

**Universidade de São Paulo**  
**Escola de Engenharia de São Carlos**  
**Departamento de Engenharia Elétrica**

**Trabalho de Conclusão de Curso**

**Análise sobre a tecnologia de rede sem  
fio Zigbee / IEEE 802.15.4**

Autor:

**Luiz Fernando Casarin Bronzatti**

Orientador

**Prof. Dr. Amílcar Careli César**

**São Carlos, Junho 2013**



**Luiz Fernando Casarin Bronzatti**

# **Análise sobre a tecnologia de rede sem fio Zigbee / IEEE 802.15.4**

Trabalho de Conclusão de Curso apresentado  
à Escola de Engenharia de São Carlos, da  
Universidade de São Paulo.

Curso de Engenharia Elétrica com ênfase em  
Eletrônica

ORIENTADOR: Prof. Dr. Amílcar Careli César

São Carlos

2013

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,  
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS  
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

C869a

Casarín Bronzatti, Luiz Fernando  
Análise sobre a tecnologia de rede sem fio Zigbee /  
IEEE 802.15.4 / Luiz Fernando Casarín Bronzatti;  
orientador Amilcar Careli César. São Carlos, 2013.

- Programa de e Área de Concentração em --  
Escola de Engenharia de São Carlos da Universidade de  
São Paulo, 2013.

1. Zigbee. 2. IEEE 802.15.4. I. Título.

# FOLHA DE APROVAÇÃO

Nome: Luiz Fernando Casarin Bronzatti

Título: "Análise sobre a tecnologia de rede sem fio Zigbee / IEEE 802.15.4"

*Trabalho de Conclusão de Curso defendido e aprovado  
em 17/06/2013,*

*com NOTA 10,0 (dez, zero), pela Comissão Julgadora:*

*Prof. Dr. Amílcar Careli César (orientador)  
SEL/EESC/USP*

*Prof. Dr. Marcelo Basílio Joaquim - SEL/EESC/USP*

*Mestre Arturo Miranda Vera - (Doutorando - SEL/EESC/USP)*

**Coordenador da CoC-Engenharia Elétrica - EESC/USP:  
Prof. Associado Homero Schiabel**

## Agradecimentos

À minha família pelo apoio e incentivo nos momentos de dificuldades e de menor progresso na elaboração do trabalho de conclusão de curso.

Ao Prof. Dr. Amílcar Careli César, pela atenção e auxílio durante o processo de definição e orientação desse trabalho.

Aos meus colegas de universidade, pelos estudos em grupo que foram importantes para se chegar até esta etapa final do curso.

## Sumário

1.	<b>INTRODUÇÃO</b>	1
2.	<b>INTRODUÇÃO AO ZIGBEE: CONCEITOS</b>	3
2.1.	<i>REDES SEM FIO: UMA INTRODUÇÃO</i>	3
2.2.	<i>WIRELESS SENSOR NETWORKS (WSNs)</i>	5
2.2.1.	Elementos de um nó-sensor	6
2.3.	<i>EXEMPLOS DE APLICAÇÕES PARA O ZIGBEE</i>	8
3.	<b>PILHA ZIGBEE / IEEE 802.15.4</b>	10
3.1.	<i>ARQUITETURA ZIGBEE</i>	10
3.2.	<i>ALGUNS CONCEITOS SOBRE UMA ARQUITETURA DE REDE</i>	12
3.2.1.	Frame	12
3.2.2.	Primitivas	13
3.2.3.	SAP (Service Access Points)	15
3.2.4.	Information Base	16
3.3.	<i>DEFINIÇÕES IEEE 802.15.4</i>	17
3.3.1.	Frequências de Rádio	17
3.3.2.	Pacotes de Dados PHY	18
3.3.3.	Modo Beacon e não-Beacon	19
3.3.4.	Estrutura superquadro	20
3.3.5.	CSMA – CA	23
3.3.6.	Varredura de Canal	26

<b>4. CAMADA NWK E REDE ZIGBEE .....</b>	<b>28</b>
<b>  4.1. ESTRUTURA DE REDE .....</b>	<b>29</b>
4.1.1. Dispositivos Zigbee.....	29
4.1.2. Topologias de rede .....	30
4.1.3. Endereçamento Zigbee .....	32
<b>  4.2. ATRIBUIÇÃO DE ENDEREÇOS.....</b>	<b>33</b>
4.2.1. Estocástico .....	33
4.2.2. Cskip .....	34
<b>  4.3. DINÂMICA DE UMA REDE ZIGBEE.....</b>	<b>36</b>
4.3.1. Formação de uma rede .....	36
4.3.2. Entrada de novos dispositivos na rede .....	37
<b>  4.4. ROTEAMENTO DAS MENSAGENS.....</b>	<b>38</b>
4.4.1. Broadcast .....	38
4.4.2. Roteamento em malha .....	38
4.4.3. Tabelas de Roteamento.....	39
4.4.4. Descoberta de rota e Custo de Link.....	39
4.4.5. Roteamento em árvore .....	42
4.4.6. Outros modos de Roteamento .....	43
<b>5. SIMULAÇÕES .....</b>	<b>44</b>
<b>  5.1. TEORIA A RESPEITO DAS SIMULAÇÕES .....</b>	<b>44</b>
5.1.1. Modo Beacon para rede na topologia em árvore .....	44

5.1.2.	Técnica CRC de checagem de erros .....	46
5.1.3.	Modulações e técnica DSSS de espalhamento espectral .....	48
<b>5.2.</b>	<b>SIMULAÇÃO DE UMA REDE WSN CLUSTER-TREE EM MODO BEACON .....</b>	<b>51</b>
5.2.1.	Procedimentos para Simulação .....	54
5.2.2.	Resultados .....	58
<b>5.3.</b>	<b>SIMULAÇÃO DO ALGORITMO CRC-16 COM SIMULINK.....</b>	<b>61</b>
5.3.1.	Procedimento para Simulação .....	62
5.3.2.	Resultados .....	63
<b>5.4.</b>	<b>SIMULAÇÃO TRANSCEPTEOR 2.4 GHz IEEE 802.15.4 COM SIMULINK.....</b>	<b>66</b>
5.4.1.	Procedimento para Simulação .....	69
5.4.2.	Resultados .....	70
<b>6.</b>	<b>CONCLUSÃO E TRABALHOS FUTUROS.....</b>	<b>74</b>
<b>6.1.</b>	<b>CONSIDERAÇÕES ACERCA DE TRABALHOS FUTUROS .....</b>	<b>75</b>
<b>Referências Bibliográficas .....</b>		<b>76</b>
<b>Bibliografias Consultadas.....</b>		<b>78</b>
<b>Apêndice .....</b>		<b>81</b>
<b>[A] CLASSIFICAÇÃO DE REDES SEM FIO .....</b>		<b>81</b>
<b>[B] PRIMITIVAS IEEE802.15.4.....</b>		<b>82</b>
<b>[C] CAMADA MAC, NWK E APL: FRAME E COMANDOS.....</b>		<b>83</b>
<b>[D] FLUXOGRAMA DE PRIMITIVAS ENTRE AS CAMADAS .....</b>		<b>86</b>
<b>[E] CONSTANTES, ATRIBUTOS E TABELAS ZIGBEE.....</b>		<b>90</b>

