



TRABALHO DE CONCLUSÃO DE CURSO

**ANÁLISE EXPERIMENTAL DE ASPECTOS DO
PROTOCOLO DyTEE QUANTO À EFICIÊNCIA
ENERGÉTICA EM COMPARAÇÃO AO PADRÃO
802.15.4 (MAC)**

**Línercky André César Soares Lustosa
Renan Barbosa da Silva**

Brasília, julho de 2015

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

ANÁLISE EXPERIMENTAL DE ASPECTOS DO PROTOCOLO DyTEE QUANTO À EFICIÊNCIA ENERGÉTICA EM COMPARAÇÃO AO PADRÃO IEEE 802.15.4 (MAC)

Línercky André César Soares Lustosa
Renan Barbosa da Silva

Relatório submetido como requisito parcial para obtenção
do grau de Engenheiro Eletricista

Banca Examinadora

Prof. Adolfo Bauchspiess, UnB/ Dep de Engenharia
Elétrica

Prof. Adoniran Judson de Barros Braga, UnB/ Dep
de Engenharia Elétrica

Prof. Lélío Ribeiro Soares Júnior, UnB/ Dep de
Engenharia Elétrica

Dedicatórias

*Dedico este trabalho à minha mãe
Andréia César Soares de Oliveira.*

Línercky André César Soares Lustosa

*Dedico este trabalho a minha família, aos
meu companheiros de Elétrica, aos meus
irmão de Comunidade, aos meus colegas de
trabalho e a minha namorada Renata*

Renan Barbosa da Silva

Agradecimentos

Agradeço em primeiro lugar a Deus que me sustentou até aqui. Também a minha mãe por ter me proporcionado toda a estrutura necessária e que foi fundamental na realização desta grande etapa da minha vida. Aos meus amigos que sofreram comigo em todos os momentos unindo forças durante todo este processo.

Aos meus familiares pelo apoio e torcida. Também à minha namorada por ter paciência para suportar a minha ausência.

Também ao doutorando Vinícius Galvão Guimarães pela disponibilidade, paciência, atenção e suporte essenciais à realização deste trabalho.

Línercky André César Soares Lustosa

Agradeço a Deus por toda força e persistência que ele me concedeu e pela vida. Aos meus pais por todo carinho. Ao meu padrasto e minha madrasta pelo apoio. Aos meus avós e tios pela oportunidade.

Agradeço toda a dedicação e empenho do doutorando Vinícius Galvão Guimarães durante a realização deste trabalho.

Renan Barbosa da Silva

RESUMO

O presente trabalho tem por objetivo implementar uma rede *wireless* de automação predial, operando em controle de temperatura, para a verificação da eficiência energética do DyTEE, um protocolo de acesso ao meio (MAC) desenvolvido para redes de automação, em comparação ao protocolo IEEE 802.15.4 MAC no que se refere a economia de energia.

ABSTRACT

This project aim to implement a *wireless* network for building automation. The system is used in a control of temperature context in order to verify the power efficiency of the medium access control (MAC) protocol DyTEE, developed for automation networks. Additoinally, a power saving comparison is made between DyTEE and IEEE 802.15.4 MAC protocols.

SUMÁRIO

1 INTRODUÇÃO.....	1
1.1 MOTIVAÇÃO.....	1
1.2 JUSTIFICATIVA.....	2
1.3 OBJETIVOS.....	2
2 FUNDAMENTAÇÃO TEÓRICA.....	3
2.1 REDES <i>WIRELESS</i> DE SENSORES E ATUADORES.....	3
2.1.1 CAUSAS DE DESPERDÍCIO DE ENERGIA EM WSANS.....	4
2.3 TOPOLOGIAS DE REDES.....	5
2.3.1 PONTO A PONTO (P2P).....	5
2.3.2 MALHA (MESH).....	6
2.3.3 TOPOLOGIA ESTRELA.....	6
2.3.4 ÁRVORE OU ESTRELA HIERARQUICA (CLUSTER TREE).....	7
2.3.5 TOPOLOGIA LINEAR.....	7
2.4 PROTOCOLOS DE COMUNICAÇÃO.....	8
2.4.1 CSMA-CA.....	8
2.4.2 IEEE 802.15.4.....	9
2.4.3 TDMA.....	10
2.4.4 S-MAC.....	11
2.4.4.1 CICLO DE TRABALHO PARA O PROTOCOLO S-MAC.....	12
2.4.4.2 ESCOLHA DE PROGRAMAÇÃO POR UM NÓ.....	13
2.4.4.3 ESTABILIDADE DA REDE.....	14
3 PROTOCOLOS IEEE 802.15.4 E DyTEE.....	17
3.1 CAMADA FÍSICA.....	18
3.2 CAMADA MAC.....	20
3.2.1 DyTEE.....	21
3.2.1.1 PARÂMETROS CONFIGURÁVEIS DO DyTEE.....	21
3.2.1.2 INÍCIO DE UMA REDE NO DyTEE.....	23
3.2.1.3 ESTRUTURA GERAL DO PROTOCOLO.....	23
3.2.1.4 TESTES ANTERIORES.....	26
4 ESPECIFICAÇÕES DO PROJETO.....	32
4.1 A REDE IMPLEMENTADA.....	32
4.1.2 METODOLOGIA DE CRIAÇÃO DA REDE.....	32
4.1.3 DECLARAÇÃO DAS PLATAFORMAS UTILIZADAS EM CADA PROTOCOLO.....	33
4.1.4 ANÁLISE DE CENÁRIOS ENCONTRADOS DURANTE A CRIAÇÃO DAS REDES NOS DOIS PROTOCOLOS.....	33
4.2 MATERIAIS UTILIZADOS.....	34
4.2.1 PROTÓTIPO DE SISTEMA TÉRMICO.....	34
4.2.2 PLACAS.....	34
4.2.3 SENSOR DE TEMPERATURA LM35.....	36
4.2.4 SAINSMART 8 CHANNEL DC 5V RELAY MODULE.....	37
4.2.5 OSCILOSCÓPIO DIGITAL.....	37
4.2.5.1 OSCILOSCÓPIO/ ANALISADOR DE ESPECTRO.....	38
4.2.5.2 DATA RECORDER (REGISTRADOR DE DADOS).....	38
4.2.5.3 ESQUEMÁTICO DE LIGAÇÃO DO OSCILOSCÓPIO.....	39
4.2.6 MULTÍMETRO DIGITAL TRUE RMS 87-5- FLUKE.....	39
5 RESULTADOS.....	41
5.1 CORRENTE UTILIZANDO O DyTEE.....	41
5.2 TENSÃO E POTÊNCIA.....	44
5.3 CORRENTE MEDIDA UTILIZANDO O PROTOCOLO ZIGBEE.....	45
5.4 POTÊNCIA.....	45
5.5 PROCEDIMENTO PARA ESTIMAR A VIDA ÚTIL DAS PILHAS.....	46
6 CONCLUSÃO.....	48
6.1 TRABALHOS FUTUROS.....	48
REFERÊNCIAS BIBLIOGRÁFICAS.....	49

LISTA DE FIGURAS

2.1	Rede <i>Wireless</i> de sensores e atuadores	3
2.2	Topologia Ponto a Ponto	6
2.3	Topologia em malha	6
2.4	Topologia Estrela	7
2.5	Topologia em Árvore Hierárquica	7
2.6	Topologia Linear	8
2.7	Diagrama temporal CSMA-CA	9
2.8	alargamento espectral dos sinais transmitidos.....	9
2.9	Organização temporal TDMA	10
2.10	Diagrama de funcionamento do S-MAC	12
2.11	Nós vizinhos com programações diferentes	13
2.12	Relação de tempo entre um receptor e diferentes emissores	14
2.13	Os nós C e F dormem quando A transmite a B.....	15
3.1	Modulação DSSS	18
3.2	Alargamento espectral em DSSS	19
3.3	Bandas de Frequência	19
3.4	Módulo Zigbit	20
3.5	Diagrama temporal do protocolo DyTEE	23
3.6	Diagrama do período de sincronização (SyncM) do DyTEE	24
3.7	Fluxograma de operação para um nó que está executando o protocolo de DyTEE..	25
3.8	Vazão na rede simulada e real	27
3.9	Consumo total de energia da rede durante as transmissões	27
3.10	Consumo total de energia da rede durante os períodos osciosos	28
3.11	Consumo da rede durante as recepções	28
3.12	Consumo total de energia da rede para os períodos de dormência	29
3.13	Potência média consumida	29
3.14	Vazão obtida em ambos protocolos em módulos reais	30
3.15	Potência consumida em módulos reais	30
4.1	Sistema implementado	32
4.2	Protótipo de controle de temperatura	32
4.3	Placa ZigBit Channel 1.0 que exerce a função de Coordenador/Gravadora	35
4.4	Placa ZigBit Channel 1.0 com funcionalidade dispositivo final	35
4.5	Placa ZigBit Breakout	36
4.6	Sensor de temperatura LM35	36
4.7	Sainsmart 8 channel DC 5V relay module	37
4.8	Osciloscópio digital MDSO	38
4.9	Esquemático de ligação do Osciloscópio.....	39
4.10	Multímetro digital True RMS 87-5- Fluke.....	40
5.1	Corrente demandada pelo módulo	41
5.2	Comportamento da corrente ao se apertar o botão reset	41
5.3	Picos de Transmissões.....	42
5.4	Período durante a recepção do <i>beacon</i> e transmissão	43
5.5	Tensão e Corrente para módulos operando com DyTEE	44
5.6	Consumo de potência pelo nó folha.....	44
5.7	Corrente demandada pelo módulo utilizando o protocolo ZigBee.....	45
5.8	Potência consumida pelo nó sensor utilizando o protocolo ZigBee	46
5.9	Curva de vida útil de pilhas alcalinas AA	46
5.10	Curva estimada para vida útil das baterias.....	47

LISTA DE SÍMBOLOS

Acrônimos

UnB Universidade de Brasília

Siglas

GBC *Green Bulding Council Brasil*

WSANS *Wireless Sensor and Actor Networks*

MAC *Media Access Control*

TDMA *Time Division Multiple Access*

S-MAC *Sensor MAC*

CSMA/CA *Carrier Sense Multiple Access- collision avoidance*

IEEE *Institute of Electrical and Electronics Engineers*

ACK *Acknowledge*

P2P *Peer to Peer*

FFD *Full function device*

RFD *Reduced function Device*

IFS *Interframe Space*

DSSS *Direct Sequence Spread Spectrum*

SNR *Signal-to-noise ratio*

GTS *Guarantee Time Slots*

RTS *Request to Send*

CTS *Clear to Send*

PDU_s *Protocol Data Units*

USB *Universal Serial Bus*

OSI *Open system Interconnection*

1 INTRODUÇÃO

Este capítulo expõe a necessidade de avaliação do protocolo DyTEE em uma rede de automação para a comprovação da eficiência energética.

1.1 MOTIVAÇÃO

As edificações demandam significativas quantidades de energia, sendo responsáveis por sérios impactos ambientais. No Brasil, segundo dados do *Green Building Council Brasil* (GBC Brasil, 2011), as instalações prediais são responsáveis por 42% do total de energia elétrica utilizada. Sendo assim, as redes de automação predial constituem um meio eficaz para se otimizar a utilização desse recurso reduzindo significativamente o consumo de energia sem a inserção de prejuízos relevantes à qualidade do ambiente de trabalho das pessoas.

Devido aos prejuízos financeiros e o impacto causado, pelas edificações, à sustentabilidade essa área tem sido objeto de estudos e pesquisas, o que proporcionou evoluções consideráveis ao aprimoramento e acessibilidade dessas tecnologias.

O setor de automação predial e residencial tem se expandido rapidamente nos últimos anos se destinando a diversas aplicações, dentre elas segurança, conforto, entretenimento, sustentabilidade e economia de energia. A possibilidade de controle de diversas utilidades como temperatura, iluminação, funcionamento de aparelhos eletrônicos, irrigação, operação de motores, sistemas de áudio e vídeo etc. impacta de maneira positiva no estilo de vida dos usuários, trazendo bem estar e conforto, bem como garantindo eficiência energética aos ambientes equipados com essas tecnologias. Além disso, a flexibilidade dos sistemas de automação, proporciona uma adequação ideal às aplicações a que se destinam, tornando propícia sua implantação nas mais variadas e diversificadas situações. Em uma edificação inteligente, diversos sensores e atuadores são interconectados formando um sistema de controle, que pode fazer uso de estruturas cabeadas ou sem fio.

Em diversas aplicações a exigência de certa estrutura física, a demanda por manutenções periódicas e os elevados custos inviabilizam a utilização de redes cabeadas. Nesse contexto surgem então como solução mais viável as redes *wireless* de sensores e atuadores WSAWs (Do inglês *wireless Sensor and Actuators Networks*) para comunicação entre os componentes da rede. Tais redes, geralmente possuem nós sensores operados por baterias, o que confere maior flexibilidade de aplicações, pois possibilitam monitoramento nos locais mais remotos e facilitam a instalação dessas redes em edificações já concluídas sem a necessidade de reformas para sua implantação. Em contra partida, geralmente há uma grande dificuldade em carregar e recarregar as baterias sendo necessário fazer sua substituição, geralmente em curtos períodos de tempo, o que ao passar do tempo se torna um incômodo. Dessa forma, a eficiência energética é a principal restrição na implementação das WSAWs tornando-se importante área de estudo e pesquisa.

Assim, a partir dessa necessidade, desenvolveu-se na Universidade de Brasília o DyTEE (do inglês *Dynamic Timed Energy Efficient*), um protocolo de acesso ao meio (MAC), com foco em economia de energia que se baseia no protocolo TDMA- modulação por divisão de tempo com acesso múltiplo (do inglês *Time Division Multiple Access*) que opera com ciclos de ativação e vagas de transmissão dinâmicas, empregando somente a utilização de um canal e sensoriamento da portadora. Além disso, o DyTEE também se baseia nas melhores características de outros protocolos como S-MAC ou Sensor MAC; CSMA/CA acesso múltiplo com verificação de portadora com anulação/prevenção de colisões (do inglês *Carrier sense multiple access with collision avoidance*) e IEEE 802.15.4.

1.2 JUSTIFICATIVA

O protocolo de acesso ao meio DyTEE foi desenvolvido na Universidade de Brasília durante o trabalho de mestrado do aluno Vinicius Galvão Guimarães e propõe novas soluções para redes *wireless* de sensores empregadas para automação predial e rural, sobretudo no que se refere a economia de energia.

Entretanto esse protocolo ainda está em processo de teste não tendo ainda passado por todas as avaliações necessárias para que possa de fato ser validado e empregado comercialmente. Simulações realizadas anteriormente demonstraram a superioridade do DyTEE a protocolos existentes no mercado como por exemplo o S-MAC e o “IEEE 802.15.4” no que se refere a economia de energia, vazão, latência e cabeçalho empregado nas transmissões.

Em módulos reais, O DyTEE apenas havia sido comparado ao “IEEE 802.15.4”. O teste em módulos reais foi feito por meio de uma implementação genérica na qual a rede era criada e colocada em funcionamento seguindo a topologia linear. Nessa rede havia transmissões de mensagens, os nós se comunicavam, porém não se destinava a automação, de forma que não havia atuação. O DyTEE neste trabalho é colocado em operação pela primeira vez em uma rede de automação real.

1.3 OBJETIVOS

O presente trabalho tem por finalidade o emprego e teste do DyTEE em uma rede de automação predial para a verificação de sua adaptação a esses cenários de aplicação, bem como a comparação com resultados apresentados anteriormente no que se refere a economia de energia em comparação com o protocolo IEEE 802.15.4.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta o conteúdo introdutório que auxilia na compreensão do texto e fornece as bases para o projeto trabalhado.

2.1 REDES WIRELESS DE SENSORES E ATUADORES

As redes de comunicação sem fio proporcionam uma forma viável de controle em aplicações de automação predial. Nessas redes, sensores são dispostos ao longo dos ambientes para monitorar diversas variáveis como temperatura, iluminação, presença e umidade e então enviar as informações coletadas ou a uma central ou diretamente a atuadores que irão executar as atuações efetuando o controle dessas variáveis.

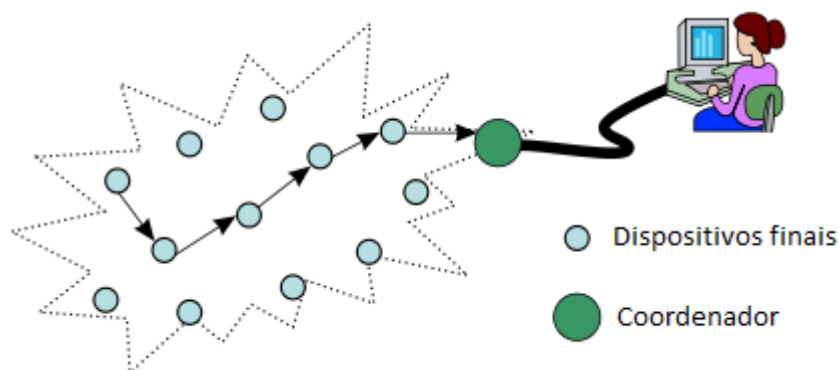


Figura 2.1- Rede *wireless* de sensores e atuadores. [22]

Os nós sensores e atuadores, em WSNs, são normalmente pequenos e possuem baixos custos financeiros. Geralmente esses dispositivos possuem módulos de comunicação, sensoriamento e processamento montados em placas de circuitos impressos, podendo ser organizados em grupos (clusters) onde pelo menos um dos sensores deve ser capaz de detectar um evento na região, processá-lo e tomar uma decisão de transmitir ou não os dados coletados para outros nós da rede. Os microcontroladores empregados são programáveis, o que configura versatilidade à sua utilização em diversas finalidades, de modo que um mesmo dispositivo pode executar tanto a função de sensoriamento, quanto atuação e coordenação da rede, dependendo da forma de programação e configuração utilizada. Para uma boa operação desses sistemas, faz-se necessário que esses dispositivos disponham de mecanismos para autoconfiguração e adaptação, para o caso de haver falhas na comunicação e perdas eventuais dos nós causadas por descarga das baterias.

As redes de controle que fazem uso de cabos comuns são menos versáteis, pois possuem custos elevados e exigem espaço e manutenção, se tornando muitas vezes inviáveis a diversas finalidades. Além disso, quando há a necessidade de sensoriamento em locais de difícil acesso, as redes *wireless* são mais viáveis pela facilidade de alocação dos nós, que não exigem demasiadas estruturas para atuarem em determinado local.

Apesar das vantagens proporcionadas pelas redes *wireless*, o armazenamento de energia é o principal problema encontrado nestes projetos, pois os nós geralmente são alimentados por baterias operando em locais onde não há nenhuma outra fonte de energia para alimentá-los. O problema relacionado à alimentação constitui uma restrição a essas redes e tem se tornando alvo de grande interesse por parte de pesquisadores em diversas regiões do

mundo. Dessa forma, busca-se um aprimoramento para que os sistemas possam operar com o mínimo de intervenção humana nos locais mais remotos e de difícil acesso. Uma WSN tende a ser autônoma e requer um alto grau de cooperação para executar tarefas definidas para a rede.

