal ativado, porque em uma rede sem sinal, os dispositivos necessitam ativar com menos frequência (FARAHANI, 2011).

4.3.1.2 *Métodos de transferência de dados*

Existem três tipos de transferência de dados suportados (FARAHANI, 2011):

- A partir de um coordenador para um dispositivo PAN;
- A partir de um dispositivo para um coordenador PAN;
- A partir de um dispositivo para outro dispositivo ponto a ponto.

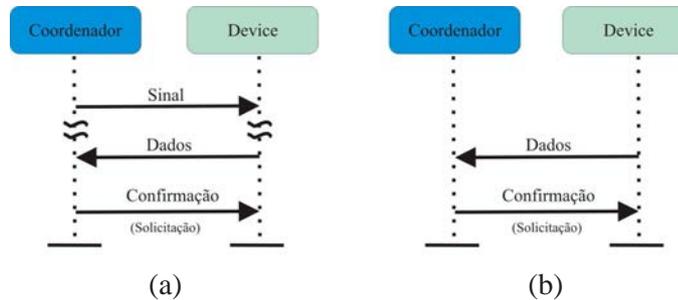
Em uma rede com topologia estrela, apenas os dois primeiros tipos de transferência são utilizados. Estas transações ocorrem de maneira diferente para redes com sinal ativo (*Beacon-Enabled*) e para redes com sinal desativado (*Non-Beacon-Enabled*) (FARAHANI, 2011).

4.3.1.3 Transferência de dados para um Coordenador

Em uma rede sem fio com sinal ativado, quando um dispositivo for transmitir dados para o coordenador, o dispositivo sincroniza seu *clock* e transmite os dados para o coordenador usando o método CSMA-CA. O coordenador pode acusar o recebimento dos dados, apenas se for solicitado pelo transmissor de dados, como apresentado na Figura 14(a) (FARAHANI, 2011).

A Figura 14 (b) mostra a sequência de transferência de dados em um sinal não ativado. Neste cenário, o dispositivo transmite os dados assim que o canal está livre. A transmissão de uma confirmação pelo coordenador do PAN é opcional (FARAHANI, 2011).

Figura 14 - Transferência de dados para um *coordenador* do padrão IEEE 802.15.4: a) Com sinal b) sem sinal



Fonte: (FARAHANI, 2011)

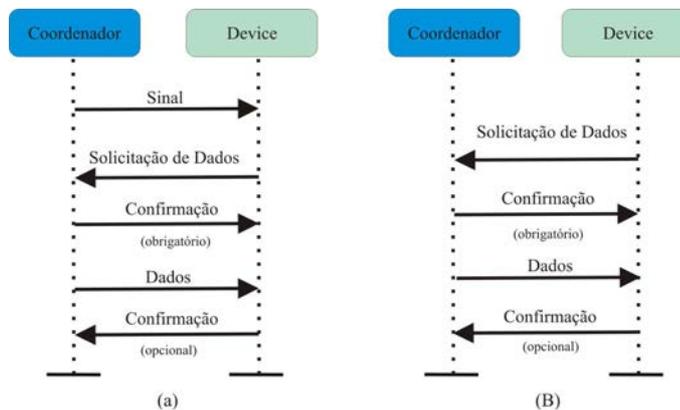
4.3.1.4 Transferência de dados a partir de um coordenador

A Figura 16(a) ilustra-se as etapas de transmissão de dados para transferir dados de um coordenador para um dispositivo *end Device*. Se o coordenador precisa transmitir dados para um dispositivo, indica na sua mensagem um sinal que uma mensagem de dados está pendente para esse dispositivo. O dispositivo envia uma mensagem de solicitação de dados ao coordenador indicando que ele está ativo e pronto para receber os dados. O coordenador confirma o recebimento do pedido de dados e envia os dados para o dispositivo. O envio da confirmação pelo dispositivo é opcional (FARAHANI, 2011).

Na transmissão de dados, numa rede de sinal não ativada, Figura 15(b) o coordenador precisa aguardar o dispositivo para solicitar os dados. Se o dispositivo solicita os dados, mas não há dados pendentes para esse dispositivo, o coordenador envia uma mensagem de confirma-

ção com um formato específico que indica que não há dados pendentes para esse dispositivo (FARAHANI, 2011).

Figura 15 - Transferência de dados de um coordenador para um dispositivo. a) com sinal b) sem sinal



Fonte: (FARAHANI, 2011)

4.3.2 Associação e dissociação

Serviços de associação e dissociação são fornecidos pelo IEEE 802.15.4 que podem ser utilizados para permitir que os dispositivos possam ingressar ou sair de uma rede. Por exemplo, quando um dispositivo quer participar de um PAN, ele envia uma solicitação de associação para o coordenador. O coordenador pode aceitar ou rejeitar o pedido de associação (FARAHANI, 2011).

Se um dispositivo associado quiser sair da PAN, ele emitirá um comando de notificação de dissociação ao coordenador. Com a recepção, o coordenador emite a confirmação. Mesmo se a confirmação não for recebida, o dispositivo se considerará desassociado (FARAHANI, 2011).

4.3.3 Segurança de uma rede ZigBee

A norma LR-WPAN especifica algumas medidas básicas de segurança na subcamada MAC incluindo a capacidade de manter uma lista de controle de acesso e prevê criptografia simétrica. Criptografia simétrica implica na utilização de uma chave única em ambos os extremos da comunicação. Esta chave pode ser compartilhada por um grupo de dispositivos ou por dois pares que se comunicam dentro de suas respectivas ACLs (*Access Control List*). Uma mensagem de integridade de código é utilizada para garantir a integridade do *frame*. A aplicação efetiva das funções de segurança, tais como a gestão das chaves e autenticação é deixada para as camadas superiores.

A norma permite quatro serviços de segurança: controle de acesso, criptografia de dados,

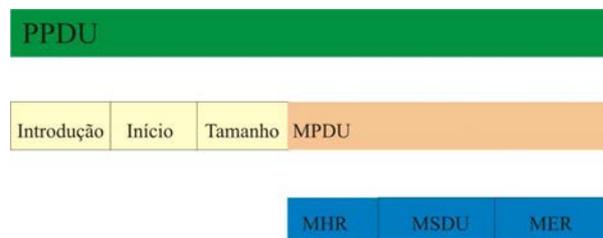
integridade de quadros e mensagens sequenciais. Sob o controle de acesso, cada dispositivo mantém uma lista de dispositivos com os quais a comunicação é desejada. Quadros de dispositivos indesejados são rejeitados. A modificação deste código requer o uso da chave criptográfica. O mecanismo é aplicável a *frames* sequenciais ordenados em sequência numérica de que foram transmitidos. O receptor é capaz de determinar se um *frame* é novo ou se teve que ser reenviado, caso os *frames* sejam reenviados, estes podem ser rejeitados (KUBAN, 2006).

4.4 ORGANIZAÇÃO DO FRAME

O padrão divide em quatro diferentes tipos de estruturas de *frames*. Sinal de *frames*, *frame* de dados, o de confirmação de recebimento de recepção de uma mensagem e de controle de transferências (KUBAN, 2006).

A estrutura geral de um frame é mostrada na Figura 16, sendo o MPDU (*Mac protocol Data Unit*) e o PPDU (*Physical Protocol Data Unit*). O MPDU contém o cabeçalho MAC, o serviço de dados MAC unidade (MSDU) e do MAC rodapé (MFR) (KUBAN, 2006).

Figura 16 - Organização do frame WPAN.



Fonte: Adaptado de (KUBAN, 2006)

4.5 GATEWAY DE UMA REDE ZIGBEE

Um *gateway*¹ de uma rede sem fio utilizando o padrão IEEE 802.15.4 - ZigBee oferece a interface entre uma rede ZigBee e uma outra rede utilizando um padrão diferente. Por exemplo, se a rede sem fio ZigBee é usado para reunir informações do paciente dentro de uma sala, as informações poderão ser transmitidas pela internet para uma estação de monitoramento. Neste caso, o *gateway* ZigBee implementa tanto o protocolo de ZigBee como o padrão de internet, para ser capaz de traduzir os pacotes de Internet para ZigBee e vice-versa (FARAHANI, 2011).

¹Gateway, ou porta de ligação, é uma máquina intermediária geralmente destinada a interligar redes, separar domínios de colisão, ou mesmo traduzir protocolos.

4.6 EXEMPLOS DE APLICAÇÕES UTILIZANDO O PADRÃO ZIGBEE

Um exemplo de uma aplicação do padrão ZigBee é o monitoramento de pacientes em tratamento médico, que ficam em suas residências. A pressão arterial do paciente e da frequência cardíaca, por exemplo, pode ser medido por meio de dispositivos portáteis. O paciente usa um dispositivo utilizando o padrão ZigBee que interage com um sensor que reúne informações relacionadas à sua saúde. Em seguida, os dados são transmitidos sem fio para um servidor local, como por exemplo, um computador pessoal, dentro da casa do paciente, onde a análise inicial é executada. Finalmente, a informação é enviada através da internet para o servidor do médico, para posterior análise (FARAHANI, 2011).

Outro exemplo de uma aplicação do padrão ZigBee é monitorar a integridade estrutural de grandes escalas de edifício. Nesta aplicação, o padrão ZigBee possui vários sensores sem fio (por exemplo, acelerômetros) habilitados, podendo ser instalado em um prédio, e todos esses sensores podem formar uma única rede sem fio para reunir as informações que serão utilizadas para avaliar o edifício e detectar sinais de possíveis danos. Após um terremoto, por exemplo, um edifício poderia exigir uma inspeção antes que reabra ao público. Os dados recolhidos pelos sensores podem ajudar a agilizar e reduzir o custo da inspeção (FARAHANI, 2011).

5 FERRAMENTAS UTILIZADAS E METODOLOGIA DE DESENVOLVIMENTO

Neste capítulo, apresentam-se as principais ferramentas que serão utilizadas no desenvolvimento do projeto, suas características e onde se enquadram dentro do projeto. Também aborda uma aplicação em interface gráfica (software), desenvolvida em linguagem Java, para captura de imagens estáticas *bitmap* utilizando compressão com perdas, preparando a imagem com a redução do seu tamanho, para ser transmitida posteriormente pelo padrão ZigBee.

5.1 LINGUAGEM JAVA

No desenvolvimento deste projeto utilizou-se a linguagem de programação Java. Esta linguagem foi desenvolvida pela empresa SUN MicroSystems, em 1991. James Gosling, o seu criador, a chamou de Oak em homenagem a uma árvore de carvalho vista pela sua janela na SUN. Em 1995 graças ao estouro da internet, a linguagem Oak foi adaptada para o desenvolvimento de aplicações para web e foi rebatizada como Java (cidade de origem de um tipo de café importado) (DEITEL, 2010).

Segundo a Sun Microsystems, Java foi estruturada para ser: orientada a objetos, ter códigos pequenos, robusta, segura, simples com alto desempenho, interpretável com linhas de execução, dinâmica e principalmente portátil a todas as plataformas, tanto o código fonte como os binários (CORNELL GARY; HORSTMANN,).

Um programa fonte escrito em linguagem Java é traduzido pelo compilador para os *bytecodes*¹, que são interpretados pela Máquina Virtual Java (JVM). A vantagem desta técnica é evidente: garantir uma maior portabilidade para os programas Java em código-fonte e compilados (DEITEL, 2010). Atualmente Java é distribuída nas seguintes versões:

- JSE (*Java Standard Edition*) - Seu uso é voltado para PCs e servidores. Contém todo o ambiente necessário para a criação e execução de aplicações *desktop* e web, de pequeno e médio porte.
- JEE (*Java Enterprise Edition*) - é voltada para o desenvolvimento de softwares corporativos, baseados em componentes que são executados em um servidor de aplicação.

¹*bytecodes* O código Java é compilado e este pseudocódigo gerado é formado por *bytecodes*. Os *bytecodes* são, em geral, executados por um outro programa, o interpretador Java (JVM - Máquina Virtual JAVA). O interpretador Java entende o conjunto lógico de instruções representado pelos *bytecodes*, simulando a arquitetura virtual para a qual foram desenvolvidos.

- JME (*Java Micro Edition*) - ambiente de desenvolvimento para dispositivos móveis ou portáteis, como telefones celulares, *ipads smartphones*.
- JRE (*Java Runtime Environment*) - significa ambiente de tempo de execução. É um pacote de softwares que é executado como um aplicativo do sistema operacional e que interpreta a execução de programas Java. A JRE é composta pela JVM somada ao conjunto de APIs. (JVM + APIs = JRE).
- JVM (*Java Virtual Machine*) - Máquina Virtual Java. Software que emula uma CPU e memória para a execução de programas Java.
- JDK (*Java Development Kit*) ou SDK (*Software Development Kit*) - significa pacote de desenvolvimento Java. Conjunto de ferramentas para a compilação, documentação e *debug* de aplicativos Java. Composto pela JRE somada as ferramentas de desenvolvimento.