**[F] EQUIVALÊNCIA ENTRE SÍMBOLOS E SEGUNDOS ..... 91**

# Índice de Figuras

Figura 1 - Estrutura básica nó-sensor e receptor .....	7
Figura 2 - Arquitetura Zigbee Detalhada .....	11
Figura 3 - conjunto dos frames Zigbee-modificada da fonte(MUNK-STANDER, SKOVGAARD e NIELSEN, 2005) .....	13
Figura 4 - exemplo de fluxo de primitivas entre as camadas MAC e NWK .....	16
Figura 5 - pacote de dados da camada Física.....	18
Figura 6 - tipos de mecanismos de acesso ao canal - modo beacon e não beacon – retirada da fonte (NIEUWENHUYSE, ALVES e KOUBÂA, 2006) .....	20
Figura 7 - Transferência de dados entre coordenador e um dispositivo no modo beacon e no modo não-beacon - retirada da fonte (IEEE, 2003).....	20
Figura 8 - Estrutura superquadro .....	21
Figura 9 - Gerenciamento de espaços GTS.....	23
Figura 10- Algoritmo CSMA-CA unslotted e slotted - retirada da fonte (IEEE, 2003) .....	25
Figura 11 - Topologias de Rede .....	32
Figura 12 - Exemplo cálculo da função Cskip - exemplo retirado da fonte (NIEUWENHUYSE, ALVES e KOUBÂA, 2006) .....	36
Figura 13–Descoberta de Rota com os custos de link – figura (a): pedido-de-rota; figura (b): resposta-de-rota .....	41
Figura 14 - Esquema para intercalar transmissões beacon em redes cluster-tree da maneira fácil – modificada da fonte (KOUBA, ALVES e TOVAR).....	45
Figura 15 – Exemplo de cálculo CRC .....	47
Figura 16 - modulação O-QPSK e BPSK - retirada da fonte (XIONG, 2006) .....	49
Figura 17 – Diagrama de fase da modulação O-QPSK.....	50
Figura 18 - Exemplo do funcionamento da técnica de multiplicação dos bits por uma sequência PN .....	50
Figura 19 – Janela do simulador Worst-Case Dimensioning of Cluster-Tree Wireless Sensor Networks .....	52
Figura 20 - Cenário 1 da simulação de uma rede cluster-tree no modo beacon.....	54
Figura 21 - Cenário 2 da simulação de uma rede cluster-tree no modo beacon.....	55

<b>Figura 22 - Cenário 3 da simulação de uma rede cluster-tree no modo beacon.....</b>	<b>56</b>
<b>Figura 23 - Delay e Buffer do roteador na profundidade 1 .....</b>	<b>57</b>
<b>Figura 24 - Resultado da simulação para o Cenário 1 .....</b>	<b>59</b>
<b>Figura 25 - Resultado da simulação para o Cenário 2 .....</b>	<b>60</b>
<b>Figura 26 - Resultado da simulação para o Cenário 3 .....</b>	<b>60</b>
<b>Figura 27 – Simulação em Simulink do funcionamento da técnica CRC .....</b>	<b>61</b>
<b>Figura 28 - Exemplo de como funciona os blocos CRC-N Generator e Selector .....</b>	<b>62</b>
<b>Figura 29 - Três Osciloscópio da simulação CRC .....</b>	<b>64</b>
<b>Figura 30 - Palavra binária transmitida e recebida em um canal com 5% de probabilidade de erro de bit .....</b>	<b>64</b>
<b>Figura 31 - Bits da CRC no receptor calculados sobre a sequência [b15r b14r ... b0r]no receptor. Transmissão sobre um canal com 5% de probabilidade de erro de bit .....</b>	<b>65</b>
<b>Figura 32- Bits da CRC no receptor calculados sobre a sequência [b15r b14r ... b0r + c15r c14r ... c0r] no receptor.Transmissão sobre um canal com 5% de probabilidade de erro de bit.....</b>	<b>65</b>
<b>Figura 33 - Modelo em Simulink de um Transmissor Zigbee.....</b>	<b>66</b>
<b>Figura 34 - Modelo em Simulink de um Receptor Zigbee .....</b>	<b>66</b>
<b>Figura 35 - Simulação em Simulink de uma transmissão Zigbee sobre um canal AWGN.....</b>	<b>68</b>
<b>Figura 36 - Janela de Configuração dos blocos Random Integer Generator (a) e PN Sequence Generator (b)..</b>	<b>70</b>
<b>Figura 37 - Influência da relação sinal/ruído sobre a probabilidade de erro dos bits.....</b>	<b>71</b>
<b>Figura 38 – Osciloscópio e sinais da simulação do transceptor.....</b>	<b>71</b>
<b>Figura 39 - Sinais visualizados no Osciloscópio para um canal com uma relação sinal/ruído alta .....</b>	<b>72</b>
<b>Figura 40 - Sinais visualizados no Osciloscópio para um canal com uma relação sinal/ruído muito baixa .....</b>	<b>73</b>
<b>Figura 41 - Alcance típico das redes WPAN, WLAN, WMAN, WWAN – retirada da fonte (THORNHILL e IKRAM, 2010) .....</b>	<b>81</b>
<b>Figura 42 - Frame AF da subcamada de framework de aplicacão e campo controle do frame .....</b>	<b>83</b>
<b>Figura 43- Frame da subcamada de suporte a aplicação .....</b>	<b>83</b>
<b>Figura 44–Frame da camada NWK e campo controle do frame .....</b>	<b>84</b>
<b>Figura 45–Frame da camada MAC e campo controle de frame .....</b>	<b>84</b>

**Figura 46 - Primitivas de formação para rede – retirada da fonte (ZIGBEE ALLIANCE, 2008) .....87**

**Figura 47 - Primitivas para entrada de um dispositivo na rede – retirada da fonte (ZIGBEE ALLIANCE, 2008) .88**

**Figura 48 - Primitivas no roteador/coordenador sobre a entrada de novos dispositivos na rede – retirada da fonte (ZIGBEE ALLIANCE, 2008) .....89**

## Índice de Tabelas

Tabela 1 - Primitivas entre PHY e MAC.....	15
Tabela 2 - ScanDuration para 2,4 GHz .....	27
Tabela 3 - Parâmetros da rede para o Cenário 1 .....	55
Tabela 4 - Parâmetros da rede para o Cenário 2 .....	56
Tabela 5 - Parâmetros da rede para o Cenário 3 .....	57
Tabela 6 - Valores dos parâmetros WPAN de saída.....	58
Tabela 7 - Valores levantados com a simulação para o gráfico Pb X Eb/No .....	70
Tabela 8 - Primitivas da camada Física .....	82
Tabela 9 - Primitivas da camada MAC .....	82
Tabela 10 - Comandos MAC .....	85
Tabela 11 - Comandos NWK.....	86

## **Resumo**

O Zigbee é uma das principais e mais conhecidas tecnologias para comunicação sem fio. Além de mobilidade, característica típica de dispositivos sem fio, o Zigbee também possibilita dispositivos móveis de tamanho e peso reduzido. Estas características, mais o baixo consumo de energia alcançado por dispositivos Zigbee, são essenciais em aplicações para as áreas de monitoramento remoto, controle e sensoriamento como, por exemplo, para as redes WSN (*Wireless Sensor Networks*). Para conseguir um baixo consumo de dispositivos sem fio, que podem ser alimentados com bateria, o Zigbee usa as camadas Física (PHY) e MAC da especificação IEEE 802.15.4 como fundação para sua pilha protocolar. A compreensão da especificação IEEE 802.15.4 é parte importante para o entendimento dos mecanismos para transmissão sem fio e para o entendimento da própria tecnologia Zigbee. Assim, neste trabalho foram descritos conceitos ligados a esta especificação, como os métodos de tentativa de acesso ao canal de comunicação, bandas de frequências usadas para transmissão e a formatação de um pacote de dados. Ainda, com a ajuda do ambiente computacional Matlab/Simulink foi possível clarificar e analisar outros conceitos IEEE 802.15.4: método CRC usado para verificar erros em transmissões, e o funcionamento de um transceptor (transmissor + receptor). Acima da pilha IEEE 802.15.4 (PHY e MAC) está a camada de rede (NWK) do Zigbee que habilita a interação dinâmica entre dispositivos sem fio e a formação de redes nas topologias em árvore ou em malha. No trabalho, também, foram explicados a dinâmica da rede Zigbee: a organização dos dispositivos na rede, a atribuição de endereço para cada dispositivo na, a formação de uma nova rede e as estratégias de roteamento de uma mensagem de um dispositivo a outro.