A particularidade das diversas aplicações possíveis trás a necessidade de adaptação dos algoritmos tradicionais como protocolos de comunicação e eleição de líder para melhor execução das tarefas requeridas. Dentre as principais características adaptáveis estão:

- Endereçamento dos nós sensores: característica que se refere à necessidade de identificação exata do local de coleta dos dados. Algumas aplicações exigem identificação precisa do local onde os dados são coletados, já em outras apenas a coleta de dados é suficiente para execução de controle.
- Agregação de dados: As redes, dependendo da programação utilizada, têm a capacidade de armazenamento dos dados coletados no caso de não haver carga útil suficiente ou necessidade de transmissão possibilitando uma economia de energia.
- Restrições de coleta: Indica restrições dos dados coletados como, por exemplo, o intervalo máximo de transmissão de seus valores para uma dada unidade de processamento.
- Quantidade de sensores: uma rede de sensores pode conter milhares de sensores, dependendo da aplicação, e, além disso, esse número é variável. Projetos robustos empregando redes *wireless* de sensores devem ser dimensionados para suportar a expansão da rede.
- Disponibilidade de energia: A vida útil de uma rede está diretamente ligada a essa variável. Nas WSNs, os nós geralmente são alimentados por baterias que possuem recursos de carga limitada. Dessa forma, em geral os dispositivos são programados para operarem até certo limite de carga e então saírem de atividade. Portanto, é necessário um bom dimensionamento tanto da carga disponível quanto do consumo dos nós para garantir uma duração viável à operação das redes com o maior período possível de substituição das baterias.

Para monitoramento e controle eficientes é necessário que todos os sensores e atuadores sejam endereçáveis sobre a rede. Diferentes protocolos e padrões industriais para automação predial têm sido propostos ao longo dos anos, como por exemplo, CE bus, EIB, BACnet, DyTEE e etc.

2.1.1 CAUSAS DE DESPÉRDIO DE ENERGIA EM WSNs

A seguir são apresentadas as principais causas, identificadas na literatura (Ye, Heidemann e Estrin, 2002), de ineficiência energética em redes *wireless* de sensores:

- Colisões: em ocasiões em que um nó recebe transmissão de dados de mais de uma fonte simultaneamente, ocorrem às chamadas colisões que resultam várias vezes em perdas de pacotes corrompidos por falhas na recepção. Assim, os pacotes corrompidos devem ser descartados e o receptor não envia de volta ao emissor a mensagem de confirmação de recepção, chamado pacote ACK (do inglês *acknowledge*). Dessa forma, faz-se necessária uma retransmissão que irá aumentar tanto a latência do sistema como o consumo de energia.

- Escuta excessiva: Eventualmente um nó da rede pode receber um pacote de dados que não é destinado a ele. O pacote recebido deve então ser descartado ocorrendo assim um desperdício energético na recepção, processamento e descarte do pacote.
- Excesso de cabeçalho: parte dos dados transmitidos na comunicação é referente ao cabeçalho (do inglês *control packet overhead*) que são dados de controle necessários para o estabelecimento de uma boa comunicação, identificando o tipo de mensagem, o emissor, o destinatário etc. Esses bits representam uma porcentagem significativa dos dados transmitidos consumindo parte da energia usada na transmissão e seu excesso representa desperdício de energia.
- Escuta ociosa: ocorre quando um nó está fazendo sensoramento em busca de receber dados enquanto o canal está inativo.
- Desencontro de informações: pode ocorrer quando uma mensagem é transmitida para um nó que está no seu período de dormência ou inatividade.

2.3 TOPOLOGIAS DE REDES

Topologia de rede se refere à forma estrutural da comunicação entre os nós, ou seja, especificam as direções de propagação dos fluxos de dados nas redes. A seguir são apresentadas algumas importantes topologias utilizadas em redes *wireless* de sensores e atuadores:

2.3.1 PONTO A PONTO (P2P)

No presente trabalho, será utilizada a topologia ponto a ponto (do inglês *Peer to Peer-P2P*). Essa topologia é geralmente composta por dispositivos que podem operar de 3 formas, são elas coordenador, roteador ou dispositivo final ou folha (*end-device*). Os módulos empregados podem ser de 2 tipos, os FFD dispositivos de funções completas (*Full function device*) que geralmente são dispositivos mais complexos necessitando de *hardware* mais complexo e podem exercer qualquer papel na rede, e os RFD dispositivos de funções reduzidas (*Reduced function Device*) que possuem *hardware* mínimo e geralmente funcionam como dispositivo final (ou folha) podendo apenas se comunicar apenas com nós FFD.

Nessa topologia todo dispositivo pode se comunicar com qualquer outro dispositivo no seu raio de alcance permitindo a formação de topologias de rede mais complexas como a topologia em malha (mesh).

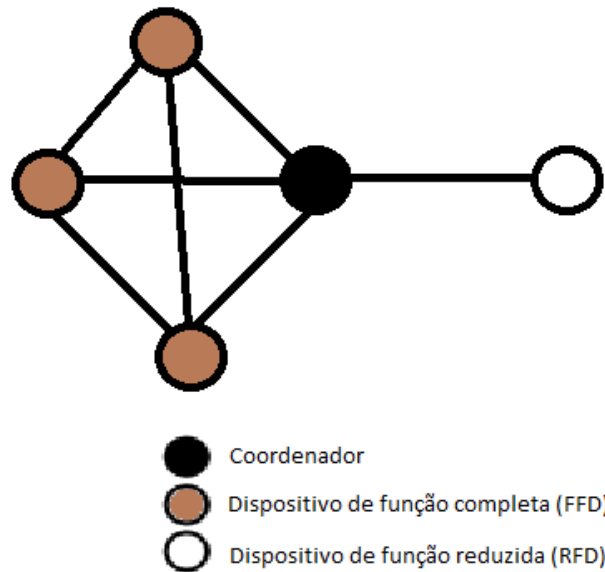


Figura 2.2- Topologia Ponto a Ponto.

2.3.2 MALHA (MESH)

Na topologia em malha existe uma ligação física direta entre cada um dos nós, isto é, todos se comunicam com todos como é possível observar na figura abaixo:

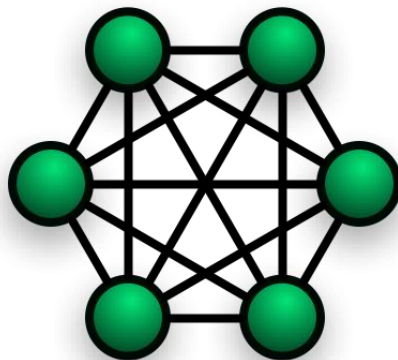


Figura 2.3- Topologia em malha.

2.3.3 TOPOLOGIA ESTRELA

Nesta topologia, um nó coordenador é conectado a quantos nós finais (do inglês *End-Device*) forem necessários. Assim, diversos nós realizam comunicação com o coordenador, o que configura um problema, pois pode haver congestionamento na rede. Geralmente, essa topologia é utilizada em locais de fácil comunicação, com poucos obstáculos, como uma sala por exemplo.

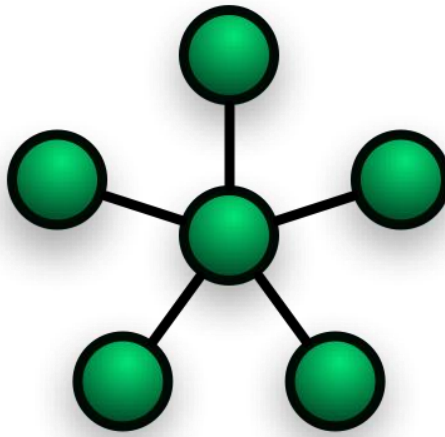


Figura 2.4- Topologia Estrela.

2.3.4 ÁRVORE OU ESTRELA HIERÁRQUICA (*CLUSTER TREE*)

Nessa topologia os participantes estão divididos em níveis de comunicação, possuindo os nós folha, geralmente sensores, e os nós roteadores que podem atuar como sensores e retransmissores. Também tem-se os nós coordenadores, que geralmente só realizam a coleta dos dados dos demais nós.

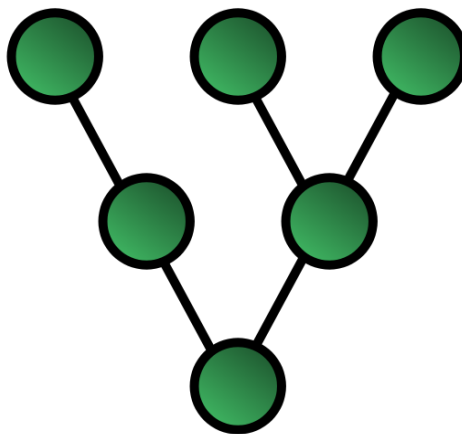


Figura 2.5- Topologia em Árvore Hierárquica.

2.3.5 TOPOLOGIA LINEAR

Nesta topologia cada nó está dentro do alcance de transmissão do outro, mas somente o primeiro nó (nó origem) gera novas mensagens em intervalos regulares. Durante os períodos ativos de um determinado protocolo, cada participante da rede só transmite sua mensagem ao seu vizinho adjacente em direção ao destino, até que a mensagem chegue ao destinatário, que atua apenas como um receptor.

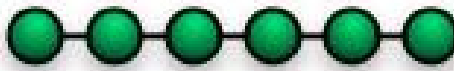


Figura 2.6- Topologia Linear.

2.4 PROTOCOLOS DE COMUNICAÇÃO

Em uma rede, o formato dos dados transmitidos e as regras a serem seguidas para que a comunicação se mantenha estável e funcional são especificados pelos protocolos de comunicação. Um protocolo, portanto, pode ser comparado a idiomas ou linguagens que servem para estabelecer comunicação entre computadores e dispositivos eletrônicos. Os protocolos determinam como os dados são preparados para transmissão e também definem o sequenciamento dos processos de comunicação. Nos trabalhos anteriores, os quais apresentaram o desenvolvimento do DyTEE, foram trabalhados protocolos como, CSMA/CA, TDMA, S-MAC e “IEEE 802.15.4 MAC”. Abaixo serão apresentados os principais protocolos de interesse nesse trabalho:

2.4.1 CSMA/CA

Em redes sem fio as distâncias entre os nós geralmente são curtas em relação ao tempo de transmissão de um pacote. Por isso, é interessante fazer o sensoriamento do meio antes de uma transmissão para prevenir colisões e conseqüentemente retransmissões que acarretam maior e desnecessário consumo de energia. Dessa forma, o protocolo de acesso múltiplo com verificação de portadora com anulação/prevenção de colisão CSMA-CA surge como uma solução para esse problema.

Esse protocolo além de se basear na escuta do meio antes de uma transmissão, possui também um mecanismo de deferência do tipo *binary exponential back-off*. Caso o meio esteja livre, um nó espera um intervalo de tempo chamado IFS (do inglês *interframe space*) e então inicia a transmissão quando o meio estiver livre. Já se o meio estiver ocupado (outros nós transmitindo), o nó espera até que o meio esteja livre. Quando o meio estiver livre novamente, então o nó espera um IFS e ativa um temporizador de contenção que se inicia com um tempo aleatório. Quando o temporizador expirar, então é feita a primeira tentativa de transmissão. Se a transmissão é realizada com sucesso, o nó espera um pacote de confirmação de transmissão ACK. Se o pacote ACK não for recebido, assume-se que houve falha na transmissão e é feita uma nova tentativa. A figura 2.7 apresenta o diagrama temporal do CSMA-CA.

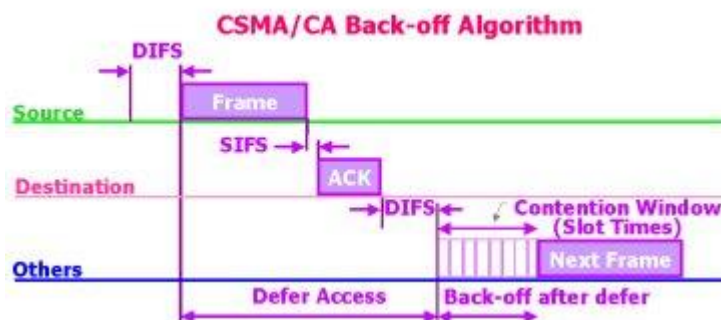


Figura 2.7- Diagrama temporal CSMA-CA.

2.4.2 IEEE 802.15.4

Este padrão desenvolvido pelo Instituto de Engenheiros Electricistas e Eletrônicos IEEE (do inglês *Institute of Electrical and Electronics Engineers*) define a camada de comunicação, camada de enlace de dados, no nível 2 do modelo OSI (*open system Interconnection*), e é responsável pelas camadas física e MAC do protocolo ZigBee que é, neste trabalho, testado e comparado ao DyTEE quanto à eficiência energética. Em alguns dispositivos, esse protocolo é definido usando o XBEE 802.15.4 OEM módulo digital. Nessa camada, as unidades de informação digital (bits) são gerenciados e organizados para se tornarem pulsos eletromagnéticos (ondas) em um nível mais baixo do modelo OSI correspondendo ao nível 1, camada física- PHY.

As frequências definidas no padrão são dispostas entre 27 diferentes canais divididos em 3 bandas principais. Das 3, a de interesse está na faixa de 2,4 a 2,48GHz fazendo uso de 16 canais. Para essa faixa de frequência, a taxa de transmissão de bits é de 250 Kbps. O protocolo IEEE 802.15.4 é eficiente no que se refere à imunidade, pois utiliza sequência direta de expansão do espectro DSSS (do inglês- *Direct Sequence Spread Spectrum*, alargamento de espectro de sequência direta) para modular a informação antes de enviá-la a camada física. Basicamente, cada bit de informação a ser transmitido é modulado dentro de 4 diferentes sinais. Tal processo faz com que a informação transmitida ocupe uma maior largura de banda, mas usando uma baixa densidade espectral para cada sinal como é apresentado na figura 2.8. Isso causa menor interferência nas bandas de frequência utilizadas melhorando a relação sinal ruído (SNR) no receptor por facilitar a detecção e decodificação da mensagem transmitida.

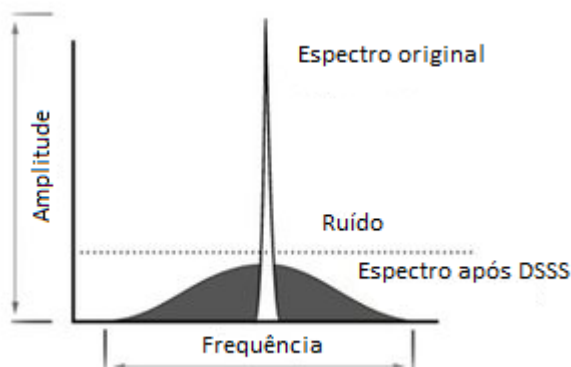


Figura 2.8- alargamento espectral dos sinais transmitidos.

No que se refere à interferência, o 802.15.4 usa duas técnicas de evitar que todos os nós comecem a emitir ao mesmo tempo. O primeiro e mais comum é o *Carrier Sense Multiple Access - collision avoidance* (CSMA/CA) no qual cada nó escuta o meio antes de transmitir. Se a energia encontrada for maior que certo nível, o nó espera durante um intervalo de tempo aleatório e tenta realizar a transmissão novamente. O segundo método corresponde ao intervalo de tempo garantido GTS (do inglês- *Guarantee Time Slots*), que usa um nó central coordenador para fornecer slots de tempo a cada nó, de modo que qualquer um deles saiba quando deve realizar transmissões. Existem 16 possíveis intervalos de tempo. Como um primeiro passo, um nó deve mandar ao coordenador uma mensagem de requisição de GTS. Como responsável o coordenador irá enviar a mensagem de sinalização contendo o *slot* de tempo alocado e o número de slots atribuído de maneira a organizar o processo.

Quanto a potência de transmissão e à sensibilidade de recepção dos dados, o padrão “IEEE 802.15.4” define a quantidade mínima de energia necessária para transmitir em -3dBm, e a sensibilidade mínima no receptor é -92dBm. Em relação à sensibilidade de recepção, o XBee mostra -92Bm e -100dBm possibilitando a esses dispositivos operar com potências de transmissão muito baixas, o que configura uma vantagem do ponto de vista energético.

Existem vários protocolos que utilizam o “IEEE 802.15.4” como sua camada MAC. O mais conhecido é o ZigBee, embora haja vários deles como por exemplo *wireless* HART, ISA - SP100, IETF IPv6 - LoWPAN entre outros.

2.4.3 TDMA

O protocolo de acesso múltiplo por divisão de tempo TDMA (do inglês *Time Division Multiple Access*) é uma abordagem clássica para acesso múltiplo em sistemas de comunicações *wireless* digitais, sendo utilizadas em diversas aplicações como, por exemplo, telefonia celular e sistemas de redes de computadores além de servir como base para protocolos de comunicação como o DyTEE e S-MAC.

Para entender melhor o funcionamento do protocolo TDMA, pode-se tomar como base sua aplicação em redes *wireless*. Nessas redes os usuários utilizam a mesma faixa de frequência para transmitir dados a uma baixa taxa de bits. Cada usuário possui um momento específico para realizar sua transmissão que é chamado *slot* de tempo e a comunicação pode ser no sentido de transmitir dados a outro usuário ou à rede (*upstream*) ou no sentido de receber dados (*downstream*). Cada receptor e transmissor de terminal opera com um ciclo de trabalho de 1/N se existem N usuários terminais com igual taxa de bits compartilhando o mesmo canal. A figura 2.9 ilustra a estrutura de operação TDMA:

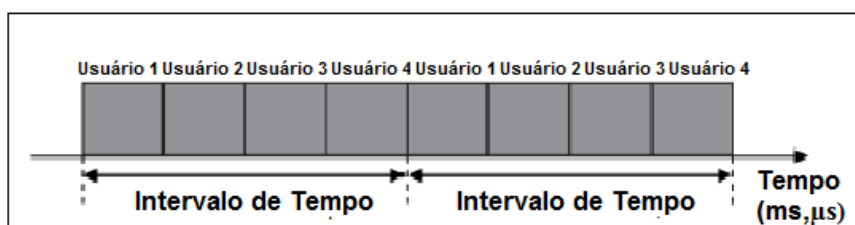


Figura 2.9- Organização temporal TDMA.

Em TDMA a interferência entre usuários é prevenida pelo cumprimento rigoroso da programação de divisão no tempo como apresentada na figura 2.9, isto é, cada usuário realiza suas transmissões apenas dentro de seu *slot* de tempo e, além disso, diferentes *slots* são separados por tempos de guarda. Outro mecanismo de prevenção a colisões é chamado de alinhamento de tempo de comunicação que consiste em dividir um *slot* de tempo em dois

períodos, um primeiro momento para o nó coordenador enviar mensagem ao dispositivo final e outro para uma comunicação no sentido contrário (dispositivo final envia mensagens ao nó coordenador). Assim o protocolo opera no sentido de prevenir colisões devido à propagação em diferentes tempos.

Cada *time slot* em um frame, conjunto de *slots*, geralmente contem bits de dados de usuário (cada um deve incluir bits de paridade e de controle de erros), bem como bits extra para sincronização, adaptação, controle, tempo de guarda etc. Quanto menor for a fração de bits de cabeçalho, mais eficiente é o TDMA pois a transmissão é a principal fonte de consumo de energia. Além da necessidade de minimizar-se os cabeçalho para uma menor economia energética, outro problema é observado no que se refere a determinação do número exato de nós na rede, pois este número é fixo uma vez que a rede é criada. A principal característica de interesse nesse protocolo que embasa tanto o S-MAC quando o DyTEE é a forma de prevenção à colisões por divisão no tempo, que permite evitar o desperdício devido a colisões.

2.4.4 S-MAC

O protocolo de comunicação S-MAC (Sensor MAC) é um dos primeiros protocolos desenvolvidos com ênfase em eficiência energética, tendo como principal foco redução no consumo, prevenção contra colisões e a solução de uma das restrições encontradas no TDMA no que se refere à flexibilidade de expansão da rede. Esse protocolo tenta reduzir o consumo energético de todas as fontes mencionadas encontradas na bibliografia (colisões, excesso de cabeçalho, escuta ociosa e escuta excessiva). Abaixo seu funcionamento é apresentado de acordo com Ye, Heidemann e Estrin (2002):

Esses problemas são solucionados por meio da utilização de ciclos de trabalhos dividido em períodos de atividade e estado de dormência (*sleep mode*) alternando entre si, prevenção contra colisões, escuta excessiva e transmissão de longas mensagens de maneira eficiente no que se refere a latência e energia (*message passing*).