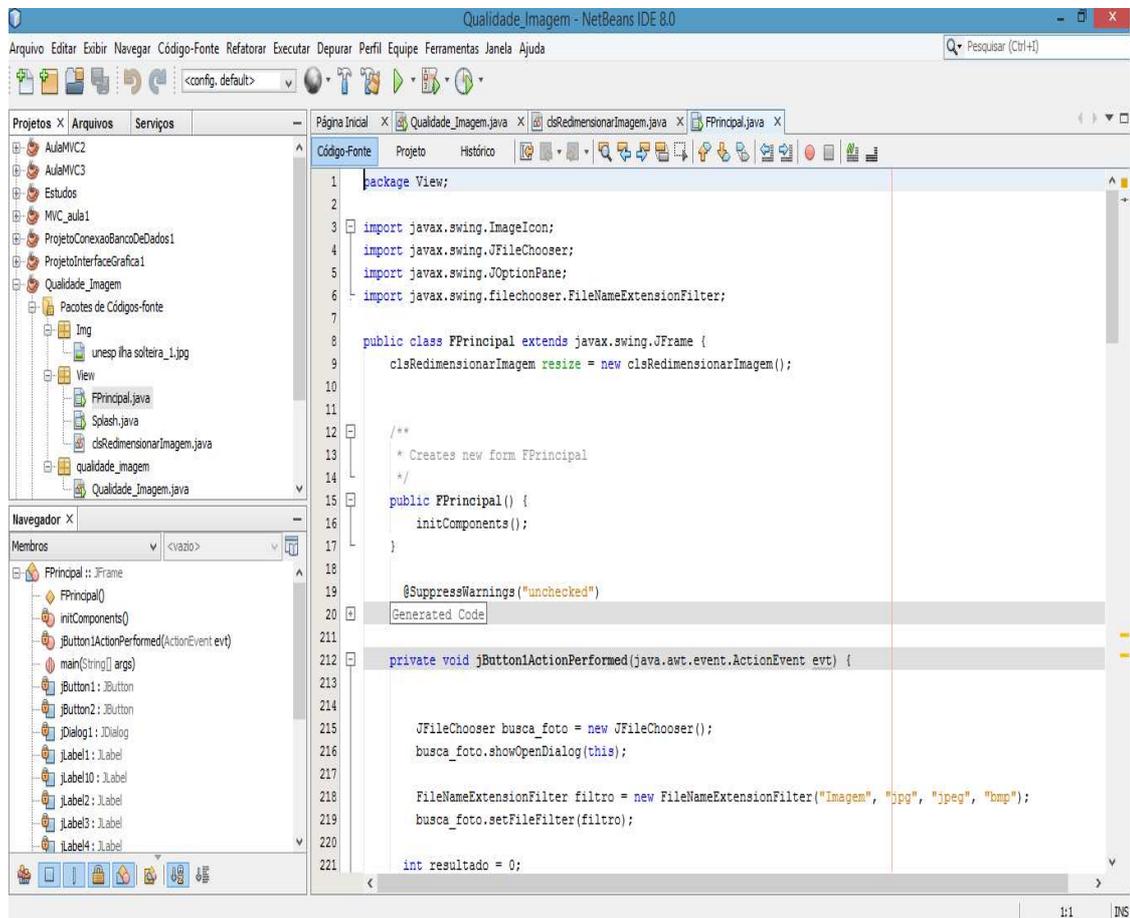
5.2 FERRAMENTA DE DESENVOLVIMENTO JAVA - NETBEANS

A linguagem de programação Java a cada dia torna-se uma boa solução para complexos problemas, podendo ser utilizada em computadores pessoais, Internet ou até mesmo celulares. A mudança de paradigma de programação pode dificultar o desenvolvimento pois a utilização de programação estruturada comparada a orientada a objeto causou preocupações para os seus desenvolvedores, mas depois do início do desenvolvimento com a utilização da orientação a objeto é factível o ganho de tempo e organização no projeto. (ORACLE, 2013).

Algumas empresas têm desenvolvido significantes melhorias em suas IDEs, ou seja, no seu ambiente integrado de desenvolvimento, escrito completamente em Java. O NetBeans nasceu de uma pequena empresa da República Tcheca, que posteriormente foi adquirida pela empresa Sun Microsystems. James Gosling, criador da linguagem Java, assegura que desenvolve suas aplicações utilizando o NetBeans, com aquisição pela Oracle Corporation divulgada em 20 de abril de 2009. (ORACLE, 2013).

A versão atual da IDE é a versão 8, onde a intenção é tornar a ferramenta uma plataforma de desenvolvimento para várias linguagens, sendo um projeto livre para uso comercial e não comercial (GONCALVES, 2008). O NetBeans é um produto de código aberto com todos os benefícios de um software livre analisado por uma comunidade de desenvolvedores (ORACLE, 2013). Na Figura 17 apresenta-se a interface da IDE NetBeans versão 8.

Figura 17 - Interface da IDE NetBeans versão 8.

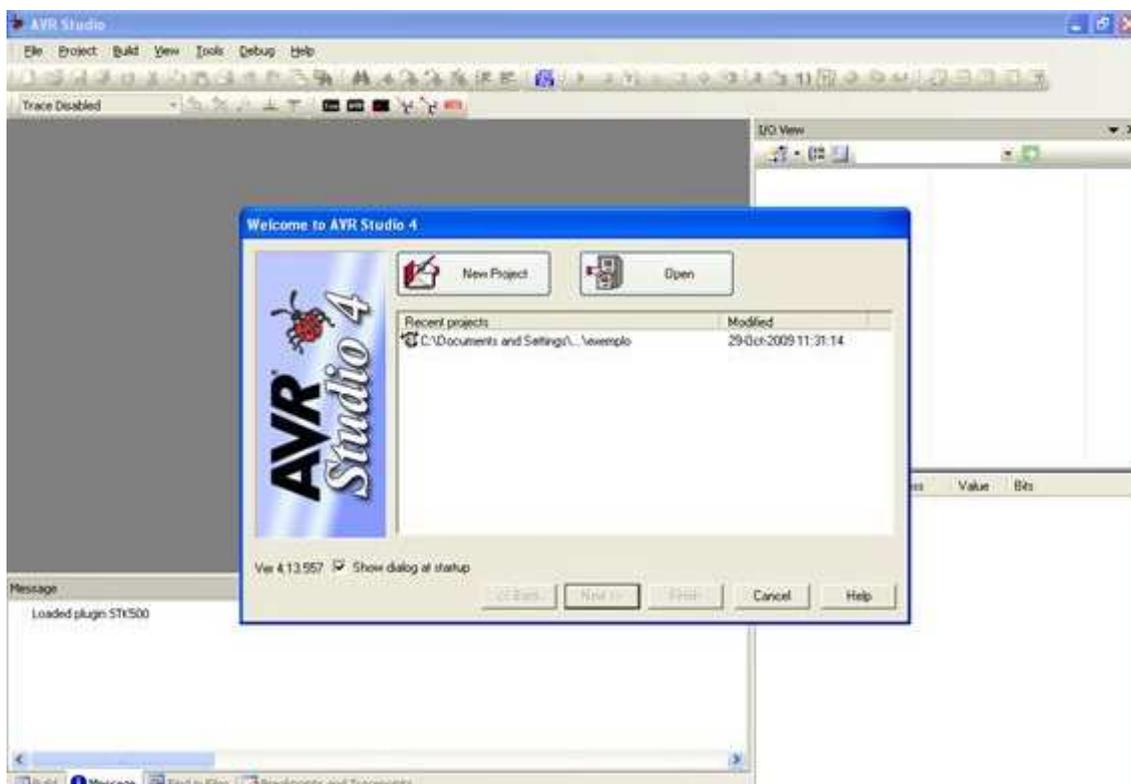


Fonte: (NETBEANS, 2013)

5.3 IDE E SIMULADOR AVR STUDIO

Desenvolvido pela empresa Atmel, o AVR Studio é um ambiente de desenvolvimento integrado, distribuído gratuitamente, desenvolvido para auxiliar a programação e simulação dos microcontroladores da família AVR da Atmel. O ambiente contém um editor de programas em linguagem de montagem, um montador e um simulador, o qual simula o funcionamento do programa desenvolvido no micro controlador. (GONCALVES, 2008). A interface da IDE AVR Studio é apresentada na Figura 18.

Figura 18 - Interface IDE AVR Studio.



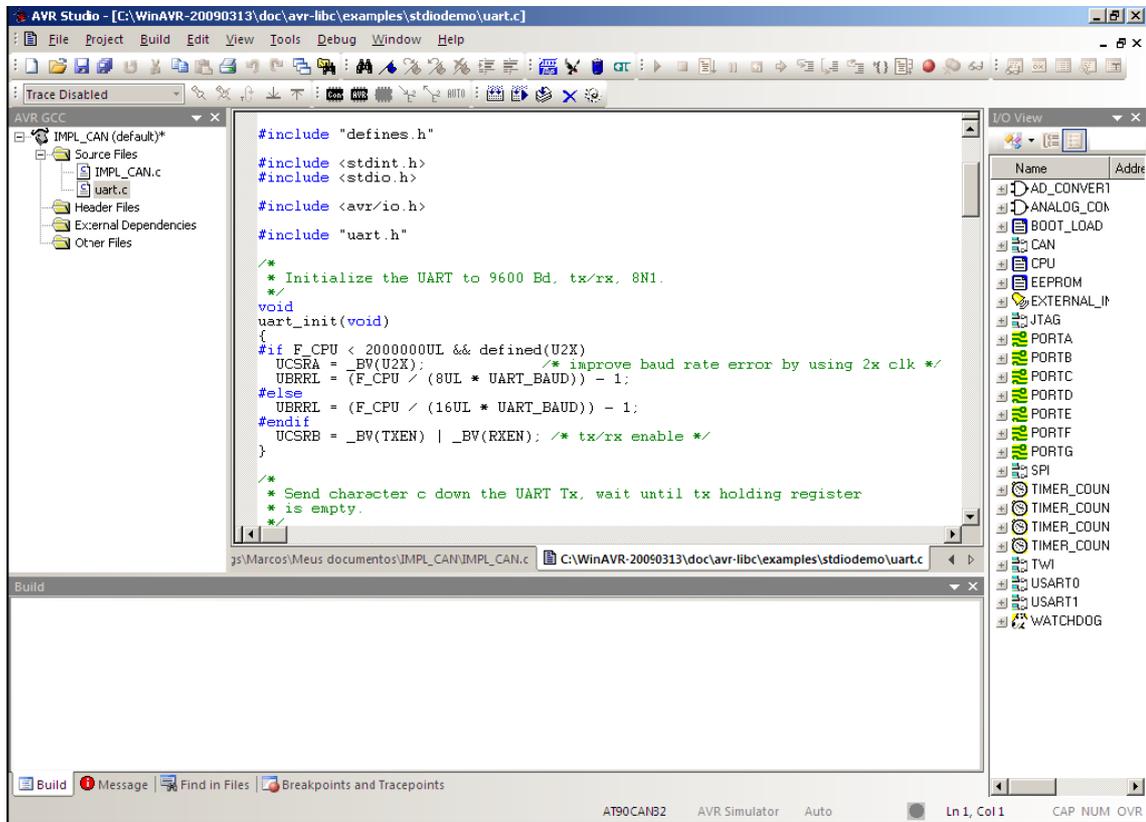
Fonte: (ATMEL, 2014).

5.4 COMPILADOR LINGUAGEM DE PROGRAMAÇÃO C - WINAVR

O compilador utilizado no projeto para transformar o código fonte em um arquivo binário para ser executado pelo microcontrolador foi o WinAVR, por ser facilmente integrado ao AVR Studio, como vantagens ele possui uma extensa biblioteca de funções na linguagem C e também é gratuito.

Na Figura 19 mostra-se a utilização da suite de aplicações AVR Studio para programação de códigos utilizando a linguagem C (WinAVR).

Figura 19 - Interface AVR Studio com WinAvr.