**Palavras Chaves:** Zigbee, IEEE 802.15.4, pilha.

## **Abstract**

Zigbee is one of the most known and important wireless technologies. In addition to mobility, typical feature of wireless devices, Zigbee also enables reduced size and weight mobile devices. These features, together with low energy consumption reached by Zigbee devices, are essential for remote monitoring, control and sensor applications, such as WSN (Wireless Sensor Networks). To provide devices with low energy consumption that can be battery powered, Zigbee uses Physical (PHY) and MAC layers specified by IEEE 802.15.4 as the foundation of its protocol stack. The comprehension of IEEE 802.15.4 specification is an important part for understanding wireless transmission mechanisms and understanding Zigbee technology. Thus, this paper described concepts related to this specification such as methods for access the communication channel, frequency bands used for transmission and the structure of data packet. Yet, with Matlab/Simulink computational tool, it was possible to clarify and analyze others IEEE 802.15.4 concepts: CRC method used for checking transmission errors and the operation of a transceiver (transmitter + receiver). Above IEEE 802.15.4 stack (PHY and MAC), it is located Zigbee network layer (NWK), layer that enables a dynamic interaction between wireless devices and the formation of tree and mesh networks. In this paper, it was also explained the dynamic of a Zigbee network: the organization of device in the network, address assignment for each device in the network and the formation of a new network and strategies for routing a message from one device to another.

**Keywords:** Zigbee, IEEE 802.15.4, stack.

## Acrônimos

<b>μC:</b> Microcontrolador	<b>ISP:</b> Internet Service Provider
<b>ADC:</b> Analogic-Digital Converter	<b>LAN:</b> Local Area Network
<b>AP:</b> Access Point	<b>LDO:</b> Low Drop Out
<b>APL:</b> Application	<b>MAC:</b> Media Access Control,
<b>BPSK:</b> Binary Phase-Shift Keying	<b>MAN:</b> Metropolitan Area Network
<b>BWA:</b> Broadband Wireless Access	<b>MHR:</b> MAC Header
<b>CAP:</b> Contention Access Period	<b>MPDU:</b> MAC Protocol Data Unit
<b>CCA:</b> Clear Channel Assessment	<b>MSDU:</b> MAC Service Data Unit
<b>CFP:</b> Contention Free Period	<b>NWK:</b> Network
<b>CI:</b> Circuito Integrado	<b>O-QPSK:</b> Offset-Quadrature Phase-Shift Keying
<b>CRC:</b> Cyclic Redundancy Check	<b>OSI:</b> Open System Interconnection
<b>CSMA:</b> Carrier Sense Multiple Access	<b>P2P:</b> Peer-to-Peer
<b>CSMA – CA:</b> Carrier Sense Multiple Access - Collision Avoidance	<b>PAN:</b> Personal Area Network
<b>DSSS:</b> Direct Sequence Spread Spectrum	<b>PCI:</b> Placa de Circuito Impresso
<b>EB:</b> Estação Base	<b>PHY:</b> Physical
<b>ED:</b> Energy Detection	<b>PIB:</b> PAN Information Base
<b>FCC:</b> Federal Communication Commission	<b>PN:</b> Pseudo Noise
<b>FCS:</b> Frame Check Sequence	<b>POS:</b> Personal Operation Space
<b>FFD:</b> Full Function Device	<b>PPDU:</b> PHY Protocol Data Unit
<b>GPIO:</b> General Purpose In/Out	<b>PSDU:</b> PHY Service Data Unit
<b>GTS:</b> Guaranteed Time Slot	<b>RF:</b> Rádio Frequência
<b>ID:</b> Identifier	<b>RF4CE:</b> Radio Frequency for Consumer Electronics
<b>IEEE:</b> Institute of Electrical and Electronics Engineers	<b>RFD:</b> Reduced Function Device
<b>ISM:</b> Industrial, Scientific and Medical	<b>SFD:</b> Start of Frame Delimiter

**SG:** Study Group

**SHR:** Synchronization Header

**SiP:** System in Package

**SoC:** System on Chip

**SPI:** Serial Peripheral Interface

**TAG:** Technical Advisory Group

**UART:** Universal Asynchronous Receiver/Transmitter

**USART:** Universal Synchronous/Asynchronous  
Receiver/Transmitter

**UWB:** Ultra Wideband

**VLSI:** Very Large Scale Integration

**WBAN:** Wireless Body Area Network

**WG:** Working Group

**WPAN:** Wireless Personal Area Network

**WSN:** Wireless Sensor Network

## 1. INTRODUÇÃO

A primeira especificação de uma rede sem fio WPAN (*Wireless Personal Area Network*) foi o padrão IEEE 802.15.4 concluído em Outubro de 2003 por um comitê do IEEE (*Institute of Electrical and Electronic Engineers*). Esta especificação define as características das camadas Física (PHY) e de Controle de Acesso ao Meio (MAC) para interconexão entre dispositivos em uma rede WPAN.

A IEEE 802.15.4 especifica um padrão de baixa taxa de transmissão que atende aos requisitos de baixo consumo de energia, simplicidade e baixo custo, com objetivo de aplicação em sistemas de sensoriamento e controle remoto.

Geralmente são criadas novas alianças entre grandes companhias para desenvolver e acrescentar novos protocolos acima das camadas PHY e MAC que, normalmente, são padronizadas pelo IEEE. Uma dessas alianças criadas foi a *Zigbee Alliance*.

A *Zigbee Alliance* adotou camadas PHY e MAC especificadas na especificação IEEE 802.15.4, acrescentando mais duas camadas sobre estas duas: a camada de Rede (NWK) e a de Aplicação (APL).

Os objetivos da Aliança Zigbee podem ser resumidos na declaração feita por Bob Heile, presidente da aliança

Muitas pessoas não percebem que uma revolução já está em andamento nos negócios, e em nossas casas. Uma grande ênfase está sendo colocada para reduzir a perda de recurso natural e em se tornar exponencialmente mais eficiente. Para alcançar as metas de eficiência global, nós devemos ter uma infraestrutura extensiva, inteligente, fácil de implantar e de baixo custo para monitorar e gerenciar o que nos rodeia. O Zigbee é uma solução tecnológica sólida para esta tarefa monumental, e trás com ele outros benefícios e melhorias do estilo de vida, tais como segurança e comodidade. (HEILE, p. 10)

As frequências de ondas de rádios empregadas pelo IEEE 802.15.4 são de 2,4 GHz, 915 MHz, 868 MHz. Nestas três frequências de rádio não há licenciamento, já que a banda é reservada para fins de Instrumentação, Científica e Médica (ISM).