No projeto desse protocolo, levou-se em consideração as peculiaridades das redes de sensores que divergem das redes tradicionais de computadores. Considerou-se que uma rede de sensores seja composta de diversos pequenos nós que tiram proveito da proximidade física para simplificar o processamento de sinal. O grande número de nós pode também proporcionar a vantagem de uma comunicação de curto alcance em múltiplos saltos (do inglês *multi-hop*) em vez de comunicação de grande alcance, o que representa uma maior conservação de energia. A maior parte da comunicação será entre nós aos pares, em vez de com uma única estação base como ocorre em telefonia celular.

Por haver muitos nós, eles serão implantados casualmente no estilo ad hoc (redes onde dois ou mais módulos se comunicam entre si) em vez de serem cuidadosamente posicionados. Os nós deverão, portanto, se autoconfigurarem.

Redes *wireless* geralmente se dedicam a aplicações específicas ou a diversas aplicações colaborativas, assim, em vez de equidade em nível de nós (como na internet), se foca em maximizar a *performance* de aplicação de todo o sistema. Processar dentro dos nós da rede (*In-network*) é crítico ao tempo de vida da rede de sensores. Uma vez que a rede de sensores é comprometida com uma ou várias aplicações, um código de aplicação específico pode ser distribuído ao longo da rede e ativar quando necessário ou distribuir sob demanda. Técnicas como agregação de dados podem reduzir o tráfego, enquanto um processamento de sinal colaborativo pode reduzir o tráfego e melhorar a qualidade do sensoriamento. Esse tipo de processamento implica em processar os dados como se fossem mensagens inteiras em vez de armazená-los e transmiti-los. Então intercalar intervalos em nível de pacotes ou fragmentos de múltiplas fontes apenas aumenta toda a latência. Leva-se em consideração também o fato

de que nessas redes, as aplicações permitem grandes períodos de inatividade podendo assim o sistema suportar considerável latência.

Em redes de sensores, algumas aplicações como vigilância ou monitoramento estarão ativas por grandes períodos de tempo, mas permanecerão inativas em grande parte do tempo até que algo seja detectado. Tais tarefas podem frequentemente suportar também alguma latência adicional. Uma abordagem de nível de aplicação para gerenciar a latência é implantar uma rede de sensores ligeiramente maior e ter nós periféricos aumentando a rede para informar quando algo é detectado. Todos esses fatores influenciaram o projeto do S-MAC.

2.4.4.1 CICLO DE TRABALHO PARA O PROTOCOLO S-MAC

O S-MAC reduz o tempo de escuta deixando que o nó vá para o estado de dormência. Por exemplo, se em cada segundo um nó dorme meio segundo e escuta na outra metade do segundo, seu ciclo de trabalho é reduzido em 50%. Então se atinge uma economia de energia perto de 50%.

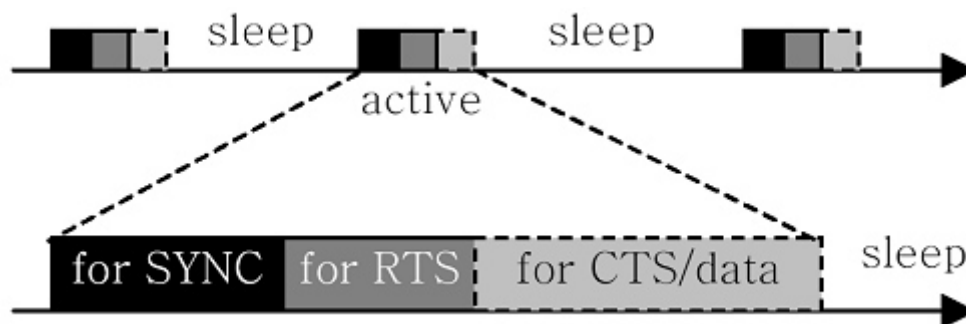


Figura 2.10- Diagrama de funcionamento do S-MAC. [2]

O esquema básico é apresentado na figura 2.10. Cada nó permanece no estado de dormência por um tempo, e então acordam e fazem sensoriamento o que é também chamado de escuta, para saber se algum outro nó deseja se comunicar com ele. Durante o período de dormência, o nó desliga seu rádio, e ajusta um temporizador para acordá-lo mais tarde. A duração dos períodos de dormência e atividade pode ser selecionada de acordo com os cenários de aplicações.

Os nós sensores não possuem um *clock* muito preciso, dessa forma esse esquema necessita de sincronização entre vizinhos para que não haja diferença entre seus clocks. Duas técnicas são usadas para tornar o sistema de sincronização robusto a erros. Primeiramente todas as marcas de tempo que são trocadas, são relativas em vez de absolutas. Em segundo lugar, o período de escuta é maior que o erro ou desvio do *clock*.

Comparada com o modelo TDMA, com slots de tempo muito curtos, o modelo para o S-MAC requer uma sincronização muito mais flexível entre nós vizinhos. Todos os nós ficam livres para escolher suas próprias programações para ativação e dormência. Entretanto, para uma redução de cabeçalho, é preferível que nós vizinhos se sincronizem juntos. Isto é, eles escutam no mesmo período e vão para o modo de dormência ao mesmo tempo. Deve ser notado que nem todos os nós vizinhos podem sincronizar em conjunto em uma rede *multi-hop*. Dois nós vizinhos A e B provavelmente terão programação diferente se eles sincronizarem com nós diferentes C e D como mostrado abaixo:



Figura 2.11- Nós vizinhos com programações diferentes. [1]

Os nós enviam sua programação por *broadcast* para todos os seus vizinhos. Dessa forma, todos os nós vizinhos podem se comunicar um com o outro mesmo se tiverem programação de tempos diferentes. Por exemplo, na figura 2.11 se o nó A quiser se comunicar com o nó B, ele apenas espera até B estiver em seu período de atividade. Se vários nós vizinhos quiserem se comunicar com um nó, eles precisam disputar pelo meio quando o nó estiver escutando. O mecanismo de disputa é o mesmo utilizado no protocolo 802.11, isto é, usa-se pacotes RTS (do inglês *request to send*) e CTS (do inglês *Clear to send*). O nó que envia o pacote RTS primeiro vence a disputa pelo meio, e o receptor irá responder com um pacote CTS. Após começarem a transmissão, os nós não seguem sua programação de dormência até o fim da transmissão

“Outra característica do S-MAC é que ele forma nós em uma topologia plana (*flat*) na qual nós vizinhos estão livres para se comunicarem não importando a programação de escuta que eles tiverem. A desvantagem desse esquema é que a latência aumenta devido aos períodos de dormência de cada nó. Além disso, o *delay* se acumula em cada salto. Então, a latência exige que a aplicação imponha um limite fundamental ao tempo de dormência.” [1]

2.4.4.2 ESCOLHA DE PROGRAMAÇÃO POR UM NÓ

Antes que cada nó comece seu ciclo de trabalho, ele precisa escolher uma programação e comunicá-la a seus vizinhos. Cada nó mantém uma tabela de programação que armazena os horários de todos os seus vizinhos conhecidos. O processo de escolha e manutenção de programação de acordo com Heidemann e Estrin segue a sequência de passos abaixo:

1. “O nó primeiro escuta durante certo período de tempo. Se ele não ouve uma programação de outro nó, ele escolhe aleatoriamente um tempo para ir para o estado de dormência e envia essa programação em uma mensagem SYNC por broadcast, indicando que irá dormir após t segundos. Chama-se esse nó de sincronizador, desde que ele escolha sua programação independentemente e outros nós irão sincronizar com ele.”
2. “Se um nó recebe uma programação de outro vizinho antes de escolher sua própria programação, ele segue essa programação ajustando sua programação para ser a mesma. Esse nó é chamado de seguidor. Ele então espera por um delay aleatório td e retransmite essa programação, indicando que irá dormir em $t-td$ segundos. O delay aleatório é para evitar colisões, para que vários seguidores disparados a partir do mesmo sincronizador não colidam sistematicamente quando retransmite a programação.”

3. “Se um nó recebe um horário diferente depois de seleccionar e transmitir seu próprio cronograma de atividades, ele adota dois horários. O nó transmite sua própria programação e depois vai para o estado de dormência.” [1]

2.4.4.3. ESTABILIDADE DA REDE

Como mencionado anteriormente, os nós geralmente não possuem um relógio com uma boa precisão, de forma que cada um possua um erro diferente e assim o tempo relativo entre eles varia. Isso configura um problema, pois para operar em ciclo sincronizado de trabalho onde os nós alternam estados de dormência e atividade conjuntamente, é necessário que o relógio de dois nós comunicantes esteja sincronizado. Dessa forma, é necessário que ocorra sincronizações periódicas entre nós vizinhos. Dessa forma, para que as transmissões ocorram com sucesso, é transmitido um pacote de sincronização SYNC sempre que se inicia uma transmissão. O SYNC é composto pelo endereço do emissor e seu respectivo período de dormência em relação ao momento em que a transmissão do pacote é concluída, ou seja, aproximadamente quando os receptores recebem o pacote. Os receptores então ajustam seus temporizadores imediatamente após receberem o SYNC. A figura 2.12 ilustra a sincronização no protocolo S-MAC:

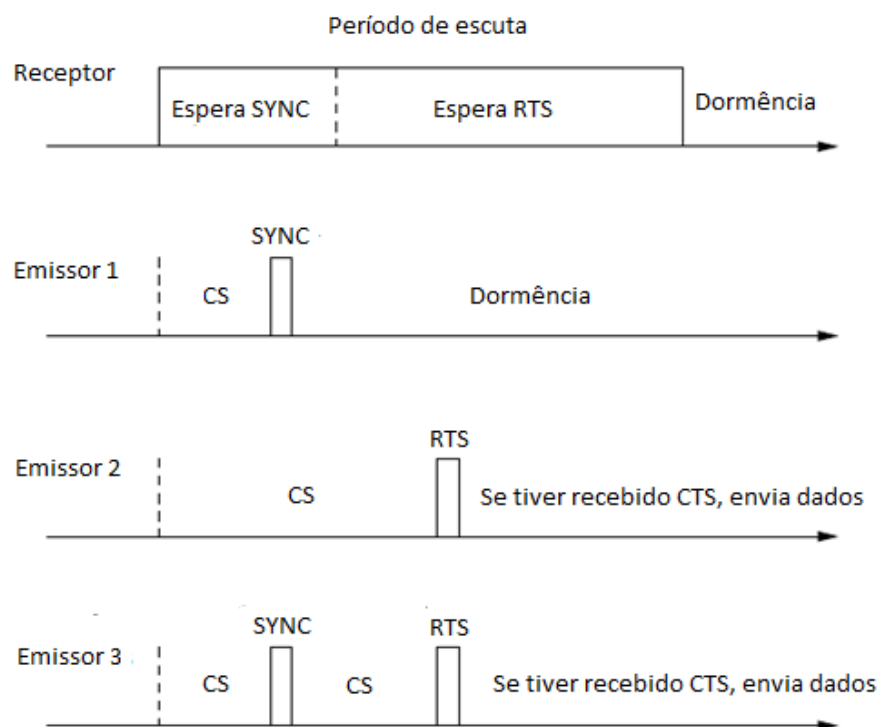


Figura 2.12- Relação de tempo entre um receptor e diferentes emissores. [1]

Cada parte é dividida em diversos *slots* de tempo para os emissores fazerem sensoriamento da portadora. Por exemplo, se um emissor deseja enviar um pacote SYNC, ele começa a sensoriar a portadora quando o receptor começa a dormir. Ele aleatoriamente seleciona o *slot* de tempo para terminar seu sensoriamento. Se ele não detecta nenhuma transmissão ao fim do *slot* de tempo, ele vence a disputa pelo meio e começa a enviar seu pacote SYNC nesse momento. O mesmo procedimento é seguido na transmissão de pacotes de dados.

A figura 2.12 mostra a relação de tempo em 3 possíveis situações em que um emissor transmite a um receptor. CS representa sensoriamento da portadora. Na figura logo acima, o

emissor 1 apenas envia um pacote SYNC. O emissor 2 apenas deseja enviar dados. O emissor 3 envia pacotes SYNC e RTS.

Quando a rede já está formada e os nós começam a realizar transmissões uns com os outros, surge um problema que é grande causa de ineficiência energética de outros protocolos que são as colisões nas transmissões. Uma colisão ocorre quando uma antena de um módulo receptor recebe pacotes de dados provenientes de mais de uma fonte ao mesmo tempo, de forma que parte das informações recebidas se perde e retransmissões são necessárias. Para evitar esse problema, nesse protocolo adotou-se um sistema baseado em contenção. Nesse esquema, todos os emissores fazem sensoriamento da portadora antes de iniciar sua transmissão. Se um nó falha em conseguir o meio, ele irá para o estado de dormência e acordará quando o receptor estiver livre e escutando novamente. Os pacotes de broadcast são enviados sem usar RTS/CTS. Pacotes transmitidos ponto a ponto seguem a sequência RTS/CTS/dados/ACK entre emissor e receptor. Dessa forma, apenas o nó que vence a disputa pelo meio realiza a transmissão, não havendo assim colisões durante esse processo. Apesar do sistema de contenção resolver o problema referente às colisões, escuta excessiva torna protocolos baseados em contenção menos eficientes no que se refere a energia do que protocolos TDMA, identifica-se assim esse novo problema que surge da solução anterior.

O S-MAC evita a escuta excessiva, permitindo que nós interferentes durmam após ouvirem um pacote RTS ou CTS. Desde que pacotes de dados são normalmente maiores que pacotes de controle, a abordagem previne a escuta excessiva de longos pacotes de dados e ACK por nós vizinhos.

A figura 2.13, apresenta uma rede de múltiplos saltos (do inglês *multi-hop*) formada pelos nós A, B, C, D, E e F. Nessa rede cada nó pode apenas ouvir transmissões de seu vizinho imediato.

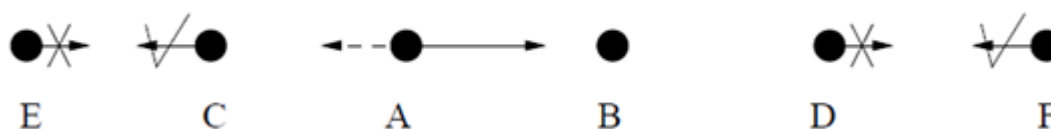


Figura 2.13- Os nós C e F dormem quando A transmite a B. [1]

Como as colisões ocorrem no receptor, se a transmissão de um nó interfere na comunicação entre dois nós vizinhos, esse nó deverá ir para o estado de dormência durante essa transmissão. Em contrapartida, se a comunicação entre dois nós não interfere na comunicação entre outros dois nós da rede, ambas as transmissões podem ocorrer simultaneamente sem que haja colisões.

Na figura 2.3, se, por exemplo, a transmissão do nó D interferir na recepção do nó B, então o nó D deve ir para o estado de dormência. Já se o nó C está há 2 saltos de B, e sua transmissão não interfere na recepção de B, então C pode transmitir mensagens a seus vizinhos como E por exemplo. De qualquer forma, C é desabilitado para receber qualquer resposta de E, por exemplo, CTS ou dados, porque a transmissão de E colide com a transmissão de A para o nó C. Em resumo, todos os vizinhos imediatos de ambos emissores e receptores devem dormir após escutarem pacotes RTS ou CTS até a transmissão corrente acabar.

O S-MAC busca uma transmissão de forma eficiente, fragmentando uma grande mensagem em diversos pacotes independentes e depois a transmitindo em broadcast, reduzindo assim o excesso de cabeçalho, pois dessa forma só faz uso de um pacote RTS e um CTS. Assim, o meio é reservado para transmissão de todos os fragmentos. Toda vez que um fragmento é transmitido, o emissor espera por um ACK do receptor. Se o ACK não é recebido, o nó emissor estende o tempo reservado para transmissões para mais um fragmento,

e então retransmite o fragmento corrente imediatamente. Além disso, se um nó vizinho escuta um pacote RTS ou CTS, ele irá dormir pelo tempo necessário para a transmissão dos fragmentos.

O propósito de usar ACK após cada fragmento de dados é prevenir o problema de ocultação de terminal. É possível que um nó vizinho acorde ou que um novo nó participe no meio da transmissão. Se o nó é apenas um vizinho do receptor, mas não é o emissor da mensagem, ele não ouvirá os fragmentos de dados sendo transmitidos pelo emissor. Se o receptor não envia ACK frequentemente, o novo nó pode erroneamente inferir de seu sensoriamento da portadora que o meio está limpo. Se ele começa a transmitir, a transmissão será corrompida no receptor. Assim, o S-MAC opera conforme descrito acima, garantindo melhor eficiência energética de transmissão, melhor latência e menor contenção com relação ao IEEE 802.11, atingindo os objetivos a que foi proposto.

3 PROTOCOLOS IEEE 802.15.4 E DyTEE

Trabalhos anteriores que fornecem as bases para este trabalho, foram desenvolvidos sobre o padrão ZigBee. Esse padrão foi escolhido por ter características como baixa potência e baixas taxas, transmissão de dados, dispositivos de tamanho reduzidos e redes com alcance reduzido (cerca de 100 metros). Tais características são fundamentais ao alcance dos objetivos deste projeto.

Esse padrão foi desenvolvido para proporcionar uma operação em redes *wireless* com baixa potência de operação, de forma que pudesse operar fazendo uso de baterias com uma longa vida útil. Esse protocolo foi desenvolvido tendo como principais finalidades automação predial e controle remoto.

A arquitetura segue o modelo de camadas OSI, com algumas alterações. As camadas física (PHY) e MAC foram modificadas seguindo os protocolo “IEEE 802.15.4”. Já as camadas de aplicação (APL), rede (NWK) e suporte a aplicação (APS) seguem o modelo definido pela ZigBee Alliance.

TABELA 3.1- Pilha do Protocolo

Usuário	Aplicação
ZigBee Alliance	Suporte e Aplicação
	Rede (NWK)/Segurança (SSP)
IEEE 802.15.4	MAC
	PHY

O ZigBee é um padrão normalizado desenvolvido pela *ZigBee Alliance* que é composta pelo IEEE em conjunto com mais de 300 empresas que visam atingir baixos custos, boa eficiência energética, segurança e confiabilidade para redes sem fio. Como diversas empresas fazem parte da *ZigBee Alliance*, existem alguns fabricantes de *hardwares* que produzem diferentes plataformas de desenvolvimento sem fio que buscam atender a esse padrão que possibilita a comunicação entre dispositivos de diferentes fabricantes.

No desenvolvimento desses dispositivos, são utilizadas várias técnicas de economia de energia. Dentre essas técnicas, otimizações no *hardware* como o uso de antenas mais eficientes, visam evitar desperdício de potência na transmissão de dados. O sistema operacional proporciona modos *sleep*, nos quais os módulos dormem quando não estão trocando dados com a rede, operam em uma espécie de dormência consumindo um mínimo de energia. Além dessas formas de economia, o processamento de sinais e a comunicação entre os dispositivos também são voltados à eficiência energética. Assim, o baixo consumo de potência atingido possibilitou a esses dispositivos operar utilizando baterias simples, pilhas AA, o que confere grande versatilidade e diversificação em suas funções e também um problema, pois a substituição das baterias faz-se necessária periodicamente.

Os dispositivos, como já mencionado, podem ser dos tipos: dispositivo de função completa FFD (do inglês- *Full Function device*) que podem operar de qualquer forma na rede (sensor, atuador, roteador e coordenador) podendo se comunicar com qualquer módulo da rede exigindo *hardware* mais potente; ou dispositivo de função reduzida RFD (do inglês *Reduced Function Device*) que só operam como atuadores ou *end device* se comunicando apenas com dispositivos FFD, o que os possibilitam usar os mínimos recursos de *hardware*.