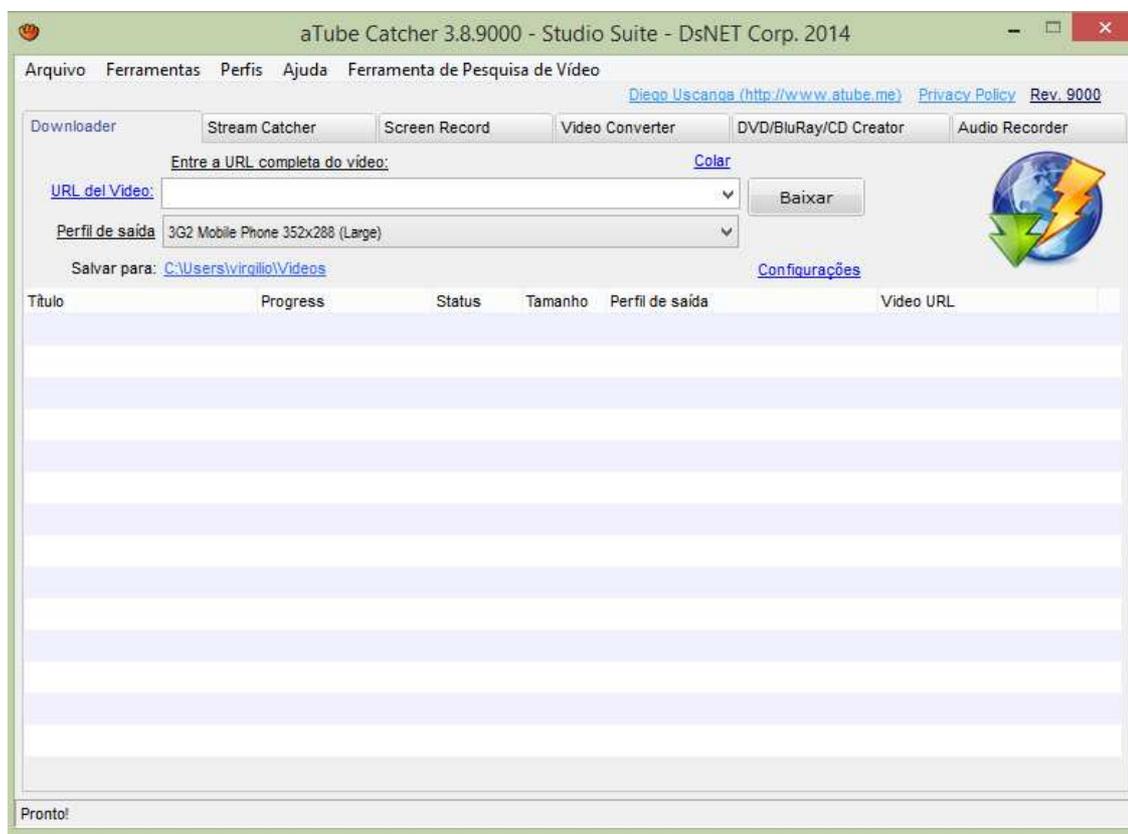


Fonte: (ATMEL, 2014)

5.5 COMPRESSÃO DE VÍDEO - SOFTWARE ATUBE CATCH

Para a compressão de vídeo foi utilizado o software livre Atube Catch cujo formato padrão utilizado pelo software é o MPEG4 AVI, mas o programa é compatível com 3GP, 3G2, AVI, XVID, MPEG4, MP3, MP2, WMA, WMV, GIF, FLAC, WAV, PSP, MPG, VOB, OGG, MOV e AVI. Conforme apresentado a interface na Figura 20 (ATUBE, 2015).

Figura 20 - Interface Atube Catch.

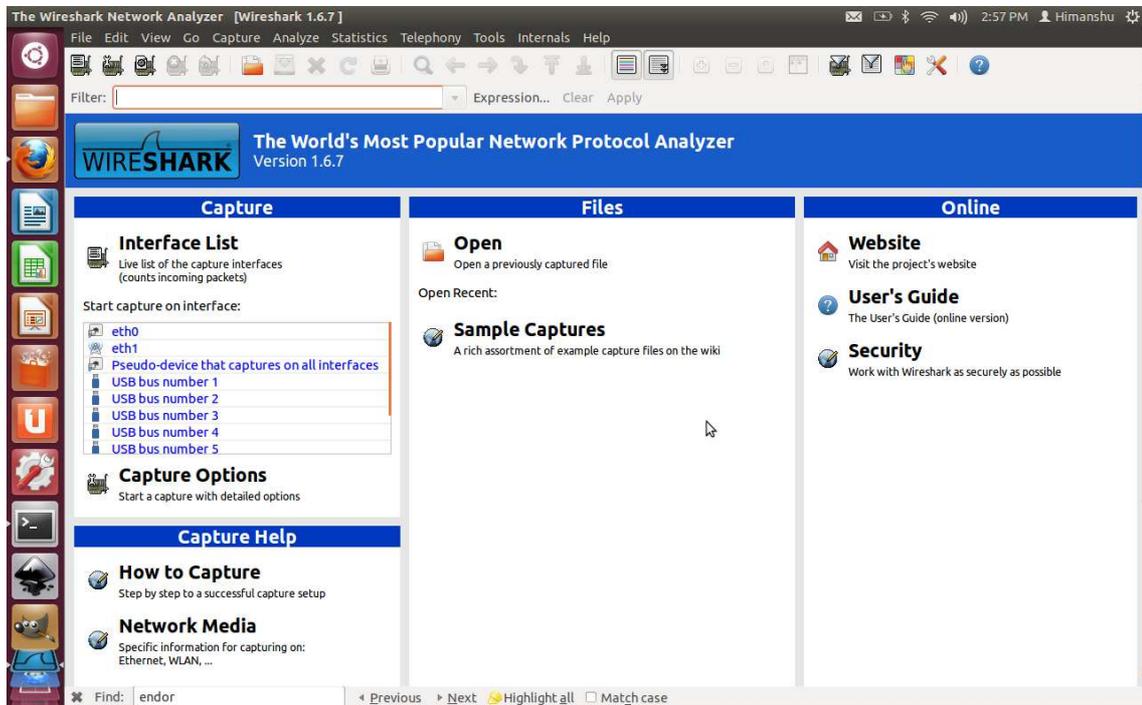


Fonte: (ATUBE CATCH, 2014)

5.6 ANÁLISE DE TRÁFEGO WIRESHARK

Para analisar o tráfego da rede foi utilizado o software livre WireShark registrado pela GNU (*General Public License*) pois o mesmo é multi-plataformas e pode ser executado em sistemas operacionais da Microsoft, Linux e outros. O software analisa o tráfego de pacotes recebidos que passa através do meio físico onde ele está conectado e os organiza por protocolo, podendo ser salvo o tráfego para ser analisado *on-line*, em tempo real de transmissão ou *off-line* com os pacotes armazenados, verificando o tempo de transmissão do pacote, camadas utilizadas e perda de pacotes (WIRESHARK, 2015). Segundo Wang, Xu e Yan (2010) Wireshark é uma excelente ferramenta com análise e medição de monitoramento de tráfego de rede. Visualiza-se a interface da ferramenta na Figura 21.

Figura 21 - Interface Interface WireShark.



Fonte: (WIRESHARK, 2015)

5.7 MÓDULOS UTILIZADOS PARA O DESENVOLVIMENTO DA REDE ZIGBEE

Para o desenvolvimento do projeto foi utilizados três dispositivos da empresa Atmel: módulo STK500, módulo de expansão STK501 e antena de comunicação RZ502. A seguir as principais características de cada um destes módulos.

5.7.1 MÓDULO STK500

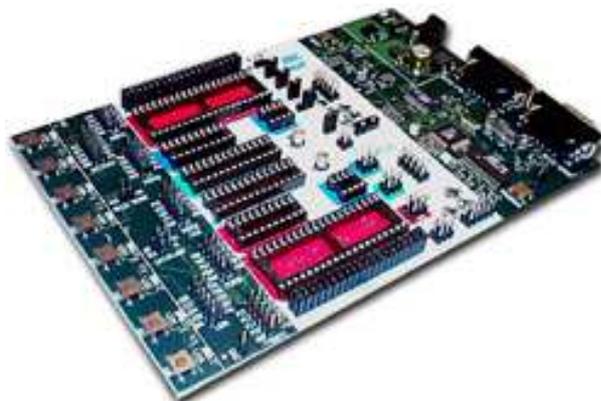
O módulo da Atmel AVR STK500 possui ferramentas de programação e sistema de desenvolvimento e microcontroladores Atmel AVR Flash. Permite um desenvolvimento rápido de códigos, protótipos, testes e depuração dos novos projetos.

O STK500 é compatível com o ambiente de desenvolvimento AVR Studio, possui interface RS232 para PC para programação e controle, regulador de fonte de alimentação para 10V - 15V DC, adaptador para dispositivos AVRs de 8, 20, 28 e 48 pinos, 8 botões de pressão e 8 LEDs para uso geral. Todas as portas de E/S de AVRs são facilmente acessíveis através de pinos em linha, porta RS232 de uso geral, conectores de expansão para módulos plug-in e área de prototipagem. Possui memória *flash onboard* para armazenamento não-volátil de dados, software de atualização para futuro suporte de dispositivos AVRs, sistema de tensão e *reset*

flexível.

Na Figura 22 apresenta-se o STK500, que através do AVR Studio pode ser configurado, programado e simulado.

Figura 22 - Interface STK 500.



Fonte: Elaborado pelo Autor

O dispositivo STK500 suporta Interface Serial para Periféricos (ISP) de todos microcontroladores AVR, assim como programação de sistemas externos. É um sistema de transferência síncrona de dados entre o dispositivo mestre e os periféricos ligados a ele. É do tipo *full duplex* e utiliza três linhas de sinais para trabalhar desde modo: receber, transmitir e *clock*. O AVR pode trabalhar tanto no modo mestre quanto no modo escravo. Uma das formas mais comuns e baratas de programação do microcontrolador faz-se através da ISP (MAURICI, 2005).

As portas de E/S de AVR são acessíveis através de pinos em fila que podem ser utilizados para ligar LEDs, botões de pressão ou sinais externos. As opções Flash, EEPROM e todos fusíveis podem ser programados individualmente ou com opção de programação automática. A frequência do relógio e a tensão de alimentação são controlada através de AVR Studio.

5.7.2 Módulo STK501

A placa STK501 apresentado na Figura 23 é um módulo de expansão projetado para adicionar suporte a 64 pinos para a câmara de desenvolvimento ATMEL STK500. O STK501 inclui conectores, *jumpers* e hardwares que permitem a plena utilização das novas funcionalidades do ATmega128 (L), enquanto o soquete ZIF (*Zero Insertion Force*) permite a fácil utilização de pacotes para prototipagem. Além de adicionar suporte para novos dispositivos, o STK501 também fornece suporte para os periféricos anteriormente não suportados pelo STK500. Apresenta também uma entrada adicional RS-232 e uma interface SRAM externa.

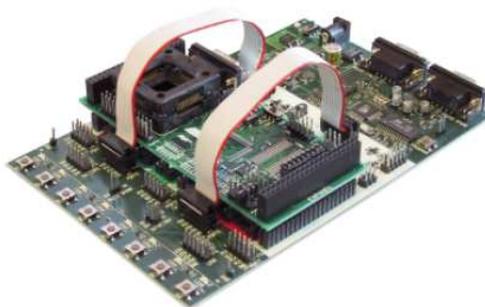
Figura 23 - Módulo Atmel STK 501.



Fonte: Elaborado pelo autor.

O módulo STK501 possui interface com o AVR Studio e compatível com STK500, conforme mostrado na Figura 24. Suporta microcontroladores Atmega64, Atmega128, Atmega1281 e Atmega2561 (ATMEL, 2009).

Figura 24 - STK 501 conectado no STK 500.



Fonte: Elaborado pelo autor.

5.7.3 Antena de comunicação RZ502

A antena de comunicação RZ502, da Atmel, faz parte dos componentes de acessórios e permite o desenvolvimento, depuração e demonstração. Topologias de rede mais avançadas podem ser desenvolvidas com a adição de componentes de acessórios. O componente é composto por duas placas com antenas de rádio SMA e dois microcontroladores AVR ATMEGA1281. Para a sua utilização é necessário o módulo STK 500, a placa de expansão STK 501 e cabos para a conexão nos pinos de E/S (ATMEL, 2014). Na Figura 25, é ilustrado o rádio transceptor RZ502.

Figura 25 - Rádio Transceptor RZ502.



Fonte: Elaborado pelo autor.

5.8 CONEXÃO DOS MÓDULOS UTILIZADOS

Para implementação do projeto, são acoplados entre os dispositivos mencionados nos itens 5.7.1 (STK500), 5.7.2 (STK 501) e 5.7.3 (antena RZ502), utilizando-se o microcontrolador Atmega1281, conforme mostrado na Figura 26.

Figura 26 - Módulo STK 500 + módulo STK 501 + rádio transceptor RZ502.