Muitas outras tecnologias também operam na banda ISM, devido ao benefício do uso global e sem a burocracia de licenciamento. O Bluetooth sobre IEEE 802.15.1, o Wi-Fi

sobre 802.11a/b/g e o *Ultra-Wideband* (UWB) sobre 802.15.3 e alguns protocolos proprietários são exemplos de tecnologias que operam em 2,4 GHz.

A estrutura desse trabalho está organizada em sete capítulos, conforme descritos na sequência:

**O capítulo 2** apresenta uma introdução sobre as redes sem fio e, ao longo do capítulo, são abordados alguns temas dentro do universo de redes sem fio, que são importantes para caracterizar o Zigbee. Na subseção 2.3 foram apresentadas exemplos de aplicações com o Zigbee.

**No capítulo 3** apresenta-se a fundação do protocolo Zigbee constituída pelas camadas física (PHY) e de controle de acesso ao meio (MAC), que são especificados pelo padrão IEEE 802.15.4.

**O capítulo 4** traz os principais detalhes da funcionalidade da camada de rede, que define as características e funcionamento da rede sem fio Zigbee.

**O capítulo 5** é destinado a simulações que convalidam parte da teoria Zigbee.

**O capítulo 6** considerações finais e trabalhos futuros.

## 2. INTRODUÇÃO AO ZIGBEE: CONCEITOS

### 2.1. Redes sem fio: uma introdução

Redes sem fio são compostas por dispositivos que trocam dados por meio de ondas de rádio, infravermelho ou qualquer outro tipo de onda eletromagnética, sem a necessidade de uma conexão física por cabo.

O uso de ondas eletromagnéticas para transmissão e recepção de dados só foi possível graças à publicação das equações que descrevem os fenômenos elétricos e magnéticos (eletromagnéticos) por Clerk Maxwell, em 1861 e 1862. As equações foram provadas corretas com a comunicação por rádio pelo físico alemão Heinrich Hertz, em seu laboratório.

A possibilidade de enviar informações pelo ar, a longa distância, foi ratificada em 1901 com a transmissão transatlântica de ondas de rádio por Marconi. Desde então, a comunicação sem fio foi evoluindo rapidamente, consolidando-se com a criação da primeira rede sem fio, chamada ALOHANET, que conectava os vários campi da Universidade do Hawaii.

Após os primeiros sucessos ainda no século XX, muito progresso ocorreu ao longo dos séculos.(Leitura Geral).

- 1901 - Telégrafo sem fio é usado para comunicação transatlântica por Marconi.
- 1920 - Inaugurou-se a primeira rádio difusora comercial do mundo, em Pittsburgh.
- 1922 - Inaugurada a primeira rádio difusão no Brasil, no Rio de Janeiro.
- 1947 - Invenção do transistor.
- 1948 - Invenção dos Telefones Celulares.
- 1957 - Lançamento do primeiro Satélite Artificial (Sputnik) do mundo.
- 1989 - A *Federal Communication Commission (FCC)* autorizou o uso de três faixas de frequência.
- 1990 - A IEEE instaurou um comitê para criar um novo padrão para conectividade sem fio.
- 2000 - Surgiram os primeiros “hotspots” para acesso da internet com as redes IEEE 802.11.

As tecnologias sem fio vêm sendo contribuídas com a melhorada integração de componentes eletrônicos, atingindo hoje o nível *Very Large Scale Integration* (VLSI). A redução na escala contribui para a redução do consumo e do tamanho dos dispositivos eletrônicos, mas sem a perda em eficiência ou capacidade de processamento destes.

A comunicação sem fio é uma aplicação da ciência e tecnologia vital no mundo moderno, devido a demanda de conectividade e fluxo de informações constantes. Ela tem sido uma das áreas mais estudadas em sistemas de comunicação, pois ainda não se chegou ao estado da arte. Um dos objetivos dos pesquisadores é desenvolver redes sem fio tão robustas e eficientes quanto às redes cabeadas e com a mesma taxa de transmissão, podendo, assim, substituí-las em muitas ocasiões, já que as vantagens da rede sem fio, sobre a rede cabeada, são muitas.

Uma destas vantagens reside sobre as variáveis de tempo e custo. Em redes cabeadas, o custo e o tempo exigidos para a instalação e manutenção de uma grande quantidade de cabos são elevados, dificultando a configuração de sistemas. Outra vantagem é a flexibilidade e mobilidade propiciada por dispositivos sem fio. Não se precisa ir longe para encontrar um exemplo de como um dispositivo sem fio substituiu, por quase completo, um dispositivo a cabo. Este exemplo é o do telefone residencial. É difícil encontrar uma casa, hoje em dia, em que o telefone seja conectado por cabo ao terminal base, haja vista que os telefones sem fio são mais práticos e podem ser deixados qualquer local da casa, oferecendo uma mobilidade maior.

Além destas vantagens, em locais de difícil acesso ou instalação de cabos, as redes sem fio se tornam uma opção ótima.

Também é aconselhável o uso de equipamentos/dispositivos portáteis que estão em constante movimento ou estão em contato direto com produtos químicos. Por exemplo, em um sensoriamento de nível de combustível, o combustível poderia danificar os fios de transmissão. Um cabo parcialmente imerso em um líquido condutor de eletricidade poderia causar um curto-circuito ou choque.

Apesar de muitas vantagens, principalmente ao que se refere à mobilidade, a comunicações por ondas-eletromagnéticas estão mais sujeitas a erros do que a comunicação cabeada. O sinal eletromagnético está sujeito à dispersão, reflexão e difração que resulta não só em atenuação da sua potência, mas também em possíveis erros na recepção do

sinal. Devido a estes fenômenos da onda, várias cópias da onda original chegam ao receptor, em diferentes instantes de tempo, causando interferências destrutivas ou construtivas. Se o atraso na recepção das várias cópias da onda for muito grande, elas irão se sobrepor, alterando o “formato” original do sinal e causando erros. Após ocorrer a sobreposição de onda é difícil reconstruir o sinal original. Quanto maior a taxa de transferência de dados, *bits/s*, maior é a probabilidade de interferência.

Outras adversidades são: a redução drástica do sinal com a distância e maior probabilidade de interferências entre sinais na mesma frequência.

## 2.2. Wireless Sensor Networks (WSNs)

Sensores são dispositivos eletro-eletrônicos sensíveis a uma propriedade física do meio, e com isso são capazes de transformar esta propriedade em um sinal elétrico (corrente ou tensão). Os sensores podem ser usados para medição de uma variedade de propriedades físicas/químicas do meio ambiente: umidade, temperatura, movimento, pressão, luz, entre muitos outros. Alguns sensores são sensíveis à quantidade de estímulo e são capazes de mensurá-lo analogicamente. Outros, por sua vez, não são sensíveis a quantidade, e sim somente a presença de um estímulo, e por isso são empregadas como gatilhos para outro evento – por exemplo, um detector de movimento, indicará que a luz do ambiente deverá ser ligada.

Redes sem fio cujo fluxo de dados seja formado por medições de sensores são chamadas, em inglês, de *Wireless Sensor Networks (WSN)*. Toda rede sem fio em que os dispositivos que a compõem são usados para sensoriamento é considerada uma WSN, independentemente do tamanho, topologia ou qualquer outro parâmetro dela. Nestas redes, os dispositivos de sensoriamento variam de sistemas computacionalmente potentes a microcontroladores simples.