Em uma rede ZigBee, o nó coordenador faz o reconhecimento de todos os nós da rede e suas respectivas funções como receber e enviar dados, além disso, servem como ponte de ligação entre diferentes redes. As redes podem dispor de milhares de nós. Um nó Roteador tem a função de executar a comunicação entre nós intermediários da rede, sem precisar do nó

coordenador possibilitando uma expansão no alcance da rede. Os demais nós são chamados de dispositivos terminais (*end device*) e são os sensores e atuadores.

Dentre as aplicações, os módulos permitem atuar em diversas tarefas, tais como: sensoriamento de umidade, temperatura, velocidade e direção dos ventos, pressão atmosférica etc. Controle de iluminação, aquecimento, ventilação, irrigação e podem atuar como alarmes, e serem utilizados em aplicações automotivas bem como desempenhar outras diversas funções sendo de grande utilidade em diversas aplicações envolvendo automação e controle.

3.1 CAMADA FÍSICA

Como pode ser observado na tabela 3.1, a camada física (PHY) segue o protocolo 802.15.4, sendo responsável pela transmissão das PDUs (*Protocol Data Units*), unidades de dados, por RF. Os módulos operam em faixas de frequência variável (868 MHz, 915 MHz e 2.4 GHz). No Brasil, é utilizada a faixa de frequência de 2,4 GHz, que não necessita de licença para operar, com taxa de transmissão de dados variando entre 20 Kbps e 250 Kbps. Sua latência máxima varia de 15 ms (periféricos) a 100 ms ou mais (automação residencial).

Na transmissão, a camada utiliza modulação alargamento espectral de sequência direta DSSS (do inglês *Direct Sequence Spread Spectrum*) que permite que o dado seja identificado como pertencente a um determinado nó, facilitando a detecção de erros. Ao espalhar os dados em todas as frequências da banda utilizada, o sinal resultante se assemelha cada vez mais a um ruído, tornando-se mais robusto a interferências.

Após ser feito a DSSS, o sinal é modulado em uma portadora para transmissão. Na faixa de frequência trabalhada (2400-2483,5 GHz) os módulos utilizam modulação O-QPSK (do inglês *Offset Quadrature Phase Shift Keying*). Abaixo são essas duas modulações são descritas em maiores detalhes:

- DSSS (*Direct Sequence Spread Spectrum*)

Para obter-se um espalhamento espectral, multiplica-se cada bit da mensagem a ser transmitida por um trem de bits de espalhamento chamada sequência de Barker (1 0 1 1 0 1 1 1 0 0 0). Assim, cada bit do sinal original passa a ser representado no sinal a ser transmitido por 11 bits. Esta modulação aumenta a banda de frequência ocupada pelo sinal. Esse processo é apresentado abaixo:

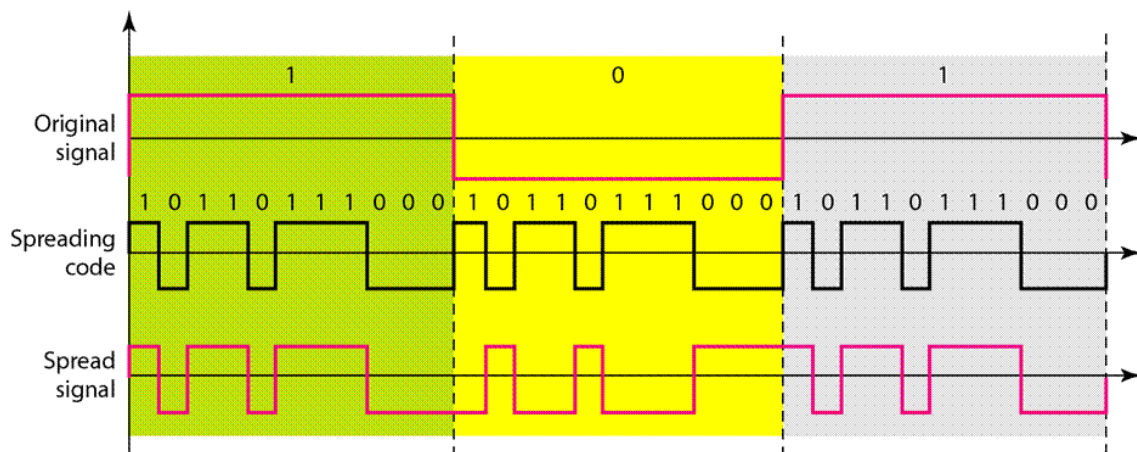


Figura 3.1- modulação DSSS. [3]

Após esse procedimento, o sinal passa a ter uma largura de banda maior que a original:

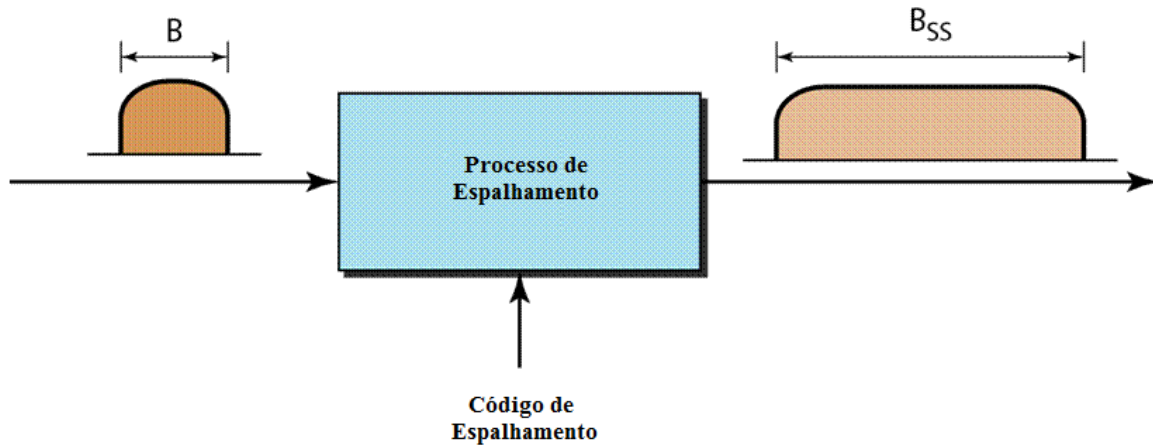


Figura 3.2- Alargamento espectral em DSSS. [3]

- O-QPSK

Também conhecido como QPSK deslocado, tem por objetivo minimizar os efeitos do alargamento espectral (causado na modulação DSSS). Essa modulação aumenta a eficiência do sistema pois permite que os sinais passem por amplificadores não lineares (mais eficientes que os lineares) ocupando a mesma largura de banda de QPSK. Diferentemente do que ocorre na modulação QPSK (bits dos feixes modulados em fase e em quadratura possuem transição em tempos iguais), na modulação O-QPSK os bits relativos ao feixe em quadratura sofrem um deslocamento no tempo em relação aos bits em fase. Esse deslocamento é de meio símbolo e conserva a forma do espectro de frequência sendo ideal em projetos que priorizam eficiência espectral e baixa potência.

Cada uma das faixas implica em uma taxa de transmissão, número de canais e espectros diferentes como mostrado à seguir:

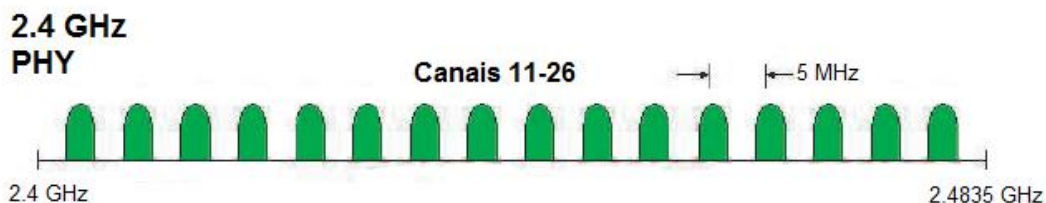


Figura 3.3- Bandas de Frequência.

A camada física também é responsável pelas características indicadas abaixo:

- Indicar qualidade de conexão

“Ao receber um pacote, o nó receptor envia de volta um pacote LQ (Link Quality) que determina qualidade do sinal de conexão. O valor desse pacote é calculado pelas camadas superiores de Rede e Aplicações de acordo com a relação sinal-ruído e o valor do pacote ED (Energy Detection).” [4]

- Detectar potência dos canais

“A detecção e manipulação desse dado são feitas na camada de rede para definir algoritmos de escolha de canais. O ED representa em 8 bits a relação em dB da potência recebida nos canais de acordo com os limites mínimos e máximos de sensibilidade. Dessa forma, os nós irão direcionar a informação por um bom caminho dentro da topologia.” [4]

- Reportar canais livres (*CCA - Clear Channel Assessment*)

“A camada física determina se os canais estão ocupados de acordo com o Carrier Sense (sensoriamento da portadora) dos sinais em DSSS e/ou caso o parâmetro ED esteja acima do limite do canal.” [4]

3.2 CAMADA MAC

Como em todas as redes médias compartilhadas, o controle de acesso ao meio MAC é uma técnica que permite a operação bem sucedida da rede. Uma das tarefas fundamentais de um protocolo MAC é evitar colisões de forma que dois nós interferentes não façam transmissões ao mesmo tempo. Nas plataformas de desenvolvimento ZigBee, essa camada segue o padrão IEEE 802.15.4. Existem no mercado diversas plataformas que atendem essas necessidades como Telos, IRIS, MICAZ e Zigbit, o presente trabalho foi desenvolvido sobre plataforma Zigbit escolhida em trabalhos anteriores por possuir diversas características que se encaixam perfeitamente aos objetivos do trabalho.

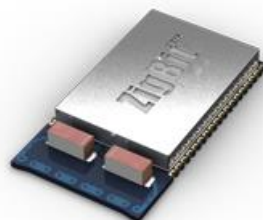


Figura 3.4- Módulo Zigbit.

O ZigBit é produzido pela Atmel para atender as necessidades do padrão 802.15.4, e proporciona baixo consumo de energia, baixo custo, robustez na troca de dados nos dispositivos, gravação remota, expansibilidade da rede e uma fácil utilização e manipulação

da rede. Assim, ao se utilizar essa plataforma, ficam garantidas a flexibilidade e segurança na troca de informações.

Os módulos Zigbit levam embarcado um ATmega 1281 que possuem baixíssimo consumo de energia nos modos de dormência (algo em torno de 6uA de corrente), boa sensibilidade de recepção melhorando as transmissões, pinos auxiliares para diversas aplicações como GPI/O, SPI, USART/UART I²C e antena com ganho baixo porém adequado à implementações em automação predial que geralmente não demandam alto alcance.

O Zigbit opera utilizando 2 sistemas operacionais, o primeiro é o BitCloud e o segundo é o pacote “IEEE 802.15.4 MAC” que é disponibilizado pela Atmel e é o principal fator na escolha dessa plataforma. O pacote “IEEE 802.15.4 MAC” é desenvolvido em linguagem C e é aberto em todas as camadas tendo como principal protocolo o 802.15.4. Desta forma, essa plataforma foi utilizada para a primeira implementação do DyTEE [2], pois os módulos são mais compactos e de menor custo, bem como a linguagem desta é mais convencional.

3.2.1 DyTEE

O presente trabalho é desenvolvido em cima do protocolo de acesso ao meio DyTEE (do inglês *Dynamic Timed Energy Efficient*), que foi desenvolvido anteriormente na Universidade de Brasília no trabalho de mestrado o Aluno Vinicius Galvão Guimarães. Esse protocolo foi desenvolvido com base nas melhores características de outros protocolos como o TDMA e S-MAC. Seu desenvolvimento foi feito sobre o pacote chamado IEEE 802.15.4 MAC que fornece um pacote completo de desenvolvimento em linguagem C aberta em todas as camadas, diferentemente do pacote principal, o BitCloud, que permite acesso apenas na camada do usuário.

No desenvolvimento do DyTEE, buscou-se eliminar o excesso de cabeçalho, a escuta excessiva e principalmente as colisões que ocorrem durante as transmissões. Assim, nesse protocolo as falhas nas transmissões ocorrem apenas por falhas na recepção e nunca por colisões.

O DyTEE é um protocolo com sincronização global, empregando um único canal de operação, e além disso, garante a transmissão pelo pior atraso. Ou seja, garante a confiabilidade do sistema ao custo de uma grande latência.

3.2.1.1 PARÂMETROS CONFIGURÁVEIS DO DyTEE

DyTEE foi proposto para operar em redes de automação predial onde se busca uma maior eficiência energética em comparação aos protocolos comuns no mercado como o S-MAC e o IEEE 802.11. Como existem os mais variáveis cenários de aplicações para esse tipo de rede, o protocolo possui vários parâmetros configuráveis, para uma melhor adaptação à aplicação em questão. Abaixo são apresentados os principais parâmetros configuráveis de acordo com [2]:

- Número máximo de nós em cada grupo

Uma rede bem dimensionada permite que o protocolo opere de forma ótima. Escolher um valor acima manterá o funcionamento da rede, porém não irá alcançar 100% da eficiência energética possível.

- Ciclo de trabalho máximo

“Determina o tempo máximo que os nós poderão ficar ativos (incluindo sincronização, pacotes de controle e dados úteis) em relação ao tempo em que ficarão inativos;” [2]

- Tamanho máximo de uma mensagem enviada no *beacon* (sinal de sincronização)
No DyTEE faz-se possível o envio de mensagens durante os sinais de sincronização.

- Quantidade de vagas para registro
Número de vagas que ficarão disponíveis para registro de novos nós.

- Máximo número de períodos para recuar (*Backoff*)
Quantos períodos um nó pode atrasar sua requisição, ainda que de forma aleatória, para requisitar novamente uma tentativa de registro com determinado grupo;

- Máximo número de tentativas de registro

- Máximo de requisições de transmissões por períodos (não é útil)
“Em aplicações que possuam maiores taxas de geração de mensagem, pode-se reduzir o consumo da rede por meio deste parâmetro. Isso pois, sabendo-se que há um grande número de requisições por tempo, torna-se possível selecionar este parâmetro, em paralelo com o ciclo de trabalho máximo, de forma que cada nó realiza em média mais de uma requisição por período. Configurando corretamente estes dois, pode-se reduzir também o total de cabeçalho utilizado no protocolo e conseqüentemente o consumo também.” [2]

- Carga útil máxima por mensagem
Quantidade máxima de bytes por mensagem. Se a camada de aplicação tiver disponibilidade de acumular os dados para posteriormente enviá-los em uma única transmissão, esse parâmetro pode reduzir o consumo de energia da rede.

- Máximo de mensagens armazenadas
Esse limite depende da memória disponível nos módulos.

- Máximo número de tentativas de transmissão de mensagens
“No DyTEE, as falhas de transmissão só devem ocorrer por falhas na recepção e não por colisão. Sendo assim, este parâmetro permite que a camada MAC responda à camada de aplicação com um resultado positivo ou negativo a respeito da requisição de transmissão de uma mensagem.” [2]

Além dos parâmetros acima que levam em consideração o funcionamento da rede, cada nó possui parâmetros individuais chamados parâmetros do módulo (PM). Isso pois ao solicitar à camada MAC uma conexão à um grupo, a função de solicitação exige estes parâmetros. Esses parâmetros são:

- Canais disponíveis

- Cargo a ser exercido
É possível pré-definir a função desejada para um nó na programação do módulo, que pode ser de coordenador ou de folha (End Device).

- Identidade original:
“Indica a camada MAC qual a identidade que este nó deseja utilizar para se registrar. Não se recomenda utilizar identidades iguais em nós operando no raio de duas vezes o de alcance. Por padrão, se utiliza dois bytes de identidade por grupo.” [2]

3.2.1.2 INÍCIO DE UMA REDE NO DyTEE

Em WSNs, é bastante comum que a rede se inicie de forma automática. No DyTEE assim como em outros protocolos, a rede tem seu início de forma automática. O ciclo de trabalho desse protocolo é apresentado na figura abaixo:

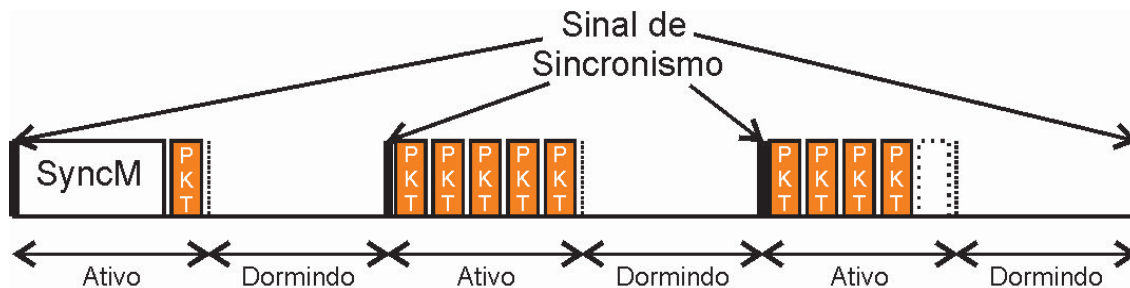


Figura 3.5- Diagrama temporal do protocolo DyTEE. [2]

O período SyncM, representa o momento onde se dá o início da rede. Nesse período, os nós esperam por um tempo aleatório por sinais de sincronização (do inglês *beacon*) de um possível nó coordenador que busca criar a rede. Esse tempo aleatório será menor para nós com maior disponibilidade de carga em sua fonte de alimentação. Nesse período se um nó não escuta esses sinais, ele então inicia a sua própria rede. Um nó folha que recebe o *beacon* do coordenador, se juntará a rede criada, e através de uma troca de informação recebe então um número de identificação para comunicação com o grupo. Para a comunicação dos nós em múltiplos saltos, formando um caminho de dados, os módulos já conectados em um coordenador podem continuar procurando por novos grupos periodicamente, a fim de manter as informações de conexão disponíveis para formar novas redes e expandir suas rotas.

Além disso, a coordenação do grupo não é concentrada apenas no nó que inicia a rede. Ela é alternada de acordo com a carga disponível para os nós. Quando um nó coordenador utiliza certa quantidade de sua bateria, ele encerra o grupo e assim, uma nova rede é iniciada pelos demais nós da mesma forma descrita anteriormente.

3.2.1.3 ESTRUTURA GERAL DO PROTOCOLO

Durante o SyncM, logo após o sinal de *beacon*, os nós tentam enviar ao coordenador uma requisição de transmissão, informando o destino para suas mensagens. As requisições são feitas em momentos programados para a comunicação de cada nó conectado, para que não haja colisão nesse processo. Dessa forma, durante o funcionamento da rede, um nó só realizará transmissões se no período SyncM o coordenador permitir a comunicação e reservar o *slot* de tempo para que ele se comunique. O coordenador então realiza toda a programação da rede, definindo o período de transmissão para cada conjunto de nós, bem como os períodos de dormência, e as os destinos da comunicação. Em seguida, ao final do SyncM, essa programação é transmitida a todos os nós da rede por broadcast. A figura 3.6 apresenta a estrutura do momento de sincronização de uma rede.