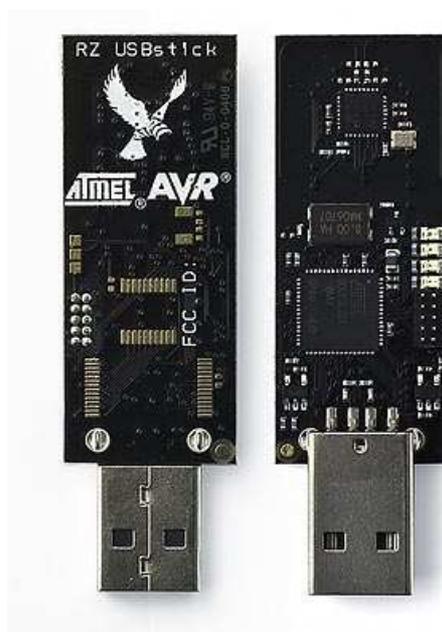


Fonte: Elaborado pelo autor

5.9 TRANSCEPTOR RZ RAVEN

O dispositivo RZ RAVEN inclui duas placas AVR com um transceptor 2,4 GHz, processadores de bordo da aplicação AVR com *display* LCD e uma USB com um transceptor 2,4 GHz para permitir conexões USB a um PC conforme apresentado na Figura 27. Ele também inclui um pacote de software para PC de configuração de rede sem fio, controle e análise (ATMEL, 2014).

Figura 27 - Módulo transceptor RZ RAVEN.

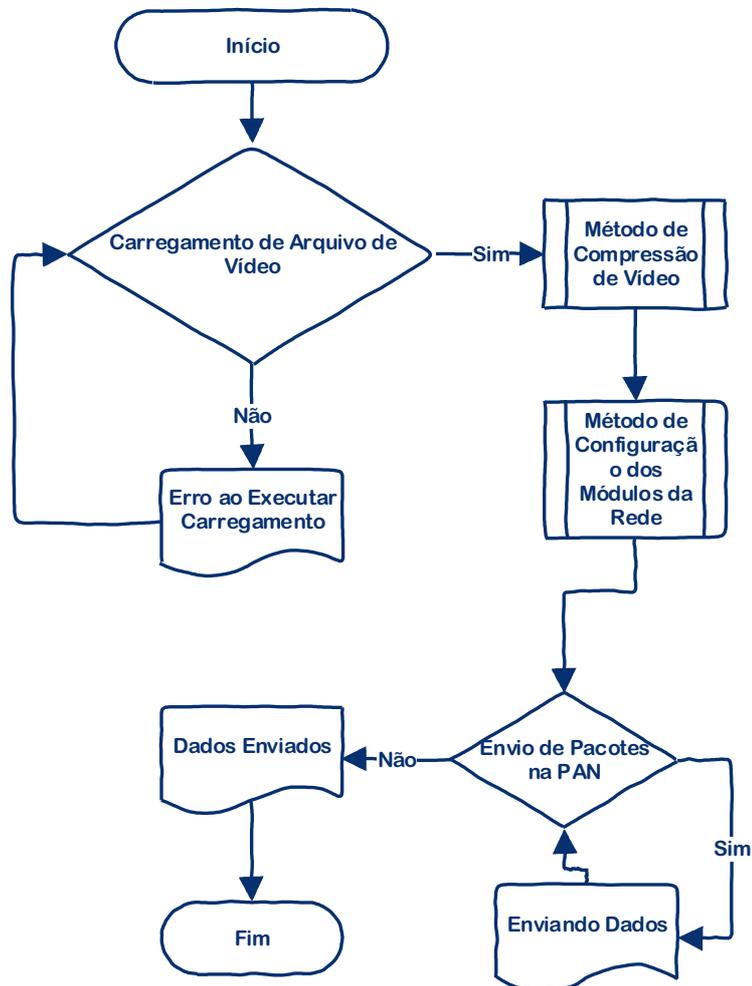


Fonte: Elaborado pelo autor

5.10 COMUNICAÇÃO SEM FIO (IEEE 802.15.4) COM IMAGENS E VÍDEOS

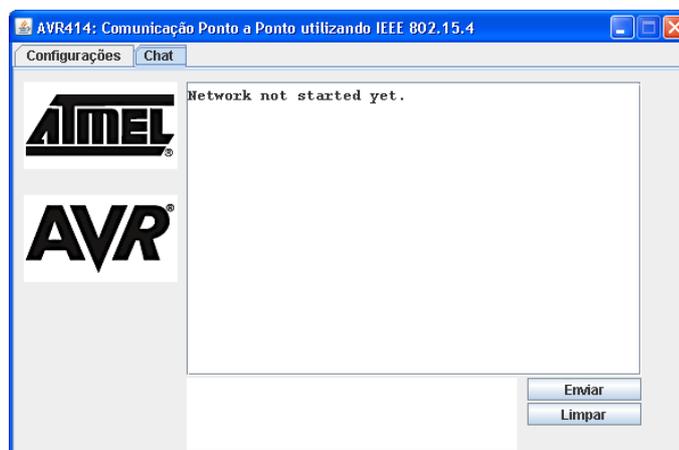
A configuração do ambiente de transmissão é idêntica ao apêndice A com as alterações na programação da aplicação na linguagem java, onde o objetivo agora não é a transmissão via texto e sim a transmissão de imagens e vídeos. Na Figura 28 apresenta-se o fluxograma do algoritmo desenvolvido em linguagem JAVA para o envio dos tipos de arquivos em vídeo e imagens, utilizando o padrão IEEE 802.15.4. A aplicação é iniciada como mostrado na Figura 29.

Figura 28 - Fluxograma do algoritmo de transmissão de imagens e vídeo.



Fonte: Elaborado pelo autor

Figura 29 - Tela de interface da comunicação ponto a ponto utilizando o padrão IEEE 802.15.4.



Fonte: Elaborado pelo autor

Após o *Coordenador* e o *End Device* estarem disponíveis na rede, tanto o *Coordenador* como o *End Device* estão aptos ao envio de imagens. Para iniciar a transmissão basta um ponto ser iniciado no botão *Enviar* da aplicação, o qual está desenvolvida para o envio do arquivo que está disponível em arquivo num local pré-selecionado e imediatamente é feito o início da transmissão. O ponto que está recebendo a transmissão imprime na aplicação o tamanho do arquivo em *bytes*. Após a transmissão da imagem na aplicação será impresso o aviso de fim da transmissão. A imagem ficará armazenada em local pré-determinado, e poderá ser visualizada por qualquer editor ou visualizador de imagem *bitmap*.

5.11 DESENVOLVIMENTOS DE UM SOFTWARE COM INTERFACE GRÁFICA PARA COMPRESSÃO DE IMAGENS.

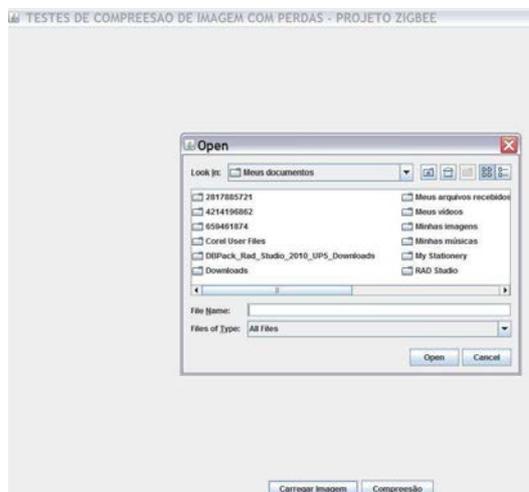
A interface gráfica proposta, desenvolvida na linguagem Java, tem como objetivo facilitar a interface com usabilidade simples, conforme apresentada na Figura 30. Tem-se como primeira opção o botão *Carregar Imagem* para assim, em seu sistema operacional, abrir o navegador para selecionar a imagem pretendida conforme mostrado na Figura 31.

Figura 30 - Tela da interface Compressão.



Fonte: Elaborado pelo autor

Figura 31 - Tela para seleção da imagem no navegador.



Fonte: Elaborado pelo autor

Após selecionar a localização da imagem a mesma será aberta dentro da aplicação gráfica para ser visualizada em tamanho real, conforme apresentado na Figura 32.

Figura 32 - Tela da interface do software.



Fonte: Elaborado pelo autor

Na Figura 33 apresenta-se a interface do software, sendo visualizada depois de executar o algoritmo de compressão de imagens com perda de qualidade, nesse caso ilustrado, utiliza-se conceitos de escalas de cinza na imagem.

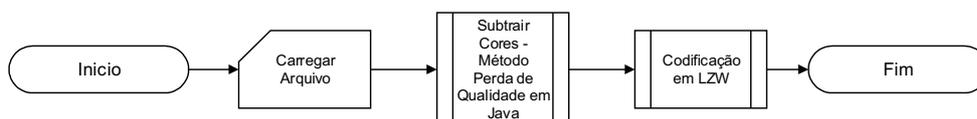
Figura 33 - Tela da interface do software.



Fonte: Elaborado pelo autor

A Interface desenvolvida para melhor transmissão e um arquivo utilizando o padrão IEEE 802.15.4 utiliza a compressão com perda, ou seja, no caso específico a imagem terá as cores retiradas por tons de cinzas e a extensão de arquivo é modificada da extensão BMP para a extensão JPG utilizando compressão LZW conforme demonstrado no fluxograma da Figura 34.

Figura 34 - Fluxograma do software desenvolvido.



Fonte: Elaborado pelo autor.

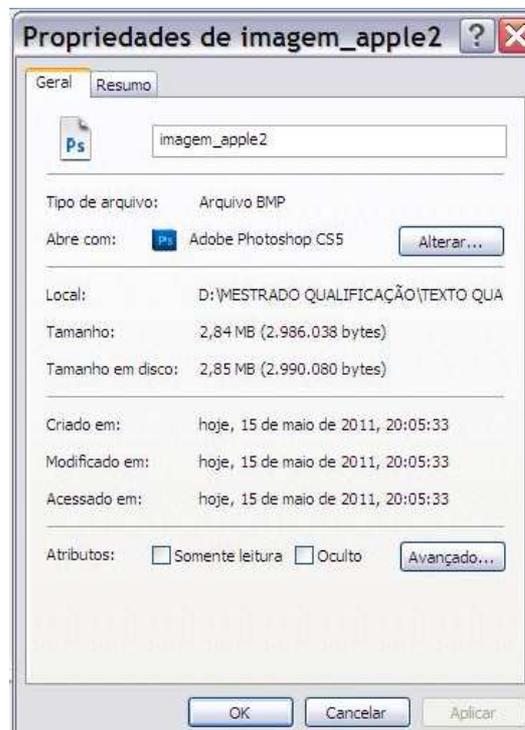
6 RESULTADOS E DISCUSSÕES

Neste capítulo serão abordados os resultados da compressão das imagens utilizando a interface gráfica desenvolvida, resultados da compressão de vídeos do software Atube Catch e, análises e cenários da transmissão de dados, utilizando o padrão IEEE 802.15.4 ZigBee.

6.1 TESTES DE COMPRESSÃO DE IMAGENS BITMAP INTERFACE GRÁFICA

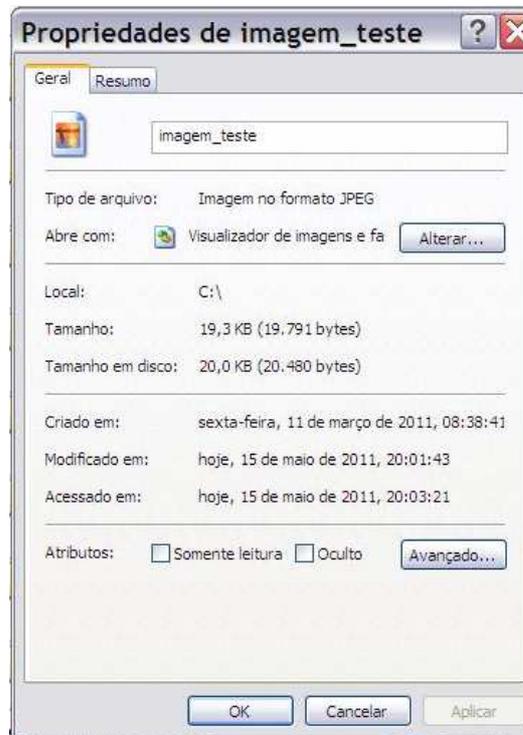
Para diminuir o tempo de transmissão foi utilizado o software desenvolvido nesta dissertação para compressão de imagens bitmap. Na Figura 35, as propriedades da imagem original selecionada, na qual o tamanho (2,84 MB) e extensão (BMP) podem ser verificados. Já na Figura 36 fica evidenciado que o tamanho do arquivo depois de ser comprimido passou para 19,3 kb sendo salvo com formato JPG e em tons de cinza.

Figura 35 - Tamanho do arquivo original selecionado.



Fonte: Elaborado pelo autor.

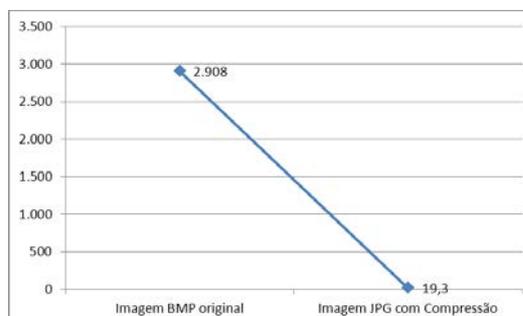
Figura 36 - Propriedades do Arquivo após a compressão



Fonte: Elaborado pelo Autor

Na Figura 37 é apresentado o resultado da compactação em kB de uma imagem com formato BMP de 2.908 kb, para uma imagem JPG com 19,3 kB depois de realizada a compressão com perda de qualidade na imagem.