As aplicações para a WSN também são vastas e variáveis, desde aquelas que demandam grande largura de banda e altas taxas de transferência de dados, quanto às com baixa largura de banda e baixa taxa de transferência de dados. Porém é mais comum que as WSNs sejam formadas por pequenos nós-sensores autônomos com funcionalidade básica, sem muito poder computacional, pouca memória, mas que, mesmo assim, são capazes de medir e depois adequar o sinal medido para transmiti-lo via onda. Entretanto isto não é uma regra.

Algumas das características das WSNs são:

- Capacidade de agregar um grande número de nós (dispositivos). No protocolo Zigbee a capacidade máxima é de mais de 65.000 nós em uma única rede.
- A rede é requisitada somente quando é acionado um evento de medição, podendo, assim, haver longos períodos de inatividade.
- A segurança é limitada se comparada com outros tipos de rede sem fio.
- São altamente dinâmicas, com frequentes mudanças de topologia e de quantidade de nós.
- São heterogêneas, com diferentes tipos de dispositivos sensores que se mantêm interconectados através de um protocolo em comum.
- São auto-organizáveis.

### 2.2.1. Elementos de um nó-sensor

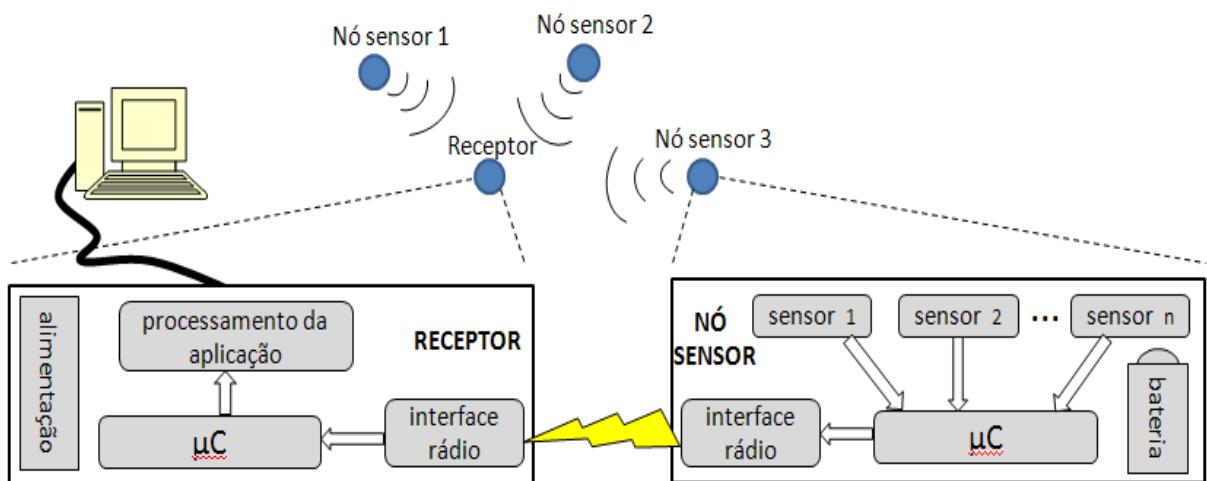
O papel dos sensores é o de apenas traduzir, de forma precisa e confiável, um atributo físico em energia elétrica, não sendo capaz, ele próprio, de fazer o processamento computacional e de transmitir os dados das medições. Para isso, outros circuitos específicos são agregados aos sensores, para se formar o dispositivo chamado de nó-sensor.

Os principais circuitos que completam a funcionalidade de um nó-sensor são: microprocessador (CPU), fonte de alimentação e interface de rádio. A CPU executa a parte computacional de análise dos dados medidos pelos sensores; a interface de rádio faz a intermediação entre a CPU e o meio de transmissão, recebendo e transmitindo os dados; e a fonte de alimentação entrega toda a energia necessária para os outros circuitos funcionarem.

Para poupar energia, muitos nós-sensores entram em modo de “suspenção”, reduzindo o consumo de energia quando não estão transmitindo ou recebendo dados. Nós-sensores tentam poupar energia, quando possível, se alimentados com bateria ou pilha. Entretanto, na rede há dispositivos que precisam estar constantemente ativos, coletando as

medições feitas pelos nós-sensores e, para tais dispositivos, o consumo de energia é maior que nos nós-sensores. Se este consumo for alto é melhor que tais dispositivos sejam alimentados por uma fonte fixa, que forneça energia ininterrupta como, por exemplo, a rede pública de energia elétrica.

A Figura 1 exibe a estrutura básica entre um nó-sensor e um nó receptor.



**Figura 1 - Estrutura básica nó-sensor e receptor**

### **2.3. Exemplos de Aplicações para o Zigbee**

(KHAN, MEHMET e KARAMI, 2008) analisaram o desempenho de uma rede de sensores sem fio Zigbee em uma aplicação de monitoramento remoto de pacientes. Eles integraram uma rede WBAN (*Wireless Body Area Network*), responsável por monitorar o paciente, com a intranet do hospital. Também deixaram à possibilidade de estender a rede à internet, o que propiciaria o monitorando a longas distâncias. No sistema proposto por eles, os nós-sensores Zigbee acoplados ao paciente transmitem, sem fio, os sinais do monitoramento para um outro nó que está conectado a um computador. O computador é responsável por armazenar todos os dados coletados do paciente e, posteriormente, os envia para a rede intranet do hospital, por onde os diversos aparelhos eletrônicos usados pela equipe médica estão conectados.

Em uma aplicação diferente da citada acima, (NOORANI) empregou uma rede de sensoriamento Zigbee com o propósito de monitorar o vazamento de água em encanamentos residenciais, trazendo benefícios econômicos, conhecimento do estado de conservação dos canos, diminuição do desperdício de água e redução do risco de contaminação. A rede é composta por sensores de temperatura e de detecção de umidade espalhados do lado de fora do encanamento residencial.

(RASIN e ABDULLAH) também propuseram o monitoramento dos recursos hídricos, porém o intuito era monitorar a qualidade de água, mantendo-a sempre dentro do padrão exigido para o consumo humano. Com uma rede de sensores, detecta-se mais rapidamente uma eventual contaminação, evitando que a água contaminada chegue à população. Medições automatizadas da qualidade da água facilitam o monitoramento em lugares de difícil acesso, como em encanamentos de água. Neste projeto foram implementados nós-sensores formados por sensores de acidez-alcalinidade (pH), temperatura e turbidez, que enviam os dados coletados para uma estação base conectada a um computador responsável pela análise dos dados. Os experimentos montados por eles mostraram uma capacidade de operação dos nós-sensores por 12 horas contínuas ligados a uma bateria recarregável de 120 mAh.

(REYES, MONJE e SANTOS) desenvolveram um protótipo, muito útil para o dia-a-dia, que consistia na automação do processo de ligar ou desligar a luz conforme se detectava a entrada ou saída de pessoas do ambiente. Os pesquisadores destacaram que neste tipo de projeto não há a necessidade de altas taxa de transferência de dados, pois as infor-

mações a serem transmitidas são apenas as dos estados possíveis de uma lâmpada: ligado ou desligado, indicando, respectivamente, quando o sensor de micro-ondas captava ou não algum movimento. A rede foi composta por três tipos de nós Zigbee: sensor, coordenador e atuador. Na detecção de um movimento, o sensor envia um sinal ao coordenador que, por sua vez, envia um sinal ao atuador, que acende a luz. No entanto, quando não se detecta movimento no ambiente, por um determinado período, a luz é desligada. O atuador foi acoplado à lâmpada, controlando a luminosidade do ambiente.