Figura 3.6- Diagrama do período de sincronização (SyncM) do DyTEE. [2]

No DyTEE, as comunicações entre os nós são restritas aos intervalos de tempo reservados para que elas ocorram, não podendo nunca exceder esse limite como ocorre no S-MAC onde um nó pode solicitar a extensão do período de transmissão. Além disso, existe um tempo de guarda entre os períodos que alternam transmissões entre diferentes pares de nós, de forma que não ocorram colisões causadas por transmissões simultâneas interferentes. Isso também permite que outros grupos possam operar simultaneamente em uma mesma área. Para que a comunicação se estabeleça corretamente, é necessário que o cabeçalho identifique a mensagem, sua finalidade, o protocolo utilizado, o emissor e o destinatário. No DyTEE o cabeçalho é fixo e de grande importância para que o receptor identifique a mensagem. o cabeçalho fixo é dado pelo primeiro byte sendo sempre o total de bytes que deverão ser lidos em seguida, enquanto o segundo corresponde ao campo de controle do pacote (FCF do inglês *frame control field*) e o terceiro e último byte fixo corresponde somente ao termo definindo o protocolo DyTEE (no caso, em hexadecimal, 0xDE). Os valores de FCF identificam o tipo da mensagem, e são interpretados da seguinte maneira:

- **FCF_FRAMETYPE_SIMPLE_BEACON:** Mensagem *beacon* para início de transmissão.
- **FCF_FRAMETYPE_SYNCM_BEACON:** Define o pacote como *BS* para *Sync*.
- **FCF_FRAMETYPE_DATA:** Define o pacote como *PKT* (mensagem).
- **FCF_FRAMETYPE_ACK:** “Define o pacote como *ACK*.”
- **FCF_FRAMETYPE_REGISTER_REQUEST:** requisição de registro à rede;
- **FCF_FRAMETYPE_DISASSOCIATE:** pedido para se desvincular da rede.
- **FCF_FRAMETYPE_REQUEST_SLOT:** requisição de transmissão, onde o nó pertencente à rede informa ao coordenador que deseja transmitir algo para outro nó;
- **FCF_FRAMETYPE_MAC_CMD:** broadcast enviada pelo coordenador para agendar os eventos do grupo.

Ao final da mensagem os bytes FCS (do inglês *frame check sequence*) são inseridos para conferir se a transmissão ocorreu de maneira correta. Ao receber a mensagem, parâmetros do canal são inseridos nesse byte, indicando a qualidade da conexão (LQI do inglês *link quality indication*) e o nível de energia (ED).

Após o envio dos sinais de *beacon* para sincronização, inicia-se o período de registro que se dá por tentativas. Para se registrar na rede um nó envia ao coordenador uma solicitação de registro RTR e recebe juntamente às demais informações no broadcast do Syncm, uma confirmação de registro ACK.

Quando há sucesso na solicitação de registro, o líder do grupo analisa a possibilidade de registro e então insere a resposta no pacote enviado durante o broadcast do SyncM. Esta pode conter um valor negativo (seja por grupo estar cheio, por motivos internos ou por falha na recepção) ou positivo. A resposta positiva contém o ID para este nó neste novo grupo.

Após o intervalo de registro, cada nó faz junto ao coordenador uma requisição de vaga de tempo para a sua transmissão (Rslot do inglês *request slot*). Por meio de um Rslot também

é possível requisitar uma desassociação do grupo, seja porque o nó irá se desligar definitivamente ou porque o nó deseja participar de outro grupo.

Em seguida o coordenador programa a agenda da rede, definindo os períodos de dormência para cada nó, bem como o momento correto para que os nós despertem e iniciem sua transmissão (slots de tempo). A agenda definida é então transmitida por broadcast à todos os nós integrantes da rede.

Quando o nó líder envia o sinal de *beacon*, os nós sincronizam o seu *clock* e então realizam o seu ciclo de trabalho pré-definido, despertando no momento configurado para realizar sua comunicação dentro de seu *slot* de tempo e indo dormir no momento também pré-definido. A figura 3.7 mostra o ciclo seguido por um nó.

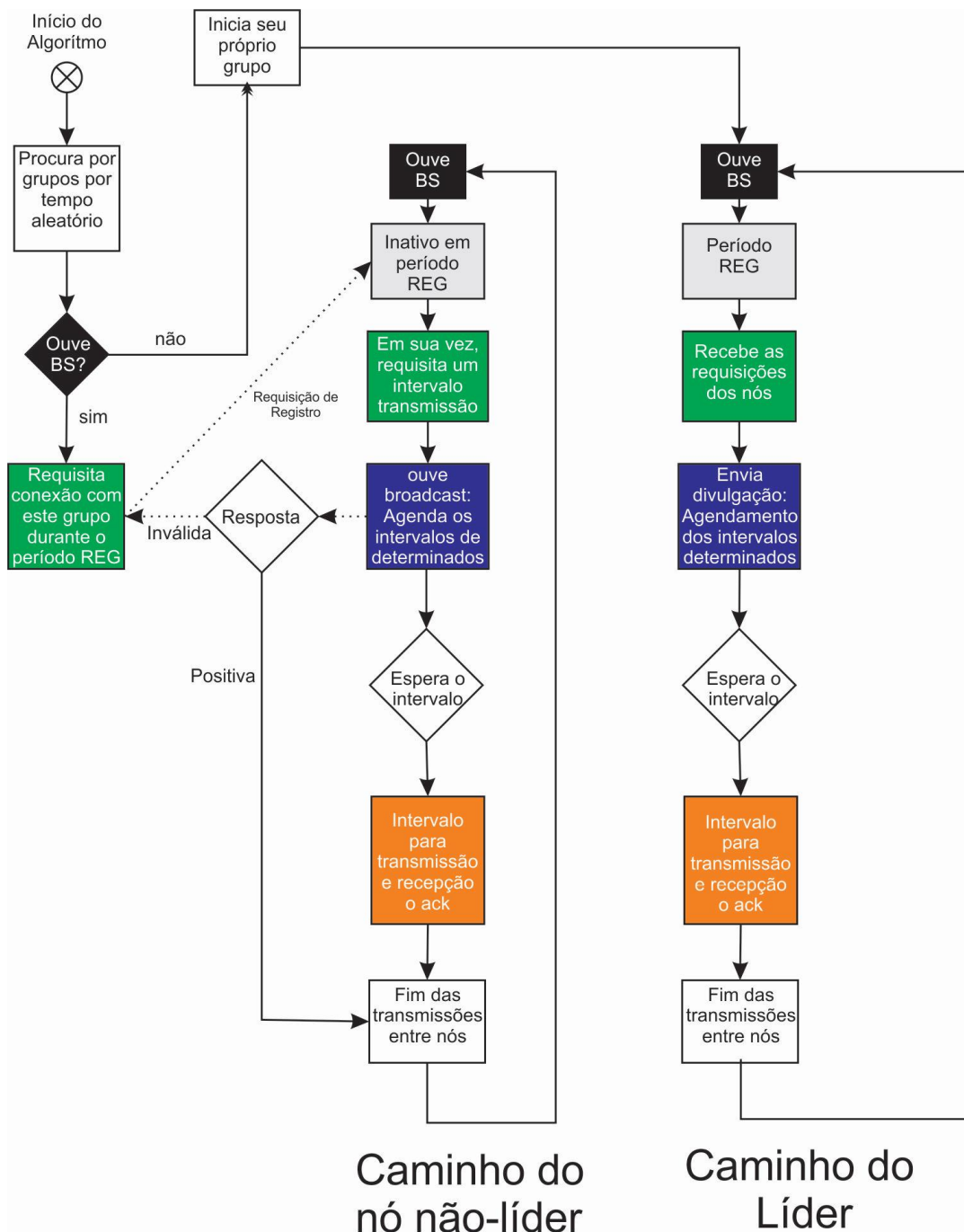


Figura 3.7- Fluxograma de operação para um nó que está executando o protocolo de DyTEE. [2]

É possível observar que o DyTEE mantém os nós da rede no estado de dormência pelo maior período de tempo, evita colisões através de seu ciclo de trabalho bem organizado e minimiza o cabeçalho uma vez que há apenas um período de sincronização. Todas essas características conferem ao DyTEE uma melhor eficiência energética, uma vez que configura um aprimoramento de características observadas em protocolos já existentes.

3.2.1.4 TESTES ANTERIORES

Em trabalhos anteriores, o DyTEE foi testado e comparado com o protocolo S-MAC apenas em simulações e ao IEEE 802.14.2. Para essa comparação [2], foi usado o software Matlab para simular o comportamento de ambos os protocolos durante seus respectivos *duty cycle*. As Simulações permitiram comparar os dois protocolos quanto:

- **VAZÃO**
O DyTEE apresenta melhor desempenho quando a rede está mais congestionada, já para taxas mais baixas o S-MAC se mostrou mais eficiente.
- **CONSUMO**
A eficiência energética do DyTEE se apresentou 5x superior ao do S-MAC. O DyTEE possui melhor economia de energia baseado tanto no tempo total necessário para transportar o número de mensagens na rede, quanto na quantidade de nós ligado por tempo.
- **LATÊNCIA**
Os resultados mostraram que quanto maior a latência, maior a vantagem do S-MAC em relação ao DyTEE.
- **EXCESSO DE CABEÇALHO**
- **COLISÕES**
Observou-se que o DyTEE não apresentou colisões, ao contrário do S-MAC.

Os testes em módulos reais, foram feitos primeiramente comparando o DyTEE com os resultados das simulações. Em seguida foram feitas criadas redes para comparação do DyTEE com o protocolo IEEE 802.14.4. As redes empregadas nesse processo foram redes genéricas onde os nós foram dispostos na topologia linear se comunicando entre si sem que houvesse atuação. Os parâmetros foram ajustados de forma que fosse possível comparar os dois protocolos. Utilizou-se um tempo ativo máximo de 500ms, na ordem 6 de intervalos de *beacons*, o que equivale aproximadamente a 1 segundo de período total. A rede foi composta de 8 módulos sendo um deles coordenador e o tamanho da identidade foi ajustada para 2 bytes (endereços curtos). A Seguir são apresentados os resultados dos testes de comparação às simulações:

- **VAZÃO**

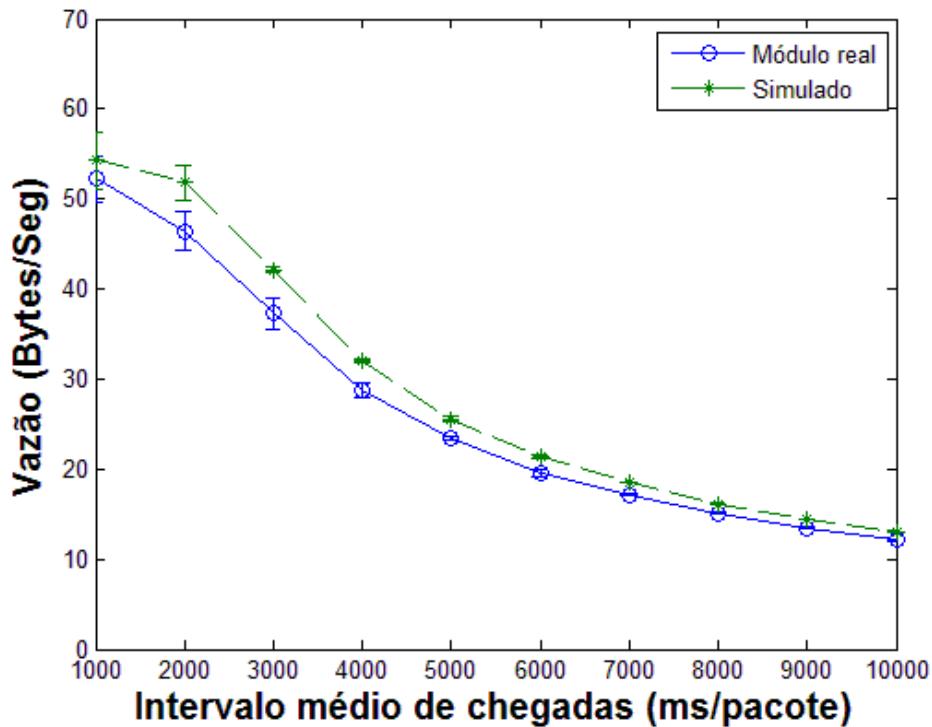


Figura 3.8-Vazão na rede simulada e real. [2]

Na figura 3.8, a curva azul representa os resultados esperados quanto a vazão no protocolo DyTEE. Observou-se que houve uma pequena diferença entre os valores obtidos e os valores esperados. A divergência entre o gráfico ocorre por falhas de assincronismo que não são modeladas nas simulações.

- **CONSUMO TOTAL DE ENERGIA**

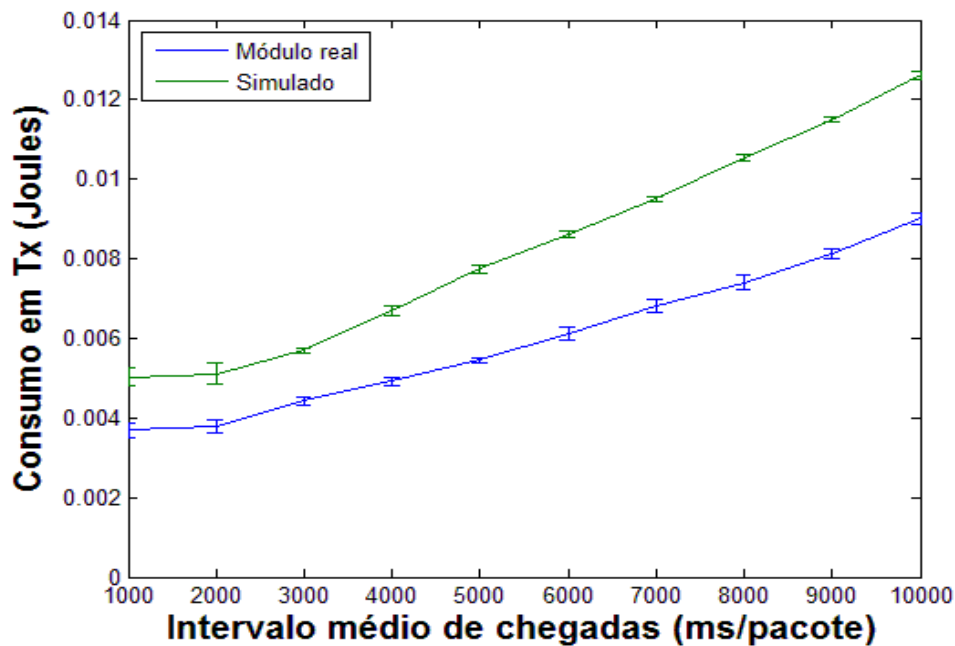


Figura3.9- Consumo total de energia da rede durante as transmissões. [2]

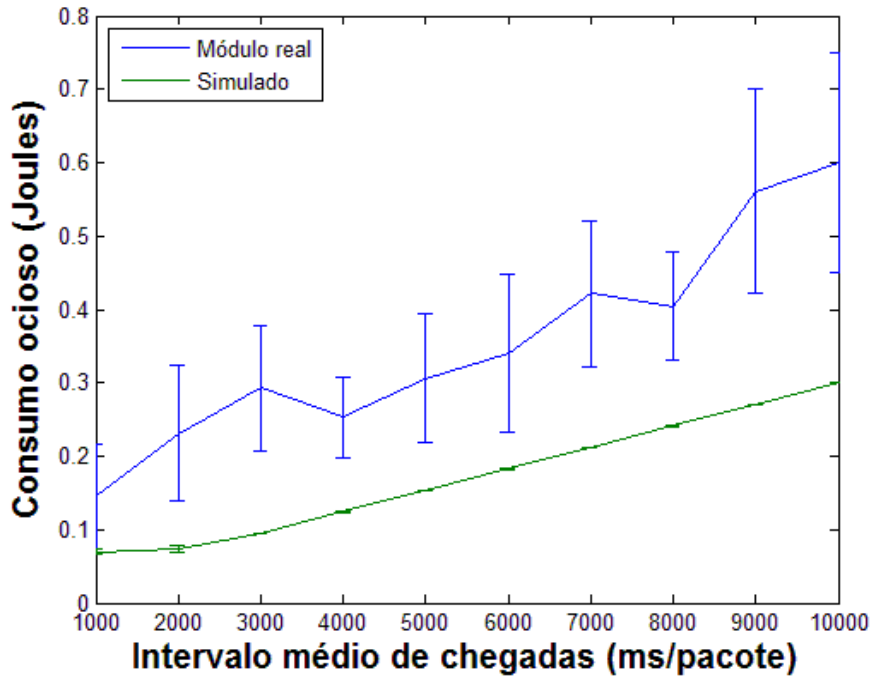


Figura 3.10- Consumo total de energia da rede durante os períodos ociosos. [2]

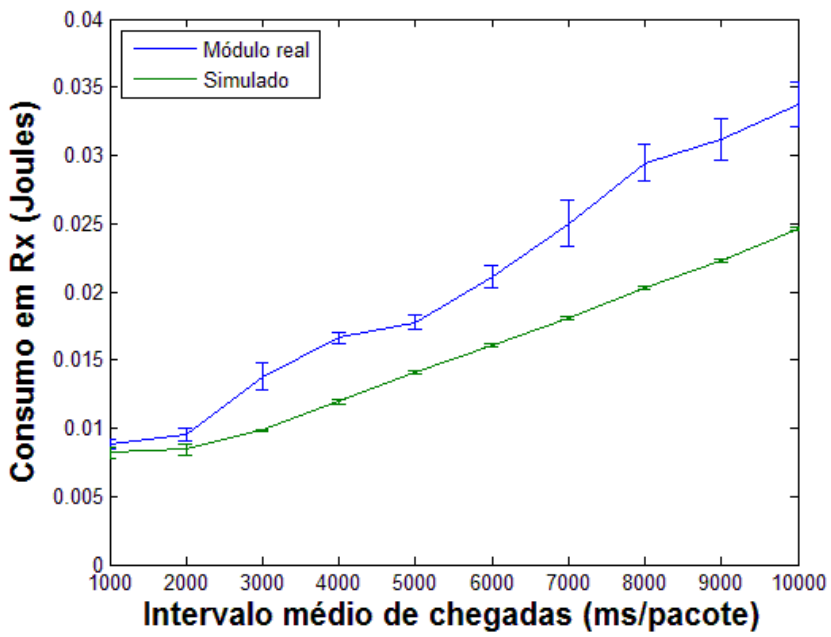


Figura3.11- Consumo total de energia da rede durante as recepções. [2]

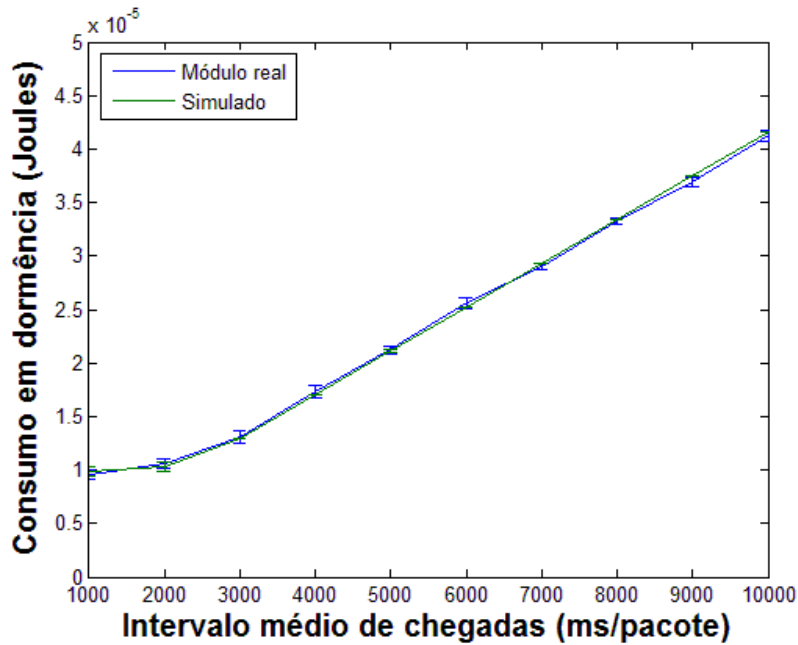


Figura3.12- Consumo total de energia da rede para os períodos de dormência. [2]