Figura 37 - Comparativo de tamanho após compressão da imagem.



Fonte: Elaborado pelo autor.

Outros testes foram realizados, como por exemplo, na Figura 38 são apresentadas três imagens coloridas em padrão RGB na extensão BMP, as mesmas foram compactadas utilizando o software, conforme verificado na Figura 39.

Figura 38 - Apresentação de três imagens bitmap coloridas.



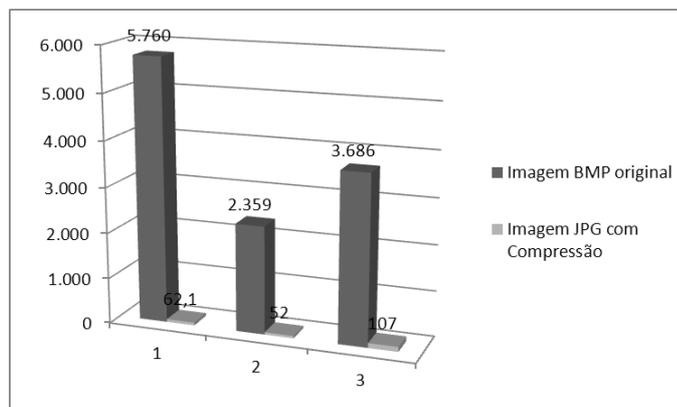
Fonte: Elaborado pelo Autor

Figura 39 - Apresentação de três imagens bitmap após ter sido compactadas pelo *software*

Fonte: Elaborado pelo autor.

Conforme mostrado na Figura 40 verificou-se através de análises que a taxa de compressão das imagens é satisfatória quando as mesmas estão em formato BMP. Os arquivos foram compactados utilizando técnicas que reduzem a qualidade do arquivo original, transformando um arquivo que era colorido para um em tons de cinzas no formato JPG, proporcionando assim, uma melhor e mais rápida transmissão utilizando o meio de comunicação sem fio através do padrão IEEE 802.15.4.

Figura 40 - Comparativo de tamanho em arquivo.



Fonte: Elaborado pelo autor.

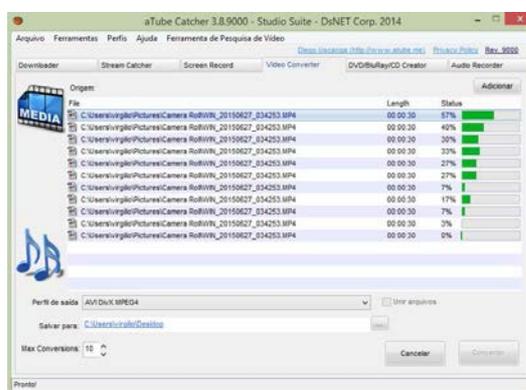
6.2 COMPRESSÃO DE VÍDEOS UTILIZANDO O SOFTWARE ATUBE CATCHER

Para a escolha da ferramenta de compressão de vídeo Atube Catcher foram realizados testes em laboratório para comprovar a sua eficácia como ilustrado na Figura 41. Os testes foram realizados nos seguintes cenários:

Cenário 1 - Teste de compressão em um vídeo de 30 segundos com 30 quadros por minuto;

Cenário 2 - Teste de compressão em um vídeo de 5 minutos com 30 quadros por minuto.

Figura 41 - *Software* Atube Catcher.



Fonte: Elaborado pelo autor.

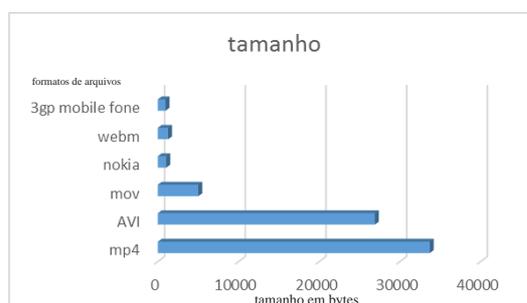
Cenário 1 - Utilização de um vídeo de 30 segundos com 30 quadros por minuto salvo na extensão de arquivo MPEG-4 no tamanho de 33 Mb, sendo utilizado o software Atube Catch com compressão para as extensões AVI, MOV, NOKIA, WEBM e 3GP na Tabela 1 e na Figura 42 apresentam-se as referidas compressões, sendo que, a de maior desempenho é a de formato 3GP. Com a compressão para a extensão 3GP conseguiu-se uma diminuição de tamanho final de arquivo em 35 vezes do tamanho original. Houve uma redução da resolução do vídeo, com pequena alteração da qualidade da imagem, mais o qual não comprometerá a visualização do mesmo.

Tabela 1 - Comparativo dos tipos de arquivos a a compressão obtida. Cenário 1.

Tipo de arquivo	Tamanho em bytes
MP4	33.710
AVI	26.900
Mov	5.029
Nokia	1.062
Webm	1.290
3GP	962

Fonte: Elaborado pelo autor.

Figura 42 - Demonstração da compressão da Tabela. Cenário 1.



Fonte: Elaborado pelo autor.

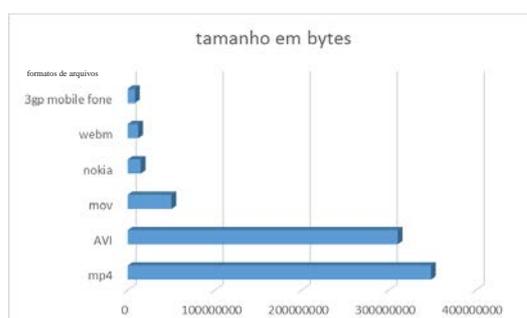
Cenário 2 - Utilização de vídeo de 5 minutos com 30 quadros por minuto salvo na extensão de arquivo MPEG-4 no tamanho de 300 Mb, sendo utilizado o software Atube Catch. Com compressão para as extensões AVI, MOV, NOKIA, WEBM e 3GP, fica também evidenciado que o melhor desempenho de compressão se dá para o formato 3GP. Nesta análise, obteve-se uma redução de 41 vezes no tamanho do arquivo original, conforme apresentado na Tabela 2 e na Figura 43.

Tabela 2 - Comparativo dos tipos de arquivos a a compressão obtida.

Tipo de arquivo	Tamanho em bytes
MP-4	348519551
AVI	310295926
Mov	50339889
Nokia	14969332
webm	12062333
3GP	8444222

Fonte: Elaborado pelo autor.

Figura 43 - Demonstração da compressão da Tabela 2. Cenário 2.



Fonte: Elaborado pelo autor.

6.3 COMUNICAÇÃO SEM FIO (IEEE 802.15.4) COM IMAGENS E VÍDEOS

Foi utilizado o software livre WireShark juntamente com o transceptor RZ Raven para capturar os pacotes de dados na transmissão dos arquivos na rede. Neste trabalho foi coletada informações dos pacotes na rede de arquivos com formatos diferentes. Utilizou-se um arquivo com extensão JPG com imagens bitmap e outro arquivo de vídeo com formato MPEG-4 para a análise dos pacotes coletados na rede através do WireShark para os seguintes cenários:

Cenário 1 - Transmissão de uma imagem bitmap de 2000 bytes. Medição de latência através da interface IEEE 802.15.4;

Cenário 2 - Transmissão de uma imagem bitmap de 5000 bytes. Medição de latência através da interface IEEE 802.15.4;

Cenário 3 - Transmissão de uma imagem bitmap de 10.000 bytes. Medição de latência através da interface IEEE 802.15.4;

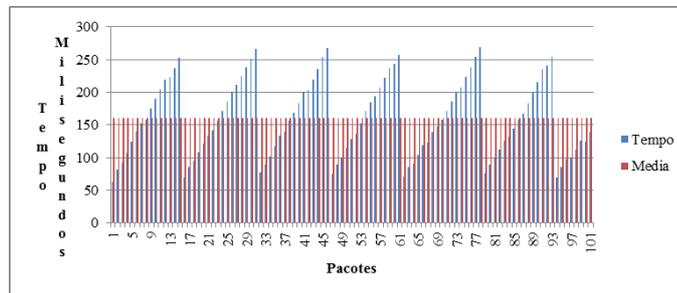
Cenário 4 - Transmissão de uma imagem bitmap de 20.000 bytes. Medição de latência através da interface IEEE 802.15.4;

Cenário 5 - Transmissão de um vídeo com 57.324 bytes. Medição de Latência por pacote, bit e byte, assim como sua latência e quantidade de pacotes perdidos na transmissão através da interface IEEE 802.15.4 para uma distância de 1 metro;

Cenário 6 - Transmissão de vídeo 57.324 bytes. Medição de Latência por pacote, bit e byte, assim como sua latência e quantidade de pacotes perdidos na transmissão através da interface IEEE 802.15.4 para uma distância de 6 metros.

Nos Cenários 1 a 4 foram coletados os tempos de comunicação entre cliente e servidor. A topologia utilizada na comunicação foi ponto a ponto, onde os nós estavam a 10 metros de distância. O tempo de troca de mensagens da Figura 44 foi obtido através de uma coleta de 100 pacotes obtidos ao acaso na transmissão das informações. Assim, a estimativa do tempo médio de comunicação para este cenário foi de aproximadamente 161 milissegundos.

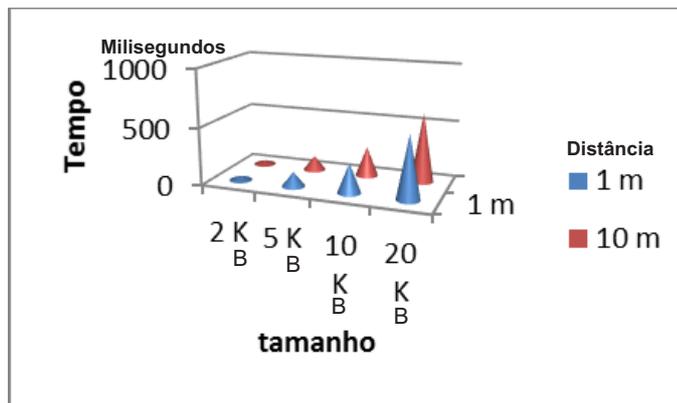
Figura 44 - Tempo de envio de pacotes.



Fonte: Elaborado pelo autor.

Também foi realizado um comparativo relativo à distância de transmissão, o qual demonstrou um aumento do tempo conforme o aumento da distância. Além disso, constatou uma quantidade significativa de pacotes perdidos ou corrompidos a partir de uma distância de 10 metros. Na Figura 45 ilustra-se um comparativo da transmissão de quatro arquivos de tamanho diferentes, sendo transmitidos a duas distâncias, 1 metro e 10 metros. Observa-se que conforme aumenta o tamanho do arquivo e a distância, o tempo de transmissão aumenta consideravelmente e a perda de pacotes se torna constante.

Figura 45 - Comparação de imagem enviada por diferentes distâncias.



Fonte: Elaborado pelo autor.

No Cenário 5 para analisar os dados da transmissão de vídeo, apresenta-se o tempo de envio dos pacotes, em bits e bytes assim, como a latência e quantidade de pacotes perdidos na transmissão. Foram realizadas três transmissões para medir e comparar os dados obtidos através do WireShark.

Na Figura 46 apresenta-se a estatística da rede IEEE 802.15.4 com a quantidade de 18.933 pacotes enviados e recebidos, sendo 9480 para o envio e 9480 para o ACK de confirmação. Na

transmissão desses pacotes, um total de 529 pacotes sofreram algum tipo de problema na transmissão. Nesta análise o tempo médio de comunicação de cada pacote foi de 0,059 segundos.

Figura 46 - Estatística de transmissão coletadas pelo WireShark.

Protocol	% Packets	Packets	% Bytes	Bytes	Mbit/s	End Packets	End Bytes	End Mbit/s
Frame	100,00 %	18933	100,00 %	355966	0,005	0	0	0,000
IEEE 802.15.4 Low-Rate Wireless PAN	100,00 %	18933	100,00 %	355966	0,005	9453	47334	0,001
ZigBee Network Layer	50,17 %	9480	30,20 %	308632	0,004	52	1737	0,000
ZigBee Application Support Layer	2,96 %	560	4,48 %	15934	0,000	29	868	0,000
Malformed Packet	2,79 %	529	4,22 %	15007	0,000	529	15007	0,000
Short Frame	0,01 %	2	0,02 %	59	0,000	2	59	0,000
Malformed Packet	30,48 %	5770	52,33 %	188057	0,003	5770	188057	0,003
Data	15,56 %	2946	27,47 %	97801	0,001	2946	97801	0,001
Short Frame	0,80 %	152	1,43 %	5103	0,000	152	5103	0,000

Fonte: Elaborado pelo autor.