(SAKHARE e DESMUKH, 2012) criaram uma rede WSN Zigbee, capaz de gerenciar o consumo de energia. A idéia deles consistia em estabelecer um *threshold* (limiar) de máxima potência permitida aos equipamentos elétricos. A implementação foi feita em uma topologia estrela, com um dispositivo central controlando o nível máximo de potência, e os sensores monitorando a potência gasta pelos equipamentos. Se o nível de potência do equipamento estivesse abaixo do *threshold*, ele seria desligado. Os pesquisadores ressaltaram estar em pauta, atualmente, o uso mais consciente da energia, devido à insuficiência das fontes energéticas em atender a demanda da população. Os setores industriais também estão sempre em busca alternativas para se eliminar ou reduzir os desperdícios de energia, que se refletem em grande parte de seus custos. Assim, uma aplicação como esta oferece uma saída para reduzir os custos operacionais.

A tecnologia Zigbee foi escolhida nestes projetos devido à proposta de baixo consumo de energia elétrica. Circuitos sensores Zigbee são facilmente alimentados por baterias, mantendo o gasto energético reduzido. Nota-se que muitas destas aplicações estão voltadas para sensoriar, monitorar e controlar o ambiente, e os requisitos de tais aplicações são, na maioria das vezes, satisfeitos com as características da tecnologia Zigbee.

### 3. PILHA ZIGBEE / IEEE 802.15.4

#### 3.1. Arquitetura Zigbee

A arquitetura básica do protocolo Zigbee, baseada no modelo *Open Systems Interconnection* (OSI), é formada por um conjunto de blocos chamados de Camadas. As três primeiras camadas Zigbee exercem funções similares as das camadas 1-3 (física, de controle do meio e de rede) OSI, enquanto a camada de aplicação (APL) engloba as funções das camadas 4-7 (transporte, sessão, apresentação e aplicação) OSI.

O OSI não é considerado como uma arquitetura e sim um modelo, pois ele apenas detalha o que cada camada faz, mas não especifica os procedimentos. Uma arquitetura de rede deve definir, claramente, os serviços e protocolos de cada camada.

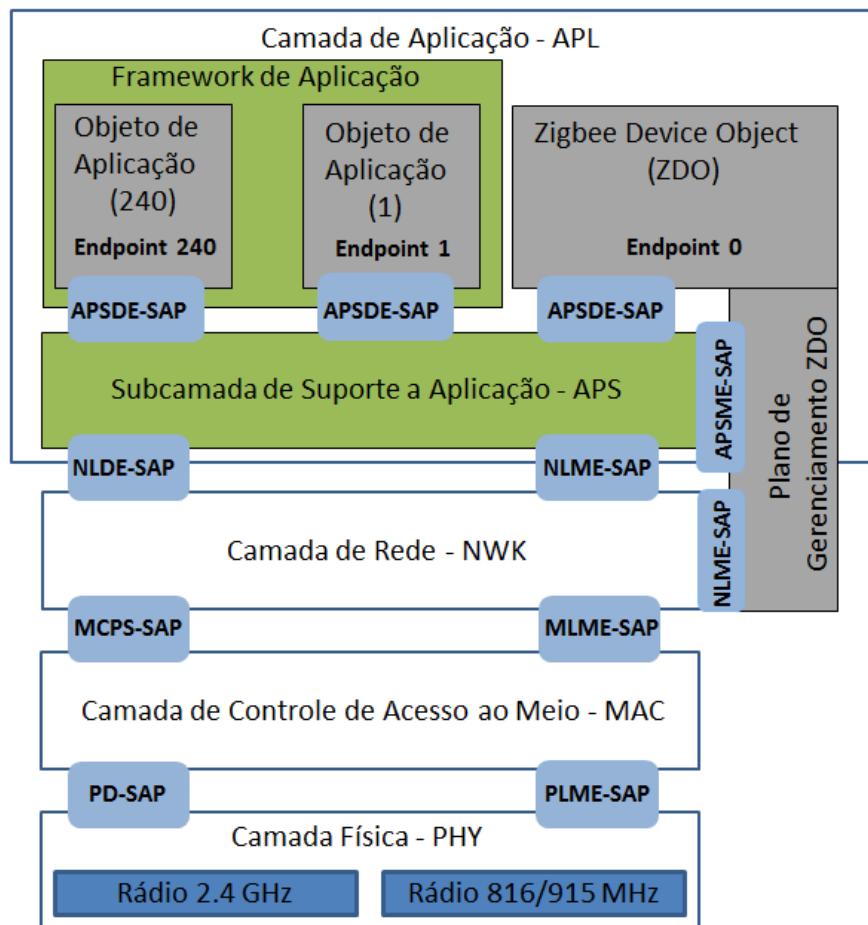
A Figura 2 mostra arquitetura da pilha Zigbee com suas respectivas camadas: Camada de Aplicação (APL), Camada de Rede (NWK), Camada de Controle de Acesso ao Meio (MAC) e Camada Física (PHY).

**Camada APL:** é definida pela Aliança Zigbee e é a camada no topo da pilha protocolar. Contém as aplicações que darão funcionalidade para o dispositivo, podendo haver até 240 aplicações em um mesmo dispositivo. A camada APL é subdividida em três partes: Subcamada de Suporte a Aplicação (APS), *Zigbee Device Object* (ZDO) e *Framework* de Aplicação.

**Camada NWK:** também é especificada pela Aliança Zigbee. Interliga o nível de aplicação Zigbee com as camadas IEEE 802.15.4. É ela que fornece as funcionalidades da rede Zigbee, como modos de roteamento, segurança e estrutura da rede e formação de uma nova rede. A camada NWK executa tarefas como formação de uma nova rede, associação e dissociação de dispositivos da rede, atribuição de endereços e descoberta de rotas.

**Camada MAC:** é especificada pela IEEE 802.15.4. Ela é responsável por prover suporte à transmissão de *frames beacons* e de reconhecimento (*acknowledgment*) e sincronização da rede.

**Camada PHY:** é a camada no nível mais baixo da pilha Zigbee e também é definida pela IEEE 802.15.4. Ela faz a interface entre as camadas acima e o meio de transmissão, montando o pacote de dados no formato padrão exigido para transmissão. Também verifica se o canal de comunicação está livre para o dispositivo transmitir. Definem-se dois tipos de camada PHY, que se diferenciam pelo espectro de frequência de operação: uma em 2,4GHz e outra em 868/915 MHz.



**Figura 2 - Arquitetura Zigbee Detalhada**

### 3.2. Alguns Conceitos sobre uma arquitetura de rede.

Dispositivos Zigbee trocam informações em pacotes de dados. Um pacote de dados é a menor unidade de informação transmitida pelo ar. O pacote limita a quantidades de bits enviados de uma vez e pode ser dividido em três partes: o cabeçalho, o corpo da mensagem e o rodapé.

O cabeçalho contém, entre outras coisas: informações sobre o conteúdo no corpo da mensagem, bits de sincronização, endereço fonte, endereço destino, tamanho total do pacote e instruções de controle. Cada camada Zigbee adiciona seu próprio cabeçalho ao pacote, aumentando, consideravelmente, o tamanho total do pacote a ser transmitido. Apesar deste aumento, o cabeçalho é importante para que a mensagem seja lida e entregue corretamente.

O corpo da mensagem é o local em que estão os dados, propriamente ditos, que uma camada x de um dispositivo quer enviar para a mesma camada x de um outro dispositivo. Normalmente, o corpo é constituído por dados da camada de aplicação, porque são as aplicações, geralmente, que tem algo a ser transmitido.