“Nos resultados de consumos apresentados (Figuras 3.9 a 3.12), a maior variação dos pontos ocorre no gráfico da figura 3.10, pois devido as transições dos estados que ocorrem durante a execução dos procedimento, a simulação possui uma transição instantânea, bem como não considera falhas e alguns atrasos incomuns que ocorrem na plataforma real utilizada. Dessa forma, o consumo ocioso dos módulos reais pode ser um pouco maior nos experimentos reais do que nos experimentos simulados, mas possuem valores próximos e com as mesma tendências.” [2]

- **Potência média consumida**

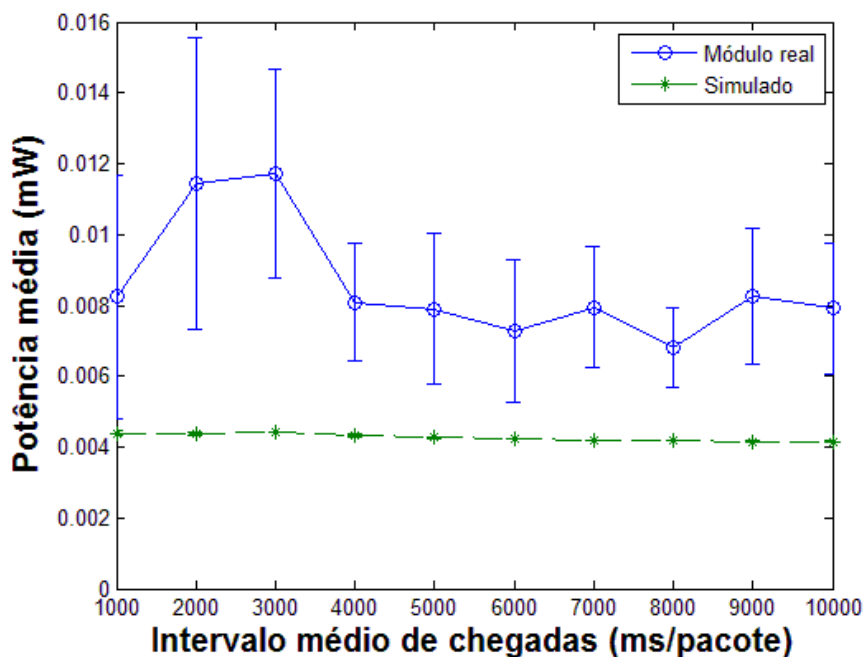


Figura 3.13- Potência média consumida. [2]

- Resultados comparando ao IEEE 802.15.4

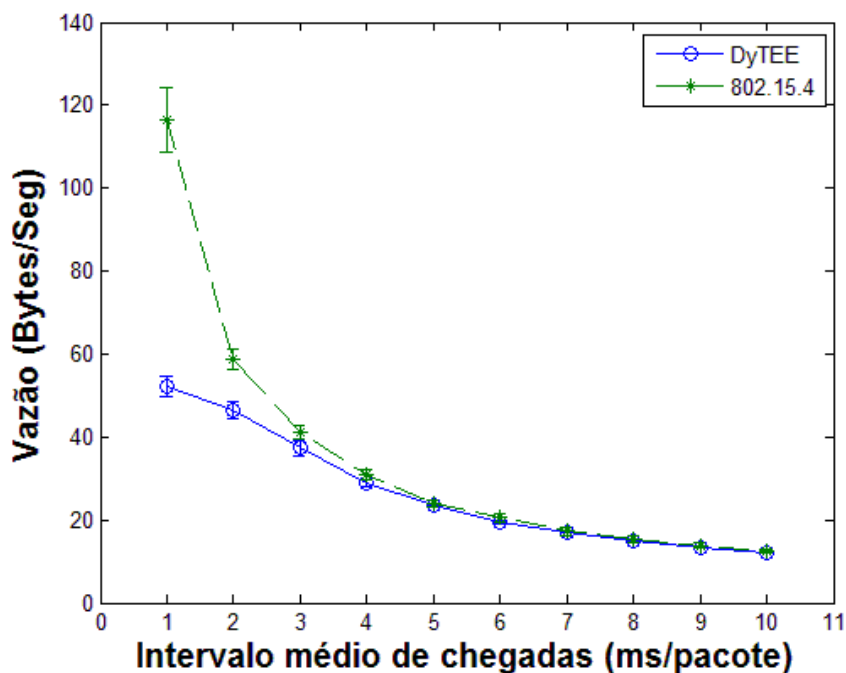


Figura 3.14- Vazão obtida em ambos protocolos em módulos reais. [2]

De acordo com as curvas acima, espera-se que para o protocolo 802.15.4 que o consumo por aumento de taxa de chegadas seja bem maior que no DyTEE. No DyTEE esse consumo ocorre dinamicamente de acordo com o número de mensagens existentes na rede.

Para o gráfico de vazão, ambos os protocolos apresentaram semelhante variação. Isso ocorre, pois o DyTEE não possui os parâmetros ideais para operar em taxas de chegadas inferiores a 2 segundos. Os parâmetros utilizados proporcionam a necessidade de no mínimo dois períodos completos (o que dura 2 segundos) para o envio de todas as mensagens recebidas pelos nós. A partir destes pontos, ambos protocolos atendem ao sistema com vazões semelhantes, as quais são somente limitadas pela taxa de chegada.

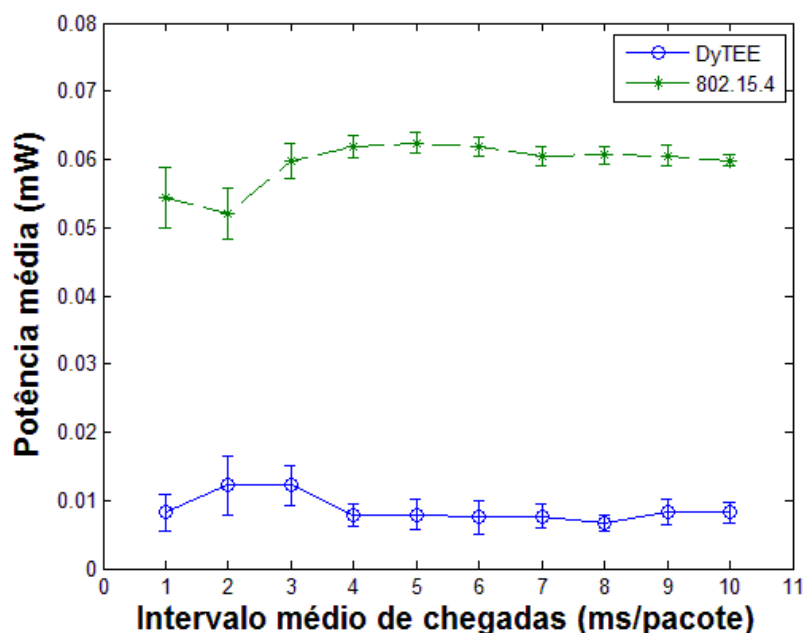


Figura 3.15- Potência consumida em módulos reais. [2]

Novamente, os pontos do gráfico apresentado na figura 3.15 não possuem um valor muito significativo, mas sim a sua magnitude em relação ao ponto do outro protocolo. Sendo assim, observa-se que o DyTEE possui um melhor desempenho do que o 802.15.4 durante todos os pontos de taxa de chegadas, mesmo nos casos onde o protocolo não foi projetado para operar de forma ótima (taxas de chegadas igual a 1 segundo).

4 ESPECIFICAÇÕES DO PROJETO

4.1 A REDE IMPLEMENTADA

A rede criada para os testes tanto do DyTEE quanto do IEEE 802.15.4, consiste de apenas dois nós, um nó coordenador e um nó sensor. Nessa rede, o nó sensor é responsável por medir a temperatura no interior do protótipo de sistema térmico e enviar os dados medidos ao coordenador que então verifica a necessidade de atuação ou não. O nó sensor foi colocado na sala 2 (*ROOM 2*). Um esquemático do sistema implementado é apresentado na figura 4.1.

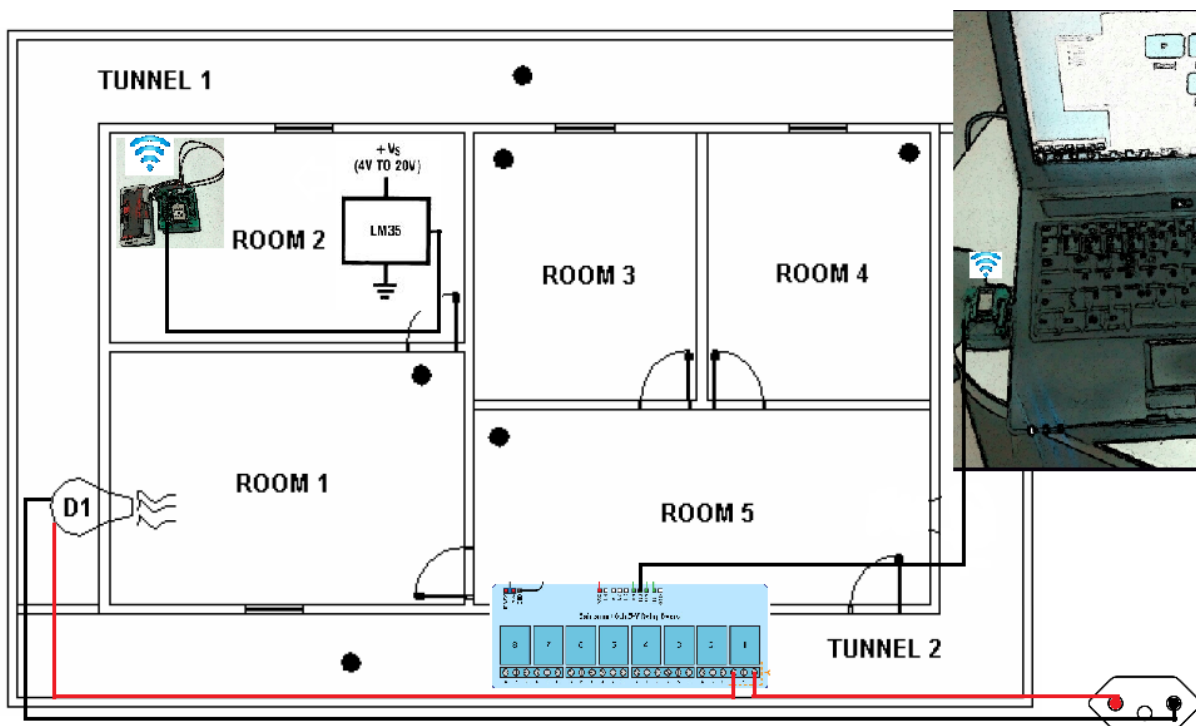


Figura 4.1- Sistema implementado.

A medição de temperatura é feita através do sensor LM35 acoplado à entrada ADCK da placa Zigbit Channel 1.0. A atuação é feita através de um relé de estado sólido que ao receber sinais do nó coordenador, liga ou desliga o secador localizado na sala 1. Os materiais utilizados são descritos em maior detalhes à seguir.

4.1.2 METODOLOGIA DE CRIAÇÃO DA REDE

Devido à finalidade desse projeto de comparar a eficiência energética dos protocolos de comunicação, consideramos a mesma topologia quanto às atividades realizadas em cada nó.

Para rede do DyTEE temos o nó definido como folha, que opera na rede realizando a leitura de temperatura fornecida pelo sensor Lm35, e envia o valor de temperatura ao nó coordenador, o qual se encarrega de fornecer um sinal de controle para ligar ou desligar a placa *Sainsmart 8* ligada ao secador, funcionando como atuador.

Já na rede do ZigBee um *end device* funcionara como o nó folha e o coordenador atuará sobre o secador com o sinal de controle.

O mecanismo de controle de temperatura ajustado no nó coordenador é bem simples apenas considerando uma temperatura acima da qual o secador deverá estar desligado, escolhida para o experimento em 35°C, e deverá enviar o sinal para que o secador ligue, quando a temperatura se encontrar abaixo dos 33°C, ou seja, liga/desliga com histerese de 2°C.

4.1.3 DECLARAÇÃO DAS PLATAFORMAS UTILIZADAS EM CADA PROTOCOLO

A implementação dos códigos que descrevem a seqüência de tarefas a serem realizadas pelos nós coordenador e *end device*, quanto a rede de automação utilizando o protocolo de rede *ZigBee*, foi realizada a partir das bibliotecas definidas no pacote *bitcloud*, em linguagem de alto nível C. O pacote de bibliotecas definidas no *bitcloud* seguem o padrão definido pelo protocolo *ZigBee*. A camada MAC nas bibliotecas do *bitcloud* está programada para funcionar com a configuração do IEEE 802.15.4 e não está acessível ao usuário.

Já para a configuração do DyTEE foi utilizado o conjunto de bibliotecas descritas no IEEE 802.15.4 MAC. Esse pacote permite uma programação em camadas de mais baixo nível em linguagem de programação C, fornecendo acesso à camada MAC onde foi implementado o DyTEE.

4.1.4 ANÁLISE DE CENÁRIOS ENCONTRADOS DURANTE A CRIAÇÃO DAS REDES NOS DOIS PROTOCOLOS

Para a criação da rede da automação utilizando o protocolo DyTTE, foi necessária a implementação da leitura analógica, a partir do conhecimento dos registradores responsáveis pelo processo, considerando a lógica de funcionamento obtida no documento [21], pois essa não se encontrava definida no pacote do IEEE 802.15.4 MAC.

Primeiramente, todo o processo da leitura analógica foi aplicado no código da função principal, *main*, na programação do nó folha. A leitura era realizada periodicamente de acordo com um tempo definido na programação.

Na rede do DyTEE os erros mais freqüentes observados durante a automação, foram de a mensagem ser enviada do nó folha e não ser recebida no período correto pelo coordenador e ainda, da saída da rede do nó folha devido a perda de sincronização após 4 novas tentativas de nova sincronização sem sucesso desse se associar novamente à rede.

Observado esse comportamento e ainda tendo em vista que no processo de multiplexação TDMA o sincronismo constitui um fator crítico ao correto funcionamento do sistema, foi realizada uma tentativa de oferecer maior capacidade de funcionamento a rede no protocolo DyTEE. Nesse sentido, implementou-se a função de leitura analógica na camada MAC, de modo que a leitura somente fosse realizada após o término de todos os outros processos da camada MAC, para que fossem evitados possíveis conflitos durante a realização das atividades durante o funcionamento da rede. Portanto, foi dada prioridade ao sincronismo da rede em relação ao processo de leitura analógica.

Contudo, não foi observada uma maior capacidade do nó folha de permanecer na rede, ainda que houvesse a prioridade da realização das atividades da camada MAC em relação à leitura analógica, pois a temporização ideal para o processo deveria levar em conta a temporização de todos os processos do DyTEE e quais seriam as atividades principais ao

funcionamento da rede. Entretanto, não constitui a finalidade deste trabalho a modificação do protocolo, mas apenas implementá-lo numa rede e compará-lo a uma outra rede.

A rede que operava com o padrão do ZigBee possuía uma maior estabilidade quanto ao funcionamento em relação à rede que operava com o DyTEE, sem que os nós se desassociem e permaneçam em comunicação enviando os dados de temperatura. A rede do ZigBee operou por um período máximo de 4 horas e a rede operando com o DyTEE apenas 9 minutos e 30 segundos. Contudo, é necessário que consideremos um mesmo período de tempo de operação para as duas redes ZigBee e DyTEE, para que haja uma possível comparação entre os protocolos.

Portanto, o consumo de energia total das pilhas será estimado, considerando que o consumo em um período de 60 segundos se repita até que a pilha forneça uma tensão abaixo da tensão mínima de operação 1,3 V em cada pilha.

4.2 MATERIAIS UTILIZADOS

4.2.1 PROTÓTIPO DE SISTEMA TÉRMICO

O protótipo de sistema térmico foi desenvolvido com o intuito de representar um ambiente de trabalho típico. Por motivo de facilidade, nos experimentos para controle de temperatura, usa-se o aquecimento em vez de refrigeração. Utilizam-se como fontes de calor secadores de cabelo de 1kW de potência. Nos experimentos realizados neste trabalho, o controle de temperatura foi feito apenas nas salas 1 e 2. Espera-se que os resultados obtidos sejam compatíveis com controle de temperatura em um ambiente real através de sistemas de ar-condicionado. O protótipo é mostrado na figura 4.2:



Figura 4.2 – Protótipo de controle de temperatura. [5]

4.2.2 PLACAS

Na implementação deste projeto, utilizou-se a placa Zigbit Channel 1.0, também desenvolvida na UnB. Essa placa opera como dispositivo FFD podendo ser utilizada tanto para operar como nó coordenador quanto como dispositivo final. Foram usados 1 módulo coordenador/Gravador e 1 módulo Dispositivo-Final.



Figura 4.3 – placa ZigBit Channel 1.0 que exerce a função de Coordenador/Gravadora.

A placa mostrada na figura 4.3 possui um circuito de tradução serial/USB, que é usado na realização da conexão direta entre o computador e a placa para a inserção de códigos (programas), comandos e envio de dados e resultados, além de alimentar a placa. Essa placa também conta com um botão reset, um LED de indicação de energia e dois LEDs de que indicam a transmissão serial.



Figura 4.4- Placa ZigBit Channel 1.0 com funcionalidade dispositivo final.

Na figura 4.4, temos a placa utilizada nos dispositivos finais alimentados por pilhas, que possui um circuito responsável pelo fornecimento de alimentação ao dispositivo, além de permitir acesso aos pinos do ZigBit.

O sensor possui faixa de medição de -55°C a $+150^{\circ}\text{C}$ com precisão de $\pm 0,5^{\circ}\text{C}$. É alimentado com tensão de 6 volts, retornando na saída um valor de tensão com a resolução de $10\text{mV}/^{\circ}\text{C}$. Seus dados técnicos são apresentados abaixo:

- Diretamente calibrada em graus Celsius
- Fator de escala Linear de $+10\text{-mV}/^{\circ}\text{C}$
- Assegura uma acurácia de $0,5^{\circ}\text{C}$ (a 25°C)
- Faixa de operação de -55°C a 150°C
- Indicado para operações remotas
- Baixo Custo
- Corrente de dreno menor que $60\mu\text{A}$
- Baixo aquecimento, $0,08^{\circ}\text{C}$ para baixo fluxo de ar
- Não linearidade típica de apenas $\pm 1/4^{\circ}\text{C}$
- Baixa impedância de saída, $0,1\Omega$ para 1-mA de carga

4.2.4 SAINSMART 8 CHANNEL DC 5V RELAY MODULE

Para a atuação no sistema, utilizou-se a placa sainsmart 8 channel DC relay module que geralmente é utilizada em conjunto com Arduino ou Raspberry Pi em sistemas de automação. Essa placa possui uma característica adicional de isolamento óptico, assim, através do acionamento de um LED o relé é acionado. Né é necessária qualquer conexão direta entre a fonte acionadora do LED e o circuito dos relés, o que confere uma segurança ao se controlar separadamente um equipamento que possui alimentação própria bem como carcaça metálica. A placa normalmente atua com uma tensão de entrada de 5 V mas também pode operar com tensões de 3,3 V.

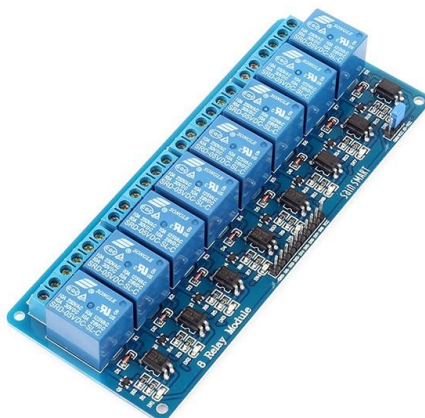


Figura 4.7- Sainsmart 8 channel DC 5V Relay module.