Na Figura 47 apresenta-se a quantidade de bytes enviados pelo padrão, iniciando a transmissão na data de 13 de julho de 2015 as 11h49min46s finalizando as 11h59min13s, num total de 9 minutos e 27 segundos de transmissão, sendo enviados 18.933 pacotes com o tamanho de 18 bytes.

Figura 47 - Dados coletados pelo WireShark.

Traffic	Captured	Displayed	Marked
Packets	18933	18933	0
Between first and last packet	567,000 sec		
Avg. packets/sec	33,392		
Avg. packet size	18,801 bytes		
Bytes	355966		
Avg. bytes/sec	627,806		
Avg. MBit/sec	0,005		

Fonte: Elaborado pelo autor.

Na Figura 48 apresenta-se o tamanho em bytes de cada pacote, sendo 9.451 pacotes de 0 até 19 bytes e 9.482 pacotes de 20 até 39 bytes.

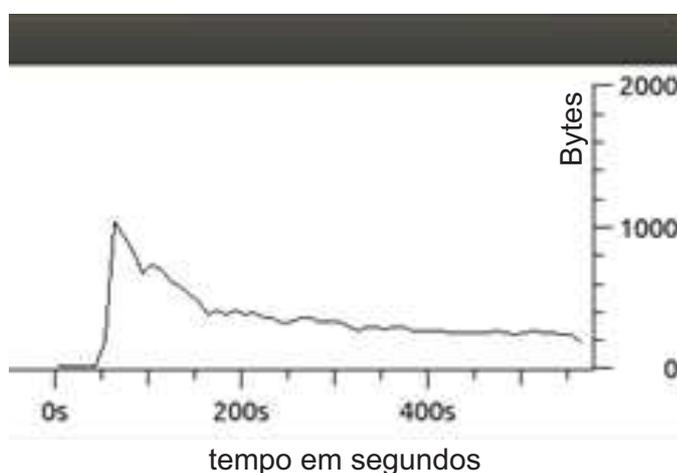
Figura 48 - Dados coletados pelo WireShark.

Topic / Item	Count	Rate (ms)	Percent
▼ Packet Lengths	18933	0,033392	
0-19	9451	0,016668	49,92%
20-39	9482	0,016723	50,08%
40-79	0	0,000000	0,00%
80-159	0	0,000000	0,00%
160-319	0	0,000000	0,00%
320-639	0	0,000000	0,00%
640-1279	0	0,000000	0,00%
1280-2559	0	0,000000	0,00%
2560-5119	0	0,000000	0,00%
5120-	0	0,000000	0,00%

Fonte: Elaborado pelo autor.

Na Figura 49 apresenta o gráfico de bits enviados por segundos na transmissão utilizando um intervalo de 10 segundos no gráfico.

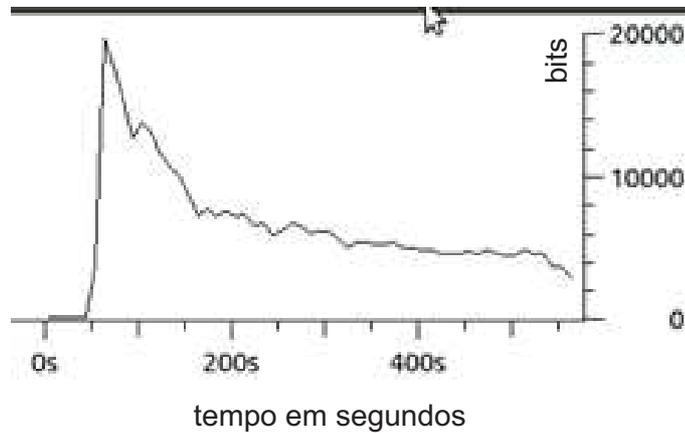
Figura 49 - Dados coletados pelo WireShark.



Fonte: Elaborado pelo autor.

Na Figura 50 apresenta o gráfico de pacotes enviados por segundos na transmissão.

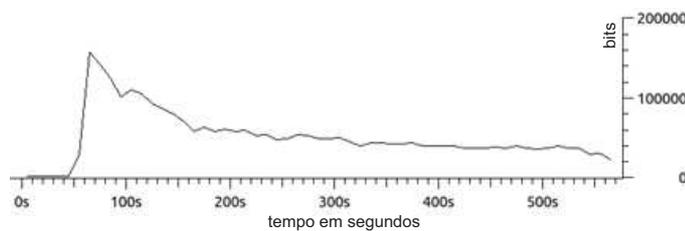
Figura 50 - Dados coletados pelo WireShark.



Fonte: Elaborado pelo autor.

Na Figura 51 apresenta a quantidade de bytes enviados por segundos na transmissão, demonstrando o pico de envio de pacotes na transmissão.

Figura 51 - Dados coletados pelo WireShark.



Fonte: Elaborado pelo autor.

O Cenário 6 é realizado a análise com transmissão a distância de 6 metros, conforme demonstra a figura 52, onde o Tempo Médio de Envio: 0,0666313 segundos no qual comprova a quantidade de pacotes e Bytes enviados e recebidos, assim como as possíveis perdas de pacotes.

Figura 52 - Dados coletados pelo WireShark.

Protocol	% Packets	Packets	% Bytes	Bytes	Mbit/s
▼ Frame	100,00 %	18859	100,00 %	354121	0,005
▼ IEEE 802.15.4 Low-Rate Wireless PAN	100,00 %	18859	100,00 %	354121	0,005
▼ ZigBee Network Layer	49,98 %	9425	86,66 %	306882	0,004
▼ ZigBee Application Support Layer	2,91 %	549	4,41 %	15631	0,000
Malformed Packet	2,75 %	518	4,15 %	14704	0,000
Short Frame	0,01 %	2	0,02 %	59	0,000
Malformed Packet	30,42 %	5736	52,79 %	186937	0,002
Data	15,58 %	2938	27,55 %	97546	0,001
Short Frame	0,80 %	151	1,43 %	5067	0,000

Fonte: Elaborado pelo autor.

7 CONCLUSÕES

Com a grande utilização de redes sem fios, o crescimento de altas tecnologias e a necessidade de baixos custos, surge um novo padrão o qual tem a proposta de utilização dentro de uma WPAN utilizando pouca energia, em muitos casos sendo possível ficar até seis meses com a mesma carga, este padrão é o IEEE 802.15.4 / ZigBee.

A necessidade de monitoramento de ambientes hostis, onde o homem muitas das vezes não possui acesso ou que poderia colocar a sua vida em risco, se faz necessário a cada dia. A diminuição do tamanho dos circuitos integrados e equipamentos, taxas de transferência e capacidade de armazenamento faz com que novas pesquisas busquem a otimização dos métodos atuais para melhor a eficácia de suas transmissões.

Como primeira contribuição deste trabalho, através da implementação em laboratório de software de compressão de imagens, com perda de qualidade, desenvolvido na linguagem Java, foi possível constatar que houve uma redução significativa no tamanho de arquivos. Em testes realizados, reduziu-se o tamanho de 2.084 kB para 19,3 kB, proporcionando assim um envio de vídeo mais rápido por uma rede sem fio padrão IEEE 802.15.4. Em arquivos de vídeo testou-se o software livre *Atube Catcher*, o qual obteve redução de até 41 vezes o seu tamanho original. Sendo assim, tem-se uma alternativa para contornar uma das limitações do padrão ZigBee que é sua taxa de transferência e tamanho do campo de dados.

A segunda contribuição deste trabalho vem através de testes realizados em laboratórios no qual pode-se verificar através da interconexão (ponto a ponto) de nós na rede por meio do padrão ZigBee, que a troca de informações (mensagens na rede) entre coordenador (mestre) e dispositivos finais - *End-Device* (Escravo) foi satisfatória a uma distância de até 10 metros. Conforme os testes realizados com distâncias superiores a 10 metros o tempo de trocas de mensagens aumenta de acordo com a elevação do tamanho de arquivo transmitido e também pela diminuição da taxa de transmissão dos pacotes, aumentando assim, a quantidade de quadros perdidos durante a comunicação entre os nós.

A disponibilidade de câmeras de baixo custo tem avançado o desenvolvimento de redes de sensores sem fio multimídia, ou seja, redes sem fio interligadas a dispositivos que são capazes de transmitir conteúdos multimídia, tais como transmissões de vídeo e imagens fixas. Apesar da redes de sensores não poderem ser usado para a transmissão de vídeo de alta qualidade, eles podem ser usados em aplicações de vigilância e de monitorização que exigem uma qualidade de vídeo mínima.

Com isso, pode se concluir que utilizando o padrão IEEE 802.15.4 é possível a transmissão de imagens/vídeo desde que sejam utilizadas técnicas para a compressão dos arquivos e que estes apresentem baixa resolução.

Em relação a trabalhos futuros, pode-se trabalhar dentro de uma aplicabilidade específica. Os resultados deste trabalho podem ser analisados em uma rede de monitoramento tais como: interferência externa, segurança, obstáculos ao sinal de radiofrequência, roteamento com um número maior de nós e consumo de energia.

REFERÊNCIAS

- ADOBE. *Photoshop*. San Jose, 2014. Disponível em: <<https://www.adobe.com/br/products/photoshop.html?promoid=KLXLS>>. Acesso em: 07 jul. 2014.
- ALLIANCE, Z. *ZigBee Alliance*. [S.l.], 2013. Disponível em: <www.ZigBee.org/Specifications/ZigBee/Overview.aspx>. Acesso em: 07 fev. 2014.
- ATMEL. *Atmel products studio*. [S.l.], 2014. Disponível em: <<http://www.atmel.com/tools/atmelstudio.aspx>>. Acesso em: 03 mar. 2014.
- ATUBE CATCH. [S.l.], 2014. Disponível em: <<http://www.atube.me/video>>. Acesso em: 10 jul. 2015.
- CORNELL GARY; HORSTMANN, C. S. *Core java vol. 1 - fundamentos*. 8. ed. [S.l.: s.n.].
- DEITEL, H. D. P. *Java como programar*. 8. ed. USA: 2010, 2010.
- FARAHANI, S. *ZigBee wireless networks and transceivers*. [S.l.]: newnes, 2011.
- FILHO, O. M.; NETO, H. V. *Processamento digital de imagens*. Sao Paulo: Brasport, 1999.
- GISLASON, D. *ZigBee wireless networking*. [S.l.]: Newnes, 2008.
- GONCALVES, E. *Desenvolvendo aplicações web com netbeans ide 6*. Rio de Janeiro: Ciência Moderna, 2008.
- GONZALEZ, R. C.; WOODS, R. E. *Processamento de imagens digitais*. [S.l.]: Edgard Blucher, 2000.
- GREBENNIKOV, A. *RF and microwave power amplifier design*. [S.l.]: McGraw-Hill New-York, 2005.
- GURSES, E.; AKAN, O. B. Multimedia communication in wireless sensor networks. In: SPRINGER. *Annales des Telecommunications*. [S.l.], 2005. v. 60, n. 7-8, p. 872–900.
- HEINZELMAN, W. R.; CHANDRAKASAN, A.; BALAKRISHNAN, H. Energy-efficient communication protocol for wireless microsensor networks. In: IEEE. *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*. California, 2000. p. 10–pp.
- HOWITT, I.; GUTIERREZ, J. et al. Ieee 802.15. 4 low rate-wireless personal area network coexistence issues. In: IEEE. *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*. [S.l.], 2003. v. 3, p. 1481–1486.