Fazendo uma analogia, uma carta dentro do envelope equivale ao corpo da mensagem, enquanto o está escrito no envelope - endereço destino e do remetente - corresponde ao cabeçalho.

O rodapé contém informações que permitem verificar erros no pacote recebido. O pacote IEEE 802.15.4 implementa a checagem de erros de bits pelo método CRC - 16 (*Cyclic Redundancy Check*).

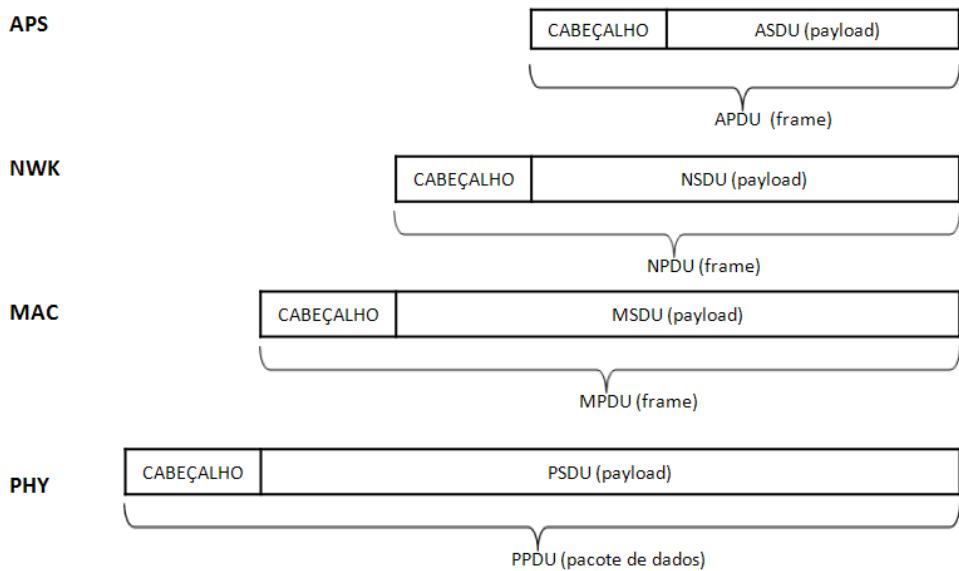
#### 3.2.1. Frame

Enquanto o pacote de dados representa o formato dos dados no ar, o *frame* representa o formato dos dados em cada camada. O *frame* é a unidade de informação básica de cada camada e, no Zigbee, ele é chamado de *Protocol Data Unit*. Os *frames* da cada camada são:

- Camada APL:
  - Subcamada de suporte de aplicação (APS): **APDU - Application Protocol Data Unit**.
- Camada NWK: **NPDU -Network Protocol Data Unit**.

- Camada MAC: **MPDU -MAC Protocol Data Unit.**
- Camada PHY: **PPDU -PHY Protocol Data Unit.**

A Figura 3 ilustra os *frames* Zigbee.



**Figura 3 - conjunto dos frames Zigbee-modificada da fonte(MUNK-STANDER, SKOVGAARD e NIELSEN, 2005)**

Quando a aplicação rodando na camada de aplicação quer transmitir algo, ela repassará os dados para a subcamada APS. A subcamada APS formará o *frame* APDU e o repassará para a camada de baixo. A camada NWK formará o *frame* NPDU, juntando o *frame* APDU com o cabeçalho NWK, e o enviará para a camada de baixo. A camada MAC, por sua vez, montará o *frame* MPDU da junção do *frame* NPDU com o cabeçalho MAC. O *frame* MPDU é enviado para a camada PHY, que formará o *frame* PPDU e o transmitirá ao meio. O *frame* da camada PHY corresponde ao próprio pacote de dados.

### 3.2.2. Primitivas

Para transportar um frame de uma camada a outra, usa-se comandos chamados de Primitivas. As primitivas são os comandos de comunicação entre camadas adjacentes. É por meio de uma primitiva que uma camada de cima requer um serviço de uma camada de baixo, obtém informações de controle ou transfere *frames*. Todas as camadas têm um pequeno conjunto de primitivas para dialogar tanto com a camada abaixo como com a camada acima dela.

São quatro os tipos de primitivas: `.request`, `.confirm`, `.indication`, `.response`.

**.Request:** gerado quando uma camada deseja iniciar o serviço oferecido pela camada de baixo.

**.Confirm:** a camada de baixo responderá com uma confirmação um pedido `.request` feito, comunicando se foi possível executar ou não o serviço.

**.Indication:** é gerado quando a camada de baixo tem informações para a camada de cima, possivelmente como resultado da recepção de um *frame*, ou quando o coordenador recebe algum pedido em especial de um dispositivo.

**.Response:** esta primitiva é tomada de decisão da camada de cima ao envio de um comando `.indication` pela camada de baixo.

Ainda, as primitivas são classificadas em primitivas de dados ou primitivas de gerenciamento.

As primitivas de dados são aquelas encarregadas de encaminhar *frames* para a camada adjacente e de confirmar a recepção deste *frame*. Por exemplo, se a camada MAC precisa repassar seus *frames* para a PHY, ela enviará uma primitiva ***PD-DATA.request*** junto com o *frame*. E a camada PHY responderá com uma primitiva ***PD-DATA.confirm***, reportando se foi possível transmiti-los.

Já as primitivas de gerenciamento são comandos de controle e configuração. Por exemplo, através da primitiva ***PLME-CCA.request*** a camada MAC pede para camada PHY que seja verificado se um canal de comunicação está livre. A PHY retorna o pedido com uma primitiva ***PLME-CCA.confirm***, confirmando a disponibilidade do canal no momento.

A Tabela 1 mostra todas as primitivas que podem ser trocadas entre a camada física (PHY) e a camada de controle de acesso ao meio (MAC).

**Tabela 1 - Primitivas entre PHY e MAC**

Primitiva de Dados	.request	.confirm	.indication
PD-DATA	✓	✓	✓
<b>Primitiva de gerenciamento</b>			
PLME-CCA	✓	✓	
PLME-ED	✓	✓	
PLME-GET	✓	✓	
PLME-SET-TRX-STATE	✓	✓	
PLME-SET	✓	✓	

### 3.2.3. SAP (Service Access Points)

SAPs são conexões entre duas camadas e por onde as primitivas e *frames* são transferidos.

Define-se dois tipos de SAPs: um para transferência de dados e outro para serviços de gerenciamento. Entre duas camadas, sempre há esses dois tipos de SAPs, como ilustrado pela Figura 2.

**SAPs de transferência de dados:** são aqueles por onde se transportam as primitivas de dados e os *frames*. Por exemplo, as primitivas do tipo **PD-DATA** passam pela PD-SAP, localizado entre as camadas MAC e PHY.