4.2.5 OSCILOSCÓPIO DIGITAL

Nas medições realizadas foi utilizado o osciloscópio digital MDSO. Esse osciloscópio faz comunicação com o computador através de porta USB e é operado pelo software *Multi Vir Analyzer* que permite armazenar tanto os pontos que geram os gráficos (em Excel) quanto as figuras geradas.



Figura 4.8- Osciloscópio digital MDSO.

Características técnicas:

4.2.5.1 OSCILOSCÓPIO/ ANALISADOR DE ESPECTRO

- Largura de banda de 20 MHz
- Amostras de 48 MHz
- Canal 2
- Tensões de entrada entre -50 V e +50 V (com ajuste X10 para a ponta de prova)
- Memória de 1M*2
- Conversor AD de 8 bits

4.2.5.2 DATA RECORDER (REGISTRADOR DE DADOS)

- Amostras entre 0.25 e 24 Mhz
- Canal 2
- Tensões de entrada entre -50 V e +50 V (com ajuste X10 para a ponta de prova)
- Conversor AD de 8 bits

A exibição dos dados armazenados pode ser contínua possibilitando zoom e movimentação da imagem. Além disso, os dados gravados podem ser convertidos tanto em

arquivo de imagem quanto em arquivos de Excel (.csv). O recurso de conversão dos dados em planilhas do Excel foi utilizado para importar os dados para o *software* Matlab e então fazer o tratamento matemático necessário à obtenção dos gráficos.

4.2.5.3 ESQUEMÁTICO DE LIGAÇÃO DO OSCILOSCÓPIO

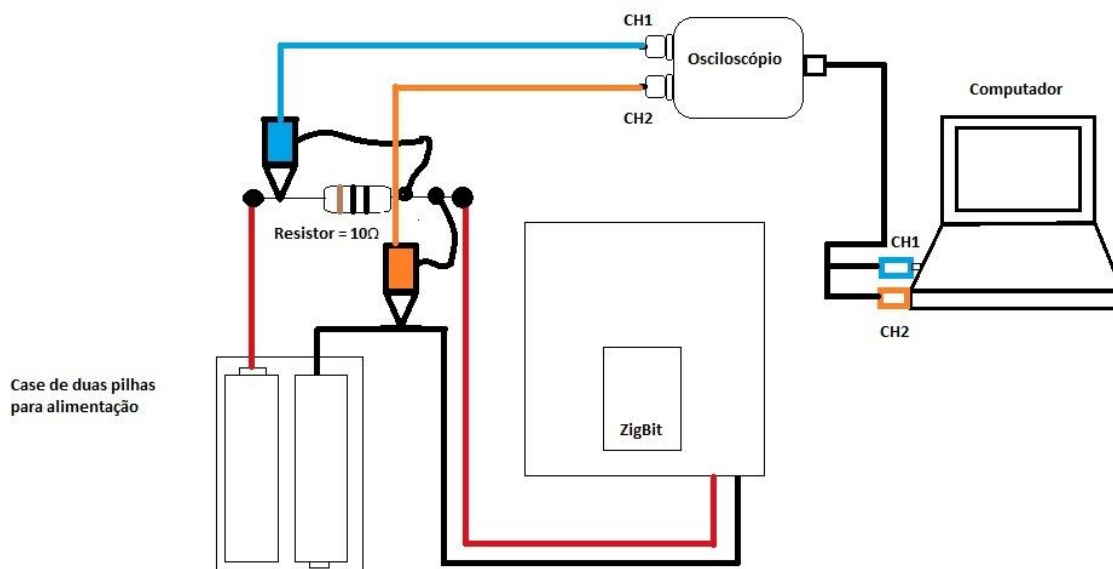


Figura 4.9- Esquemático de ligação do Osciloscópio.

Na figura 4.9 temos o esquemático de ligação para obter as medições de corrente e tensão nas pilhas. Onde a corrente foi obtida indiretamente utilizando a lei de Ohm, considerando a razão entre tensão dos terminais do resistor de 10Ω e o resistência desse. Já a tensão foi medida nos terminais das pilhas invertidamente, a fim de que se obtivesse o mesmo referencial no osciloscópio, contudo inverteu-se o sinal de tensão quando da inserção dos dados no Matlab. Para efeito de comparação, a potência consumida pelo resistor foi desconsiderada, pois sua influência é a mesma em ambos os cenários de teste.

4.2.6 MULTÍMETRO DIGITAL TRUE RMS 87-5- FLUKE

Algumas medidas de tensão sobre o resistor de 10 Ω foram feitas fazendo uso do multímetro *digital true RMS 87-5- FLUKE* cujas características técnicas de interesse são apresentadas a seguir:

Tensão CC

- Tensão máxima: 1000 V
- Precisão: $\pm (0,05\% + 1)$
- Resolução máxima: 10 V

Corrente DC

- Amperes máximos: 10 A (20 A durante um máximo de 30 segundos)
- Precisão em amperes: $\pm (0,2\% + 2)$
- Resolução máxima: 0,01 A
-

Resistência Interna

- Resistência máxima: 50 M Ω
- Precisão: $\pm (0,2\% + 1)$
- Resolução máxima: 0,01 Ω

Duty cycle

- Duty cycle máximo: 99,90%
- Precisão: $\pm (0,2\%$ por kHz $+0,1\%)$
- Resolução máxima: 0,10%



Figura 4.10- Multímetro digital True RMS 87-5- Fluke.

5 RESULTADOS

5.1 CORRENTE MEDIDA UTILIZANDO O DyTEE

A curva para a corrente demandada pelo nó folha foi obtida inserindo um resistor em série com o circuito de alimentação desse. E, portanto, por meio da tensão sobre o resistor, obtém-se a corrente, que é dada pela lei de Ohm $V=RI$. O padrão observado no primeiro minuto de medição é apresentado na figura 5.1 e se repete ao longo da vida útil da rede enquanto os nós se comunicam.

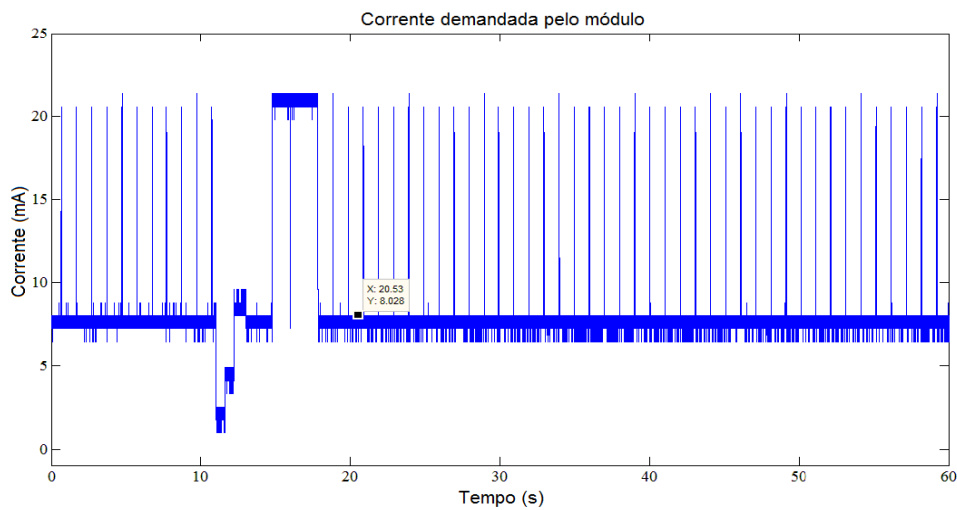


Figura 5.1- Corrente demandada pelo módulo.

A curva na Figura 5.1 foi tomada com o *duty cycle* ajustado para 1 s (intervalo de tempo entre as transmissões). O comportamento apresentado após a inserção do nó na rede se repetiu ao longo de toda a sua vida útil, enquanto esse se encontrada inserido na rede. Uma ampliação no gráfico da figura acima é apresentada na figura 5.2 e permite observar todo o ciclo de tarefas de acordo com a programação do nó implementada.

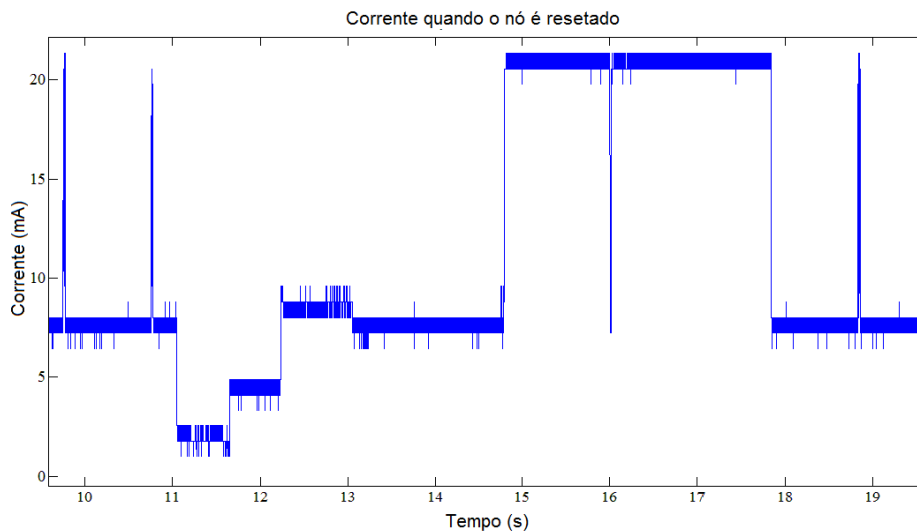


Figura 5.2- Comportamento da corrente ao se apertar o botão reset.

O botão reset foi pressionado aos 11 segundos de medição. A partir desse momento é possível observar, na figura 5.2, o nó executando a sequência de tarefas ajustadas em sua programação. Durante o período de 11 a 13 segundos, ocorre a inicialização dos periféricos e demais recursos relativos às camadas PAL, TAL e BMM.

Logo após os 13 segundos, o nó folha procura uma rede para que possa se associar. Quando o nó encontra a rede, aos 15 segundos, ocorre um salto na corrente demandada pelo nó indicando que esse encontrou uma rede e decidiu se associar. Essa associação ocorre durante o período de SyncM fazendo a requisição para participar da rede, contudo aos quatorze segundos irá dormir, aguardando a confirmação da sua requisição de pertencer à rede, mas não conseguindo se associar, novamente faz outra requisição, e recebendo a confirmação irá dormir antes dos 18 segundos, para acordar no período determinado pelo nó coordenador enviado no *broadcast*. O consumo de corrente durante o período de dormência varia entre 7,2 e 8 mA. Os dois picos longos apresentados entre 15 e 18 segundos são de 21,36 mA.

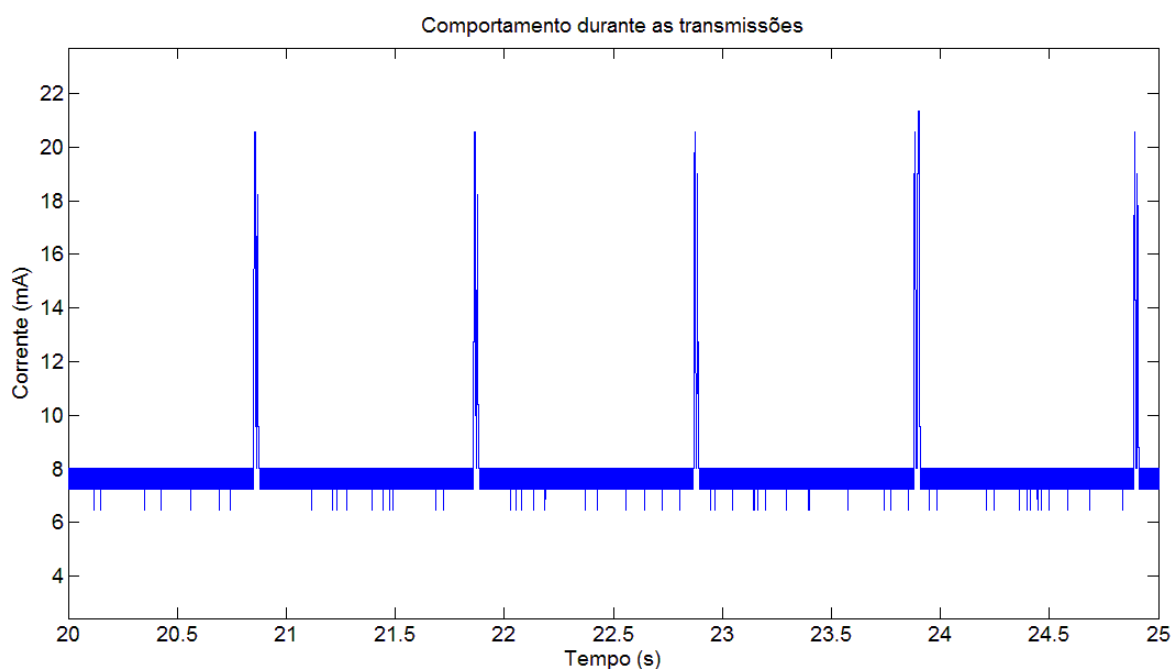


Figura 5.3- Picos de Transmissões.

A figura 5.4 apresenta vários o *duty cycles* do DyTEE após o período de SyncM. Os períodos de menor consumo com a corrente variando entre 7,2 de 8 mA, correspondem aos momentos em que o nó está dormindo e os picos representam a recepção do *beacon* e as transmissões do nó determinadas pelo coordenador. Ampliando a imagem da Figura 5.4 é possível observar mais claramente o que ocorre no período da recepção do *beacon* e da transmissão.

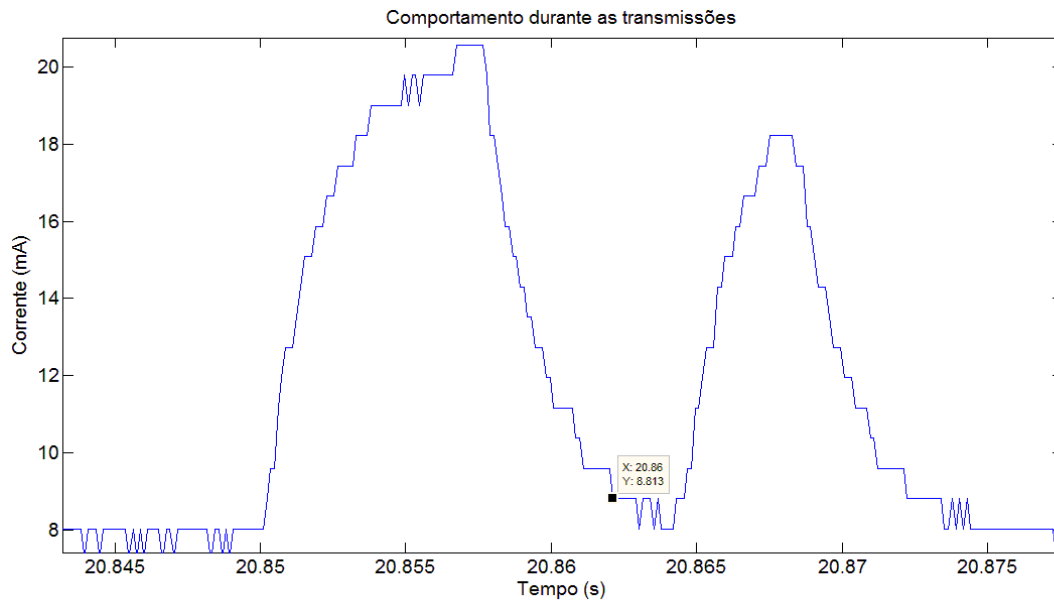


Figura 5.4- Período durante a recepção do *beacon* e da Transmissão.

Observa-se na Figura 5.5 a existência de 2 picos neste período. O primeiro pico tem duração de 0,01 s e possui uma amplitude de pico de 21,36 mA, e corresponde ao momento do *duty cycle* do DyTEE em que o nó acorda para receber a sincronização de seu relógio com o relógio do coordenador. A queda até o ponto destacado na figura corresponde ao período após a recepção do *beacon*. Já o segundo pico indica a transmissão da mensagem com a duração de 0,01 s e amplitude de 20,85 mA. O período apresentado durante a operação normal foi de 0,83 s.

Integrando a curva de corrente e dividindo pelo tempo, obtemos como resultado a média de corrente consumida pelos módulos operando com o protocolo DyTEE. O valor obtido a partir dessa média foi de 7,8 mA.

5.2 TENSÃO E POTÊNCIA

As medições abaixo foram realizadas com períodos de transmissão maiores que os dos procedimentos anteriores. Abaixo são apresentados os gráficos de corrente e tensão respectivamente:

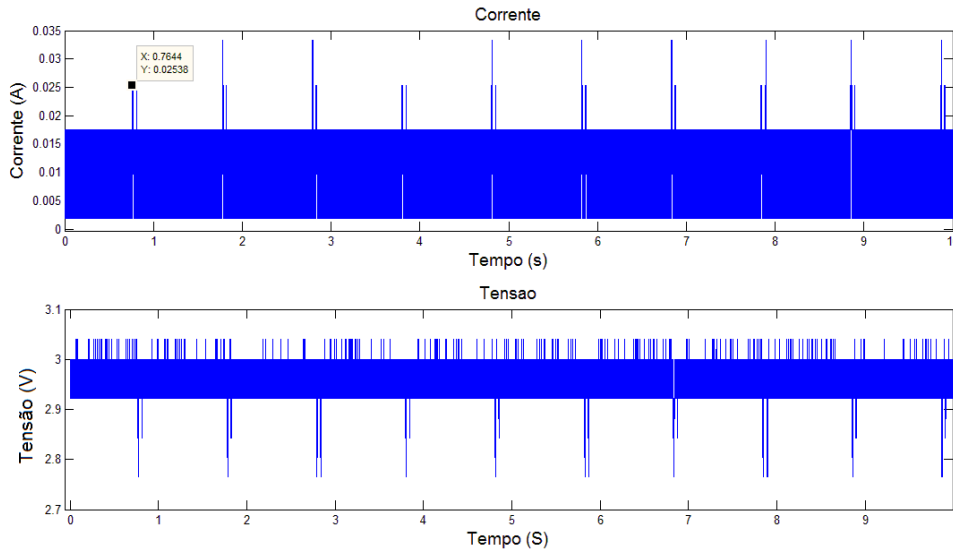


Figura 5.5- Tensão e Corrente para módulos operando com DyTEE.

Observa-se na figura 5.5 que, nos períodos ativos do DyTEE, enquanto a corrente demandada aumenta, há uma diminuição da tensão, mantendo uma baixa variação na corrente consumida pelos nós.

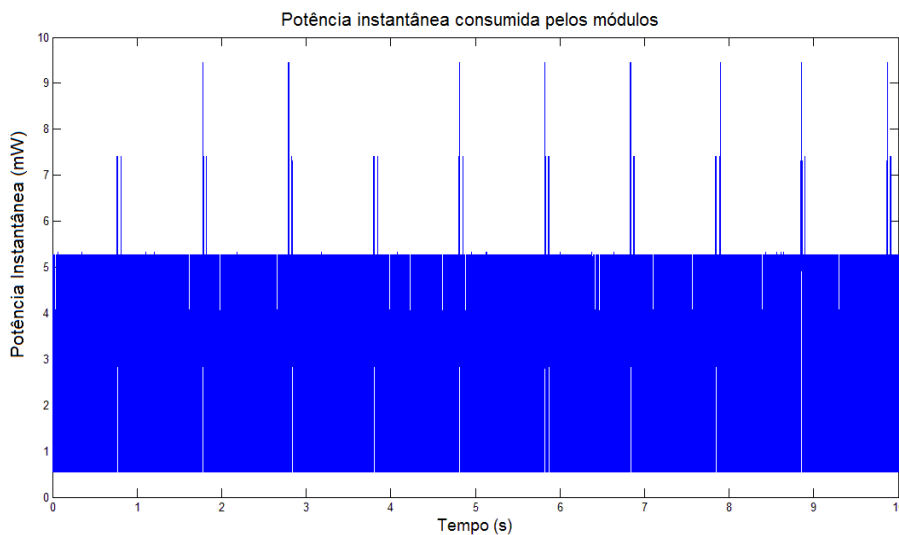


Figura 5.6- Consumo de potência pelo nó folha.