- IRGAN, K.; ÜNSALAN, C.; BAYDERE, S. Low-cost prioritization of image blocks in wireless sensor networks for border surveillance. *Journal of Network and Computer Applications*, Elsevier, v. 38, p. 54–64, 2014.
- JUCÁ, S. C. S.; PEREIRA, R. I.; VASCONCELOS, M. W. M. Sanusbee: ferramenta para gravação wireless de microcontroladores via bluetooth e zigbee. In: *VII CONNEPI-Congresso Norte Nordeste de Pesquisa e Inovação*. [S.l.: s.n.], 2012.
- KUBAN, P. A. *An architecture for the extension of fixed controller area networks to iee 802.15. 4 wireless personal area networks*. [S.l.]: University of Louisville, 2006.
- KUROSE, J. F.; ROSS, K. W.; MARQUES, A. S.; ZUCCHI, W. L. *Redes de computadores ea Internet: Uma abordagem top-down*. [S.l.]: Pearson, 2010.
- LABIOD, H.; AFIFI, H.; SANTIS, C. D. *WI-FITM, bluetoothTM, zigbeeTM and wiMAXTM*. [S.l.]: Springer Science & Business Media, 2007.
- LECUIRE, V.; DURAN-FAUNDEZ, C.; KROMMENACKER, N. Energy-efficient transmission of wavelet-based images in wireless sensor networks. *Journal on Image and Video Processing*, Hindawi Publishing Corp., v. 2007, n. 1, p. 15–15, 2007.
- MESSIAS, A. R. Controle remoto e aquisição de dados via xbee/zigbee. *Dispositivo de comunicação sem fio. Disponível em: <http://www.rogercom.com/ZigBee/ZigBeePag03.htm>*. Acesso em, v. 14, 2009.
- MUSTAFA, M. A.; ELAHI, M. M.; HOSSAIN, M. A.; ISLAM, M. M. Comparative performance analysis of mpeg4, flv and 3gp multimedia file formats using wireless network parameters. In: *IEEE. Computer and Information Technology (ICCIT), 2010 13th International Conference on*. [S.l.], 2010. p. 306–311.
- NETBEANS. *Atmel products AVR RAVEN*. California, 2013. Disponível em: <www.netbeans.org/features/java/>. Acesso em: 08 mar. 2014.
- NISHAD, P.; CHEZIAN, R. M. Behavioral study of data structures on lempel ziv welch (lzw) data compression algorithm and its computational complexity. In: *IEEE. Intelligent Computing Applications (ICICA), 2014 International Conference on*. [S.l.], 2014. p. 268–274.
- ORACLE. *Atmel products AVR RAVEN*. [S.l.], 2013. Disponível em: <http://www.oracle.com/lad/corporate/press/press-release-ladapr20-09-333619-esa.html>. Acesso em: 01 mar. 2014.
- PAULISTA, U. E. *Manual de identidade visual*. [S.l.], 2014. Disponível em: <unesp.br/acises/conteudo.php?conteudo=1361>. Acesso em: 10 jan. 2014.
- PEKHTERYEV, G.; SAHINOGLU, Z.; ORLIK, P.; BHATTI, G. Image transmission over iee 802.15. 4 and zigbee networks. In: *IEEE. Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*. [S.l.], 2005. p. 3539–3542.
- PMISA, P. M. d. I. S. *Foto Cidade Ilha Solteira SP*. Ilha Solteira - SP, 2013. Disponível em: <www.ilhasolteira.sp.gov.br>. Acesso em: 03 mar. 2013.

SWAMINATHA, T. M.; ELDEN, C. R. *Wireless security and privacy: best practices and design techniques*. Boston: Addison-Wesley Longman Publish Co., Inc., 2002.

THIRD generation partnership. 3GPP. London, 2015. Disponível em: <<http://www.3gpp.org/>>. Acesso em: 22 jun. 2015.

WANG, S.; XU, D.; YAN, S. Analysis and application of wireshark in tcp/ip protocol teaching. In: IEEE. *E-Health Networking, Digital Ecosystems and Technologies (EDT), 2010 International Conference on*. [S.l.], 2010. v. 2, p. 269–272.

WIRESHARK. *WIRESHARK*. Mountain View, CA, 2015. Disponível em: <<https://www.wireshark.org/>>. Acesso em: 15 mar. 2015.

ZAINALDIN, A.; LAMBADARIS, I.; NANDY, B. Video over wireless zigbee networks: multi-channel multi-radio approach. In: IEEE. *Wireless Communications and Mobile Computing Conference, 2008. IWCMC'08. International*. Paphos, 2008. p. 882–887.

APÊNDICE A - CRIAÇÃO DO ARQUIVO DE GRAVAÇÃO DO MAC NA EEPROM DO ATMEGA1281

Para a implementação da comunicação sem fio utilizando o dispositivo RZ502 da ATMEL é necessário que seja gravado na memória EEPROM do microcontrolador ATMEGA1281 o número do MAC do rádio transceptor. Cada rádio possui uma identificação que segue um padrão da ATMEL. Os rádios que foram utilizados neste trabalho possuem os seguintes números de identificação 0030.3990 e 0030.4007. Para a gravação deste endereço na EEPROM do microcontrolador deve-se primeiro criar um arquivo de extensão .hex com o número do MAC a ser gravado. Como é ilustrado na Figura 53, tem-se o comando para a criação do arquivo .hex. Com a aplicação HexMaker consegue-se a construção do arquivo nodea.hex que contém o MAC do rádio. Cabe salientar a necessidade da inversão da sequência dos números que se encontram padronizados na antena do rádio. Em uma das antenas, possui a seguinte identificação, 0030.3990. Para que seja gravado corretamente no EEPROM do microcontrolador deve-se proceder da seguinte forma: 90-39-30-FF-FF-25-04-00, pois o endereço MAC deve ser armazenado nos primeiros 8 *bytes* da memória EEPROM. A sequência FF-FF-25-04-00 é definida de acordo com a norma da ATMEL.

Figura 53 - Comando para a criação do arquivo .hex para a gravação do MAC no EEPROM do microcontrolador ATMEGA1281.



```
C:\> Prompt de comando
C:\avr414\bin>HEXMaker.exe nodea.hex 903930ffff250400
```

Fonte: Elaborado pelo autor.

Na Figura 39 é ilustrada a execução do comando evidenciado na Figura 54. Observa-se que o arquivo nodea.hex foi criado com o seu respectivo endereço MAC.

Figura 54 - Execução do comando de criação do arquivo com o endereço MAC.

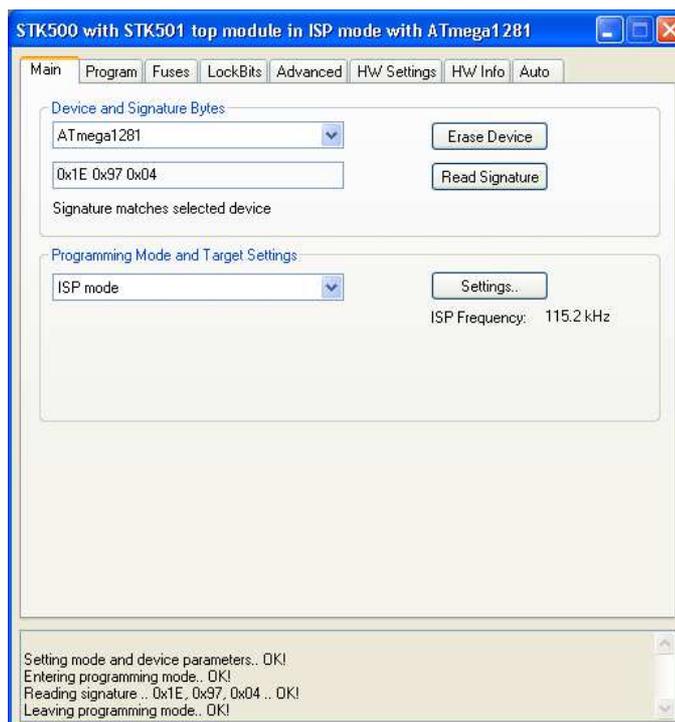


Fonte: Elaborado pelo autor.

A.1 CONFIGURAÇÃO DO AMBIENTE AVR STUDIO PARA A GRAVAÇÃO COM ARQUIVOS NO MICROCONTROLADOR

Para a gravação dos arquivos no microcontrolador ATMEGA1281 primeiramente deve-se configurar o AVR Studio para o reconhecimento correto do CI utilizado. No momento da seleção da aplicação para a gravação no CI deve-se informar corretamente o tipo de microcontrolador a ser utilizado. A seguir mostra-se na Figura 55 a interface de configuração definida neste trabalho ATmega1281. Na aba *Main* e no campo *Device and Signature Bytes*, seleciona-se o microcontrolador correto.

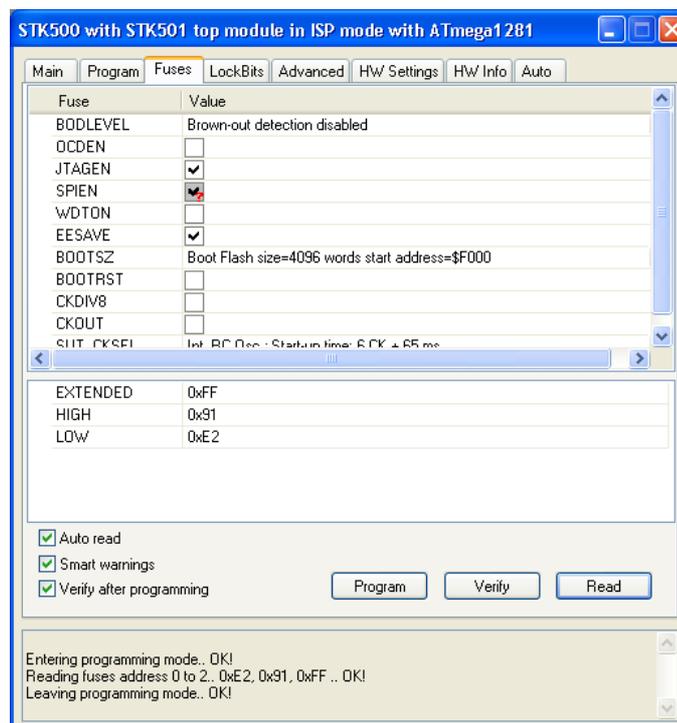
Figura 55 - Formulário de seleção do dispositivo a ser gravado pelo AVR Studio.



Fonte: Elaborado pelo autor.

Na Figura 56 mostra-se a configuração do padrão após o reconhecimento do dispositivo pela IDE AVR Studio. Na aba *Fuses* da janela de configuração do dispositivo devem ser realizadas algumas modificações conforme a aplicação a ser elaborada. Neste passo é importante ressaltar que a divisão do *clock* deve ser desativada (*CHKDIV8=disabled*). A configuração correta do dispositivo no AVR Studio propicia a execução da aplicação mais facilmente e sem possíveis transtornos.

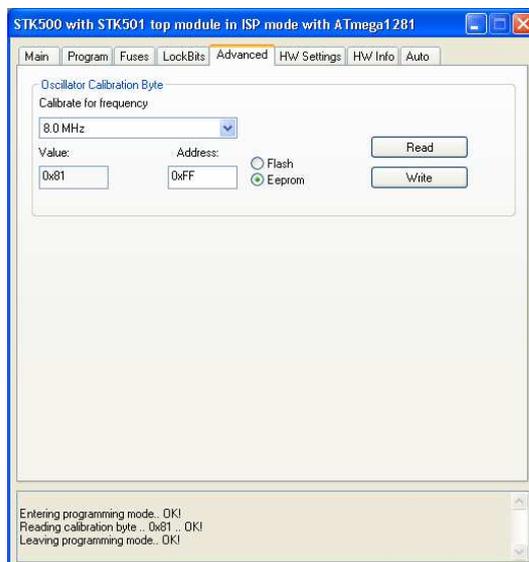
Figura 56 - Configuração do dispositivo conforme trabalho a ser realizado.



Fonte: Elaborado pelo autor.

Como ilustrado na Figura 57, na aba *Advanced* é configurado a frequência do microcontrolador a ser utilizado. O ATMEGA1281 trabalha com uma frequência de 8 MHz. Para que seja configurada a frequência padrão, primeiro deve-se ser realizada a leitura do dispositivo (*read*) e posteriormente configurado o valor encontrado utilizando o comando de escrita (*write*).

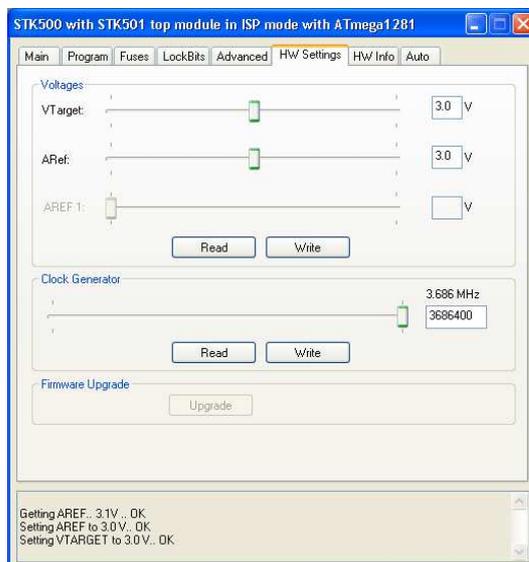
Figura 57 - Definição do oscilador de frequência do dispositivo.



Fonte: Elaborado pelo autor.

Na Figura 58 especifica-se os limites de tensão do ATMEGA1281. É importante salientar que para a proteção dos dispositivos as tensões tanto do VTarget quanto a do ARef devem estar configuradas em 3,0. Depois de configurada a tensão, é importante gravá-la para que o processo de configuração tenha sucesso. Além da tensão, também é importante realizar a leitura e gravação da frequência do *clock*. Este processo é realizado automaticamente pelo ambiente de desenvolvimento.

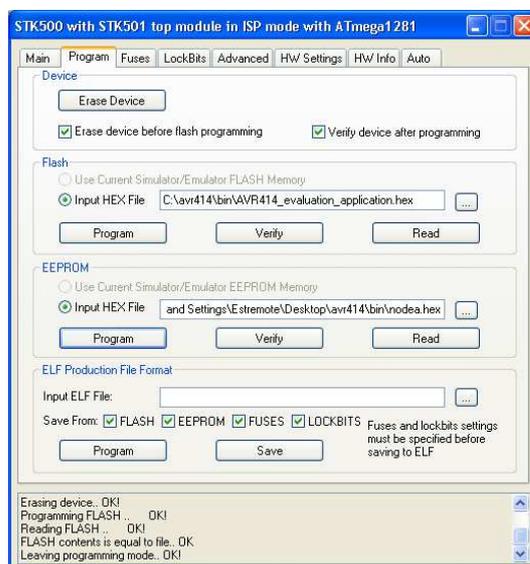
Figura 58 - Configuração da Tensão do Dispositivo e Frequência do Clock.



Fonte: Elaborado pelo autor.

Na Figura 59 é ilustrado o processo de programação do dispositivo. Na aba *Program* é permitido realizar tanto o teste e verificação dos programas a serem gravados, quanto à limpeza dos arquivos programados anteriormente no microcontrolador. No campo Flash podem ser carregados os algoritmos implementados na linguagem C utilizando a ferramenta WinAVR do AVR Studio e compilados na forma de arquivos .hex. A programação deste arquivo no microcontrolador se dá através da conexão da porta RS232 do computador, a porta RS232 do dispositivo de desenvolvimento STK500 da ATMEL. É importante ressaltar que a gravação do ATMEGA1281 é feita no módulo de expansão STK501 acoplado ao STK500.

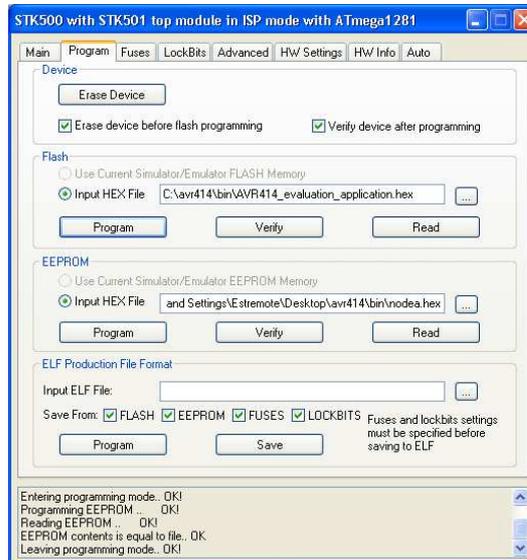
Figura 59 - Gravação do arquivo .hex na memória Flash do ATMEGA1281



Fonte: Elaborado pelo autor.

Na Figura 60 apresenta-se o processo de gravação da memória EEPROM do microcontrolador. Neste caso, o que será armazenado na memória EEPROM é o endereço MAC da antena do dispositivo RZ502, processo este, relatado na seção 4.1 deste capítulo.

Figura 60 - Processo de gravação na EEPROM do ATMEGA1281.



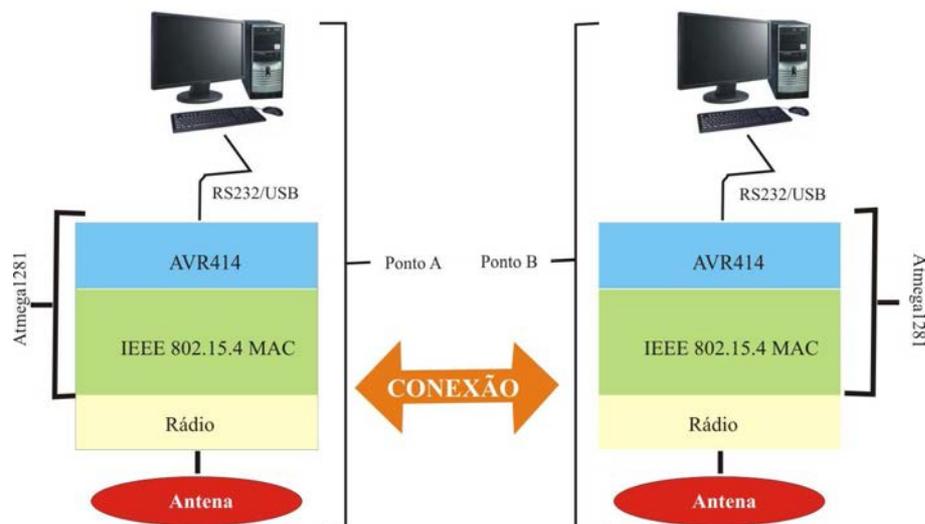
Fonte: Elaborado pelo Autor

APÊNDICE B - CONFIGURAÇÃO DA COMUNICAÇÃO SEM FIO (IEEE 802.15.4) VIA TEXTO

Para a realização dos testes de comunicação entre dois dispositivos utilizando o padrão IEEE 802.15.4 foi utilizada uma aplicação Java com comunicação serial.

Na Figura 61 é mostrado o esquemático do sistema desenvolvido. Este aplicativo é usado para enviar e receber mensagens em formato de texto, na estrutura de redes ponto a ponto, onde são utilizados dois módulos de comunicação, a antena RZ502 da Atmel é utilizada para enviar e receber mensagens.

Figura 61 - Visão geral da aplicação utilizando o microcontrolador Atmega1281.



Fonte: Elaborado pelo Autor

Para inicializar a aplicação, basta abrir o arquivo chamado AVR414.jar que encontra-se na pasta bin. A interface apresenta na figura 62 consta a opção CONECTAR, o qual faz a conexão com a porta de comunicação COM.

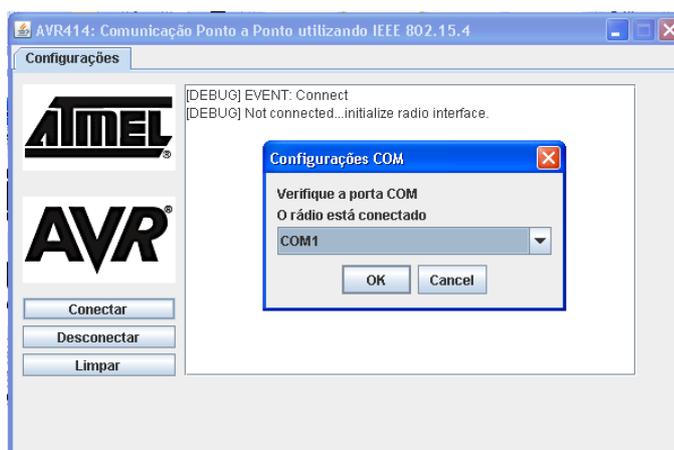
Figura 62 - Tela de início da Aplicação.



Fonte: Elaborado pelo Autor

Na sequência deve-se marcar a COM utilizada na aplicação. No projeto vamos marcar-se a COM1, onde conecta-se a porta serial da placa STK501, e o botão OK, como apresentada na Figura 63.

Figura 63 - Tela de configuração da COM.



Fonte: Elaborado pelo Autor

Na Figura 64 ilustra-se o canal que deveria ser trabalhado. O canal utilizado da aplicação, será o canal 11. Para configurar o tipo de dispositivo utilizado, quando se inicia uma aplicação, deve-se primeiro marcar a opção coordinator em seguida acionar o botão configure.

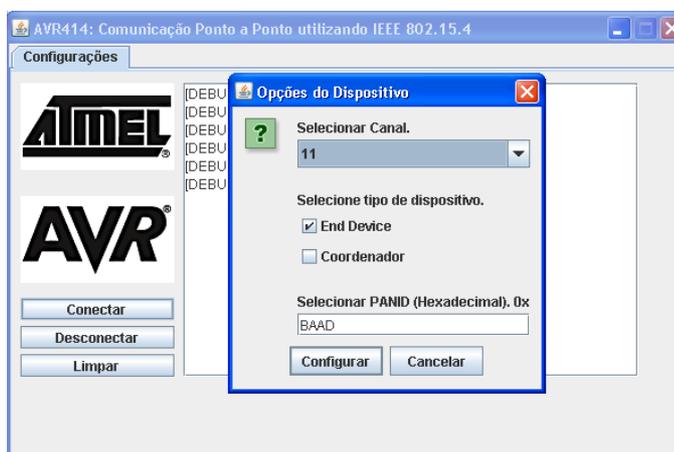
Figura 64 - Tela de configuração *Coordinator*.



Fonte: Elaborado pelo Autor

Na sequência, devem-se repetir os passos das Figuras 65 e 66, conforme apresentada na Figura 67, deve-se marcar o canal utilizado na aplicação, canal 11. Após, marcar o *End Device* como tipo de dispositivo, desta forma será realizada uma conexão coordenador com *End Device*. O PANID (hexadecimal) continuam o padrão, BAAD, para finalizar, deve-se acionar o botão Configurar.

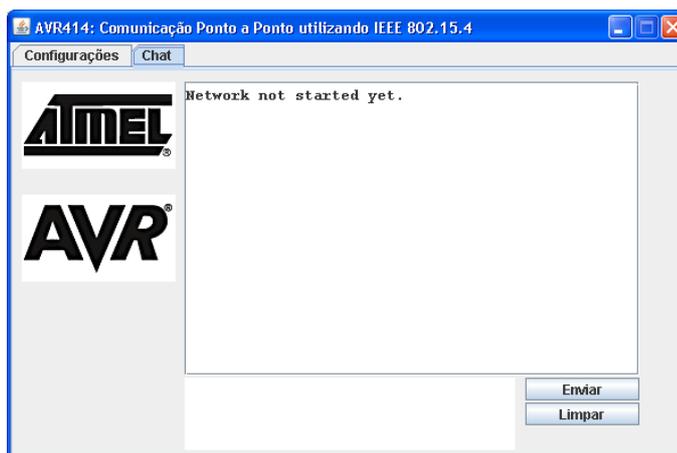
Figura 65 - Tela configuração da COM.



Fonte: Elaborado pelo Autor

Na Figura 66, apresenta-se a mensagem ao usuário quando o coordenador está aguardando a conexão com o *End Device*. Após a conexão com o *End Device*, uma nova aba será adicionada à aplicação para realizar os testes de conexão.

Figura 66 - Tela de interface da comunicação ponto a ponto utilizando o padrão IEE 802.15.4.



Fonte: Elaborado pelo Autor

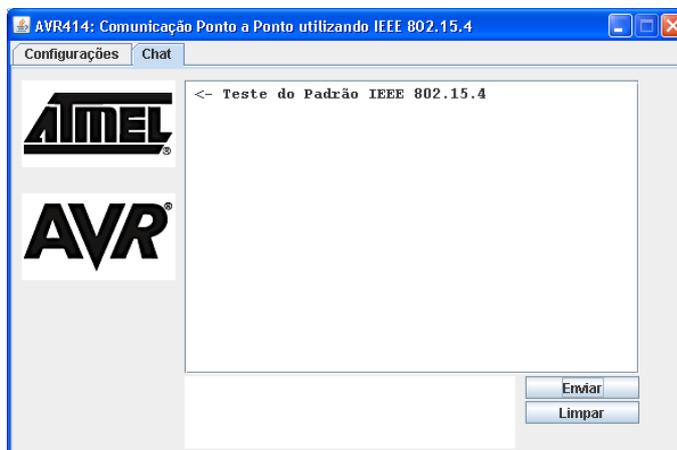
Na Figura 67, ilustra-se o envio da mensagem teste do padrão IEEE 802.15.4 para o *End Device*, e a Figura 68 confirma-se o recebimento da mesma.

Figura 67 - Tela da Interface da aplicação ponto a xonto enviando uma mensagem ao *End Device*.



Fonte: Elaborado pelo Autor

Figura 68 - Tela da Interface da aplicação ponto a ponto recebendo uma mensagem do Coordenador.



Fonte: Elaborado pelo Autor

Na Figura 69 a aplicação *End Device* mostra o recebimento da mensagem anterior ("teste do padrão IEEE 802.15.4") e envia a mensagem "*Msg End Device -> Coordenador*", para o Coordenador.

Figura 69 - Tela da Interface da Aplicação End Device enviando uma mensagem para o Coordenador.



Fonte: Elaborado pelo Autor

Na Figura 70 ilustra-se a confirmação da mensagem "*Msg End Device -> Coordenador*", recebida pelo coordenador.

Figura 70 - Tela da Interface da Aplicação - Coordenador recebe a mensagem do End Device.



Fonte: Elaborado pelo Autor

Para encerrar a conexão entre os módulos, basta voltar para a aba configurações e pressionar o botão "desconectar", assim finaliza-se a aplicação e também a conexão.