A figura 5.6 mostra que o consumo de potência pelo nó folha varia entre 5,266 mW nos períodos de dormência e 9,5 mW nos períodos ativos. O resultado está de acordo com o teste anterior apresentado na figura 3.12.

5.3 CORRENTE MEDIDA UTILIZANDO O PROTOCOLO ZIGBEE

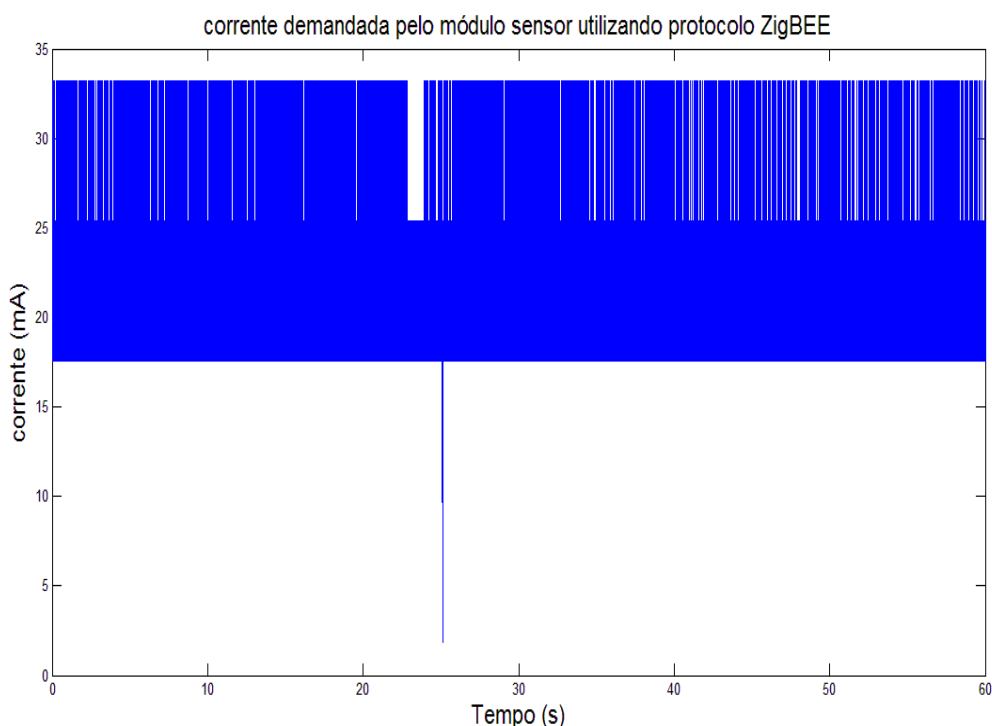


Figura 5.7- Corrente demandada pelo módulo utilizando o protocolo ZigBee.

O processo de aquisição dos dados para a curva da figura 5.7 foi o mesmo para aquisição dos dados para as medidas utilizando o DyTEE. A curva da figura 5.8 apresenta a corrente consumida na alimentação do nó sensor. É possível observar que o comportamento da corrente é muito instável oscilando entre os 16 e 33,9 mA. Para se obter um parâmetro para cálculo, tomou-se a média para a corrente consumida. O valor obtido foi 25mA.

5.4 POTÊNCIA

Para diminuir o ruído inserido no circuito ao se usar mais de uma ponta de prova, optou-se por utilizar um multímetro para medir a tensão nos terminais das baterias, pois seu valor era constante. O valor medido foi de 3,15 V possuindo pequenas variações ao longo do tempo. Então, o gráfico de potência foi obtido por meio da multiplicação dos valores obtidos para corrente pela tensão medida. A potência instantânea varia entre 52,25 mW e 104,7 mW.

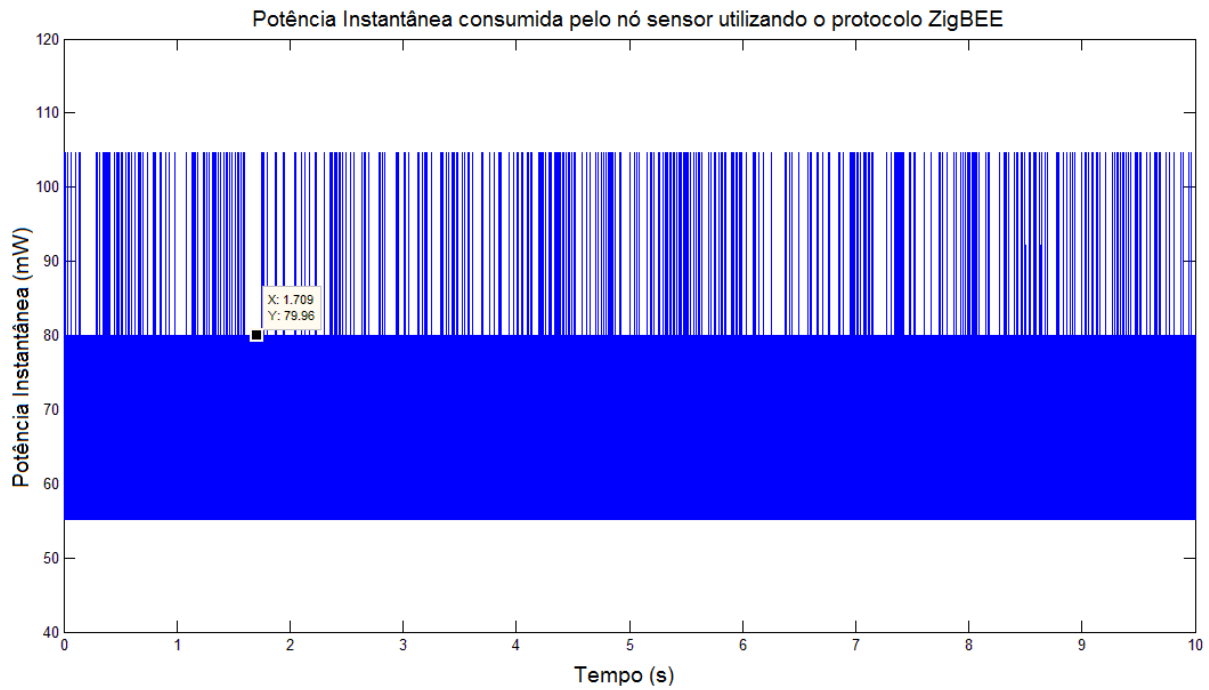


Figura 5.8- Potência consumida pelo nó sensor utilizando o protocolo ZigBee.

5.5 PROCEDIMENTO PARA ESTIMAR A VIDA ÚTIL DAS PILHAS

Tomamos as curvas de tempo de funcionamento em horas da bateria pela tensão nos terminais da bateria, conforme a figura 5.10.

A consideração na imagem do *datasheet* da pilha é de que a corrente fornecida pela pilha permanece constante durante todo o tempo em que a pilha oferece potência ao circuito. Contudo há um decréscimo no valor da tensão oferecida pela pilha

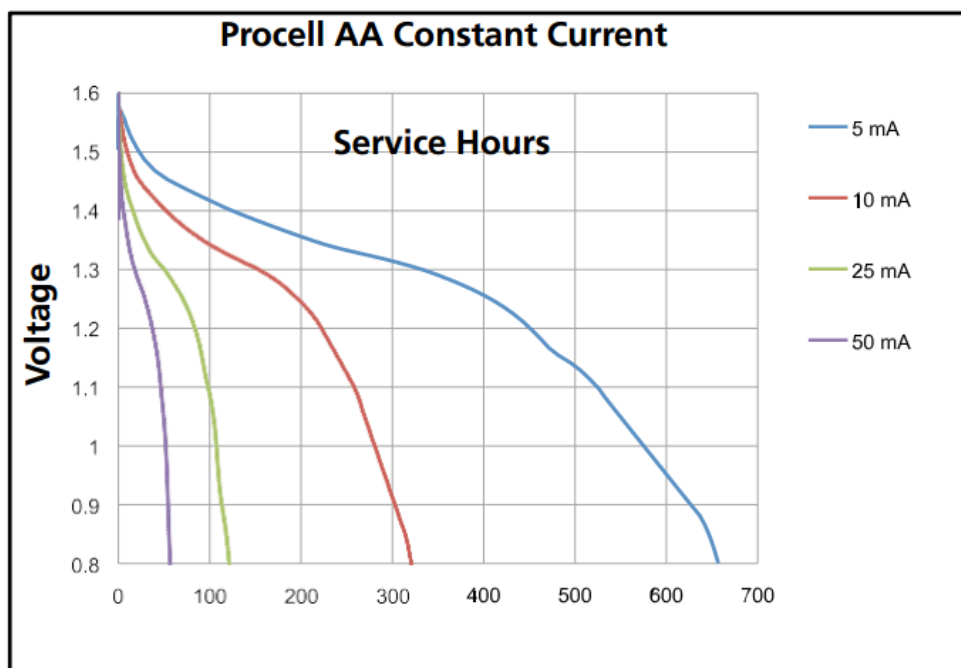


Figura 5.9- Curva de vida útil de pilhas alcalinas AA.

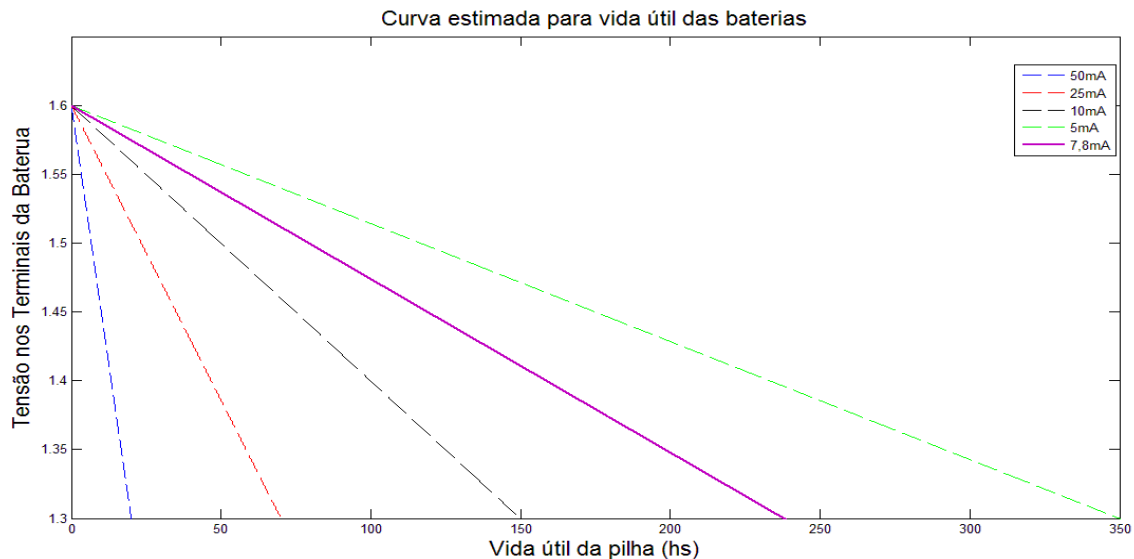


Figura 5.10- Curva estimada para vida útil das baterias.

Considerou-se na implementação da leitura analógica, uma tensão de referência interna de 2,5V. Essa tensão de referência é fornecida pelas pilhas no nó folha, portanto as pilhas devem garantir ao menos esse valor de tensão como referência dado que um valor abaixo compromete a medida de temperatura oferecida pelo Lm35.

O procedimento realizado para estimar a curva que corresponde a corrente de 7,8 mA, para o DyTEE, foi tomada considerando os intervalos entre os pontos de 1,6 V e 1,3 V, como aproximadamente linear. De acordo com o que se observa nas curvas do MatLab da fig 5.11. O valor de corrente média consumida pelo nó, *end-device*, já possui uma curva, estimando um valor de 70 horas de funcionamento.

Como o valor de corrente do DyTEE não possui uma curva correspondente na figura 5.9, tampouco na figura 5.10, foi estimada a curva correspondente à corrente consumida pelo módulo *Zigbit Channel 1.0* operando com o DyTEE. Para gerar essa curva considerou-se que o valor de corrente consumido pelo módulo encontrava-se no intervalo entre 5 mA e 10 mA. O valor de 7,8 mA é 56 % maior do que o valor de corrente de 5mA, e portanto considerou-se que esse aumento de corrente de 56 %, geraria uma diminuição de 56% de horas a menos de vida útil da bateria. Graficamente o resultado é bem considerável conforme observado no gráfico da figura 5.10

Portanto, os módulos operando com o protocolo DyTEE consomem 31,2 % da corrente consumida pelos mesmos módulos operando com o protocolo IEEE 802.15.4 (MAC). Isso reflete em uma duração, das baterias, 3,4 vezes maior em horas de funcionamento.

6 CONCLUSÃO

O DyTEE é um protocolo de comunicação muito eficiente quanto ao consumo de energia, tendo em vista a proposta por ele estabelecida de ser um protocolo gerado para atender a demanda de automação, onde não há a necessidade de uma taxa de transmissão de dados muito alta, sendo que não há perda significativa ao processo por apenas uma perda de informação, sendo pois a latência, o atraso entre a percepção da informação e o seu envio e percepção por outro dispositivo, um fator não crítico para o sistema de automação, pois a mudança na variável tal como a temperatura não ocorre de maneira muito rápida para que possa prejudicar o controle dessa variável pelo sistema. Contudo o DyTEE não possui um sistema muito independente e de alta durabilidade da rede em comunicação, tendo sido visto isso durante a criação da rede de automação e ainda de não se encontrar completamente implementado no pacote utilizado IEEE 802.15.4 MAC. A rede criada pelo protocolo ZigBee é mais estável e possui uma longa duração, conforme visto experimentalmente no contexto de automação. Contudo o alto gasto de energia não se apresenta como uma solução completamente viável num ambiente de automação, no qual a independência energética um fator muito relevante. Pela proposta apresentada o DyTEE é de fato um protocolo de comunicação mais eficiente, do ponto de vista energético, oferecendo uma autonomia de energia das pilhas de 3,4 vezes maior o padrão ZigBee.

6.1 TRABALHOS FUTUROS

O DyTEE deve ser implementado em ambientes *outdoor*, para verificar o seu desempenho em projetos de automação em redes que necessitem de uma distância física maior entre os nós. E ainda, introduzir os testes do DyTEE em outras topologias de rede, além de usar os nós na configuração de roteadores, possibilitando a expansão da rede em outros formatos e distâncias.

Reduzir ainda mais as atividades realizadas pelo micro controlador no “modo *sleep*” operando com o DyTEE de forma a reduzir o consumo de energia das baterias dos nós, observando o devido cumprimento da temporização das atividades dos nós já que se trata de um sistema operando em comunicação TDMA.

Utilizar o DyTEE em outros ambientes de comunicação sem fio não necessariamente ligados à automação, onde não haja a exigência de uma alta taxa de transmissão de dados e que seja necessária a economia de energia, tais como em áreas de difícil acesso em ambientes isolados.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] YE, W.; HEIDEMANN, J.; ESTRIN, D. An energy-efficient mac protocol for *wireless* sensor networks. In: IEEE. *INFOCOM* 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, 2002. v. 3, p. 1567-1576.
- [2] GUIMARÃES, V.G. Dissertação de Mestrado. Desenvolvimento de protocolo MAC eficiente em energia e livre de colisões para redes de sensores sem fio.
- [3] 2015. Spread Spectrum. Disponível em: <<http://sharmamonika95.blog.com/sample-page/spread-spectrum/>>.
- [4] 2015. Universidade Federal do Rio de Janeiro. Disponível em: <http://www.gta.ufrj.br/grad/10_1/ZigBee/padrao.html>.
- [5] BAUCHSPIESS, A., SOUZA, A.S. LEITE, A.A.C., RAMOS, L.M.A., PEREIRA, E.S., SANTOS, R.J. (2004) *Fuzzy Thermal Control with Remote Access for Building Automation*. 11th IFAC INCOM, Savador/Brazil, 6 pp
- [6] J. RUELA; *wireless LANs IEEE 802.11 e 802.11e*. MIEEC- 2009/10
- [7] 2015. Atmel AVR2052: Atmel BitCloud Quick Start Guide. Disponível em: <<http://www.atmel.com/images/doc8200.pdf>>.
- [8] 2015. AVR2050: BitCloud Developer Guide. Disponível em: <http://www.atmel.com/Images/Atmel-8199-BitCloud-Developer-Guide_User-Guide_AVR2050.pdf>.
- [9] ATMEL. 2015. Sistema operacional Bitcloud. Disponível em: <<http://www.atmel.com/tools/bitcloud-ZigBee.aspx>>.
- [10] ATMEL. 2015. Pacote de comunicação sem fio aberto IEEE 802.15.4 MAC. Disponível em: <http://www.atmel.com/tools/ieee802_15_4mac.aspx>.
- [11] POPOVICI, E.; MAGNO, M.; MARINKOVIC, S. Power Management Techniques for *wireless* Sensor Networks: a Review. Ireland: 2013
- [12] Power Conservation in ZigBee Networks using Temporal Control. In: *wireless* Pervasive Computing 2007, ISWPC '07 2nd International Symposium, IEEE- San Juan.
- [13] GUIMARÃES, V. G.; BAUCHPIESS, A.; MORAES, R. M. D. An energy efficient Mac protocol for *wireless* sensor networks in automation environments. In: in Proc. of XI SBAI. Fortaleza,CE, Brasil: 2013.
- [13] YE, W.; HEIDEMANN, J.; ESTRIN, D. Medium access control with coordinated adaptive sleeping for *wireless* sensor networks. *Networking, IEEE/ACM Transactions on, IEEE*, v. 12, n. 3, p. 493_506, 2004.
- [14] HEIDEMANN, J.: Medium Access Control With Coordinated Adaptive Sleeping for *Wireless* Sensor Networks
- [15] GUIMARÃES, V.G. Trabalho de Graduação Protocolo MAC eficiente em energia e livre de colisões para redes de sensores sem fio.

- [16] REZAEI, Z.; MOBININEJAD, S. Energy SAVING in *wireless* Sensor Networks, International Journal of Computer Science & Engineering Survey (IJCES) Vol.3, No.1, February 2012.
- [17] FALCONER, D.; ADACHI, F.; GUDMUNDSON, B. Time Division Multiple Access Methods for *wireless* Personal Communications, Communications Magazine, IEEE (Volume: 33 , Issue: 1).
- [18] LEE, K.; LEE, J.; KWON, W.; KO, B.; KIM, Y. Performance Evaluation of CSMA/CA Protocol in Low-Speed PLC Environments, 7th International Symposium on Power-Line Communications and its Applications. Kyoto, Japan, March 26-28, 2003. Session B4: MAC Layer.
- [19] MATOS, L.G.;MELO, N.C. Trabalho de Graduação. Economia de Energia e Gravação Remota de Dispositivos ZigBee Visando a Automação Predial.
- [20] LOUREIRO, A.F.; NOGUEIRA, J.M.; RUIZ, L.B.; MINI, R.A.F.; NAKAMURA, E.F.; FIGUEIREDO, C.M.S. Redes de Sensores sem Fio. Departamento de Ciência da Computação. Universidade Federal de Minas Gerais.
- [21] 2015. Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V. disponível em: <http://www.atmel.com/images/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf>.
- [22] 2015. *Wireless* Sensor Networks. Disponível em: <https://en.wikipedia.org/wiki/wireless_sensor_network>.

Esquemático da Placa ZigBit Channel 1.0