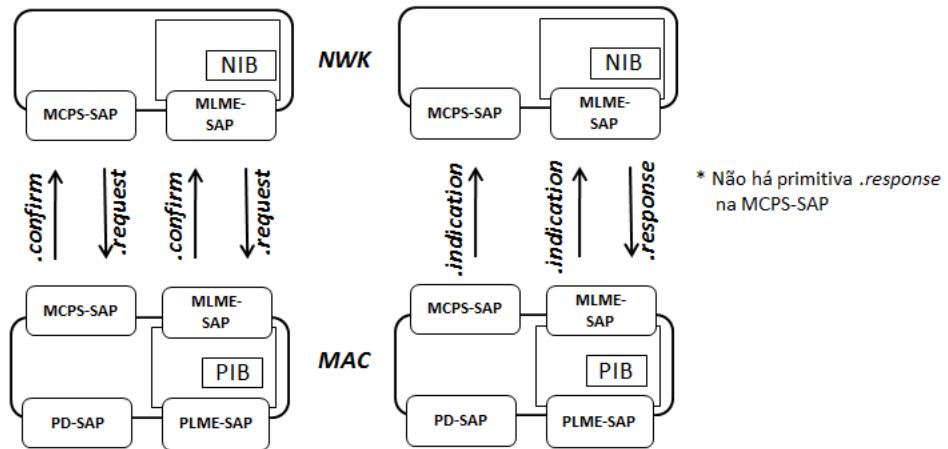
**SAP de gerenciamento:** são aqueles por onde passam as primitivas de gerenciamento. As primitivas do tipo **PLME-CCA**, por exemplo, passam pelo MLME-SAP, também localizado entre as camadas MAC e PHY.

No Zigbee, os SAPs de serviços de transferência de dados são:

- **APSDE-SAP** (entre os *Endpoints* e APS).
- **NLDE-SAP** (entre APS e NWK).
- **MCPS-SAP** (entre NWK e MAC).
- **PD-SAP** (entre MAC e PHY).

Os SAPs de gerenciamento são:

- **APSME-SAP** (entre ZDO e APS).
- **NLME-SAP** (entre APS e NWK; e entre ZDO e NWK).
- **MLME-SAP** (entre NWK e MAC).
- **PLME-SAP** (entre MAC e PHY).



**Figura 4 - exemplo de fluxo de primitivas entre as camadas MAC e NWK**

### 3.2.4. Information Base

A *Information Base* é uma Base de dados onde são mantidos os atributos e as constantes de cada camada. Cada camada Zigbee apresenta uma *Information Base*. Nas camadas PHY e na MAC ela é chamada de PIB (*PAN Information Base*). Na camada NWK é chamada de NIB (*Network Information Base*) e na camada APS ela é nomeada de AIB (*APS Information Base*).

Por meio de primitivas de gerenciamento de leitura e escrita, a camada de cima pode verificar ou modificar o valor de um atributo dentro da *Information Base* da camada de baixo.

**Atributos:** São parâmetros de configuração da rede ou do dispositivo que podem ser alterados/modificados.

**Constantes:** definem características como duração de um evento ou tamanho máximo/mínimo de *frames*, *payload* ou cabeçalho. Os valores das constantes são fixos e não podem ser mudados, como nos atributos. Por exemplo, a PIB da camada PHY contém a constante *aMaxPHYPacketSize* = 127 bytes que define o tamanho máximo permitido para um pacote de dados.

Cada camada tem seus próprios atributos e constantes, que caracterizam as funções de camada.

### 3.3. Definições IEEE 802.15.4

#### 3.3.1. Frequências de Rádio

Para estar de acordo com a regulamentação de redes sem fio na Europa, Japão, Canadá e EUA, a especificação IEEE 802.15.4/2003 permite o uso de três espectros de frequência e dois tipos de modulação, mostrados abaixo.

- Espectro de 868 – 868.6 MHz: modulação BPSK (*Binary Phase - Shift Keying*)  
taxa máxima de 20 kbps
- Espectro de 902 – 928 MHz: modulação BPSK  
taxa máxima de 40 kbps
- Espectro de 2400 – 2483.5 MHz: modulação O-QPSK (*Offset - Quadrature Phase - Shift Keying*)  
taxa máxima de 250 kbps

Cada um dos três espectros de frequência é dividido em faixas menores para formar diversos canais de comunicação. Somando os canais dos três espectros, o Zigbee possibilita 27 canais de comunicação enumerados de  $k = 0$  a 26. No espectro de 868 MHz há apenas 1 canal de transmissão/recepção, em 915 MHz há 10 canais e em 2,4 GHz existe 16 canais à disposição. A frequência central de cada canal é calculada, em função de  $k$ , pelas seguintes fórmulas.

$$f_{central} = 868.3 \text{ MHz}, \quad k = 0;$$

$$f_{central} = 906 + 2 \cdot (k - 1) \text{ MHz}, \quad 1 \leq k \leq 10;$$

$$f_{central} = 2405 + 5.(k - 11)MHz, \quad 11 \leq k \leq 26;$$

O número e a largura de banda de cada canal crescem com a frequência. A largura dos canais é de 0,6 MHz em 868 MHz; 2 MHz em 915 MHz; e 5 MHz em 2,4 GHz.

### 3.3.2. Pacotes de Dados PHY

O pacote de dados da camada PHY é chamado de **PHY Protocol Data Unit (PPDU)**. Após o cabeçalho PHY ser adicionado ao *frame* MAC, o pacote será transmitido ao meio.

O PPDU pode ser fragmentado em três partes: o cabeçalho de sincronização (SHR), cabeçalho de informação do tamanho do *frame* (PHR) e a parte de dados que contém o *frame* vindo da MAC (PSDU).

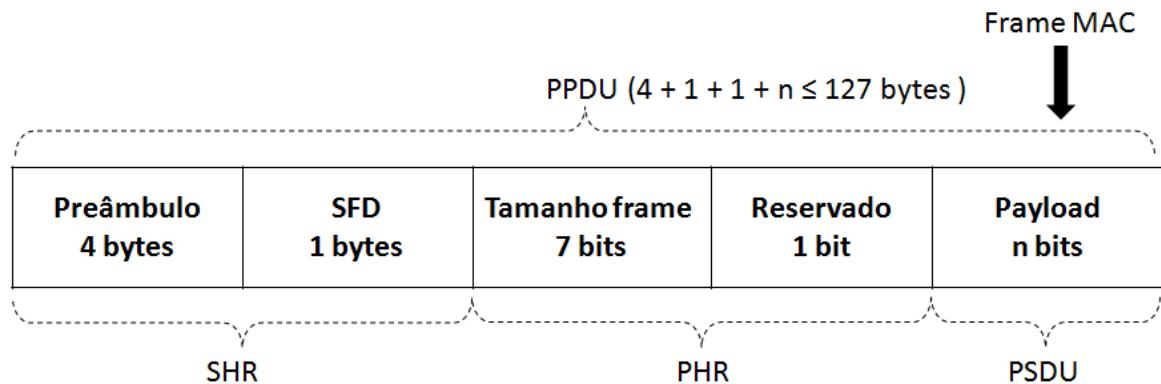


Figura 5 - pacote de dados da camada Física

O cabeçalho **SHR** é formado pelos campos **preâmbulo** e pelo **SFD** (*Start of Frame Delimiter*). O preâmbulo é formado por uma sequência de bits usada para sincronizar o dispositivo receptor com o pacote de dados recebido, e o campo SFD só delimita o fim dos bytes de sincronização e o início dos campos de dados.

A frequência do oscilador é, quase sempre, diferente de dispositivo para dispositivo. Esta diferença, apesar de pequena, causa um descompasso na temporização de eventos. Um descompasso entre transmissor e receptor pode prejudicar a recepção de um pacote de dados, mesmo que a frequência de oscilação entre eles seja ligeiramente diferente. Caso este problema ocorra, ele será corrigido por meio da sequência de bits do preâmbulo, que o primeiro campo a ser enviado pelo transmissor e contém 32 bits nulos. Já o campo

SFD deve ser setado com a sequência de bits 11100101, onde nesta sequência o bit LSB está localizado o mais à esquerda.

O campo **PHR** descreve o tamanho do campo *payload* ou PSDU, que varia de 0 a 127 bytes.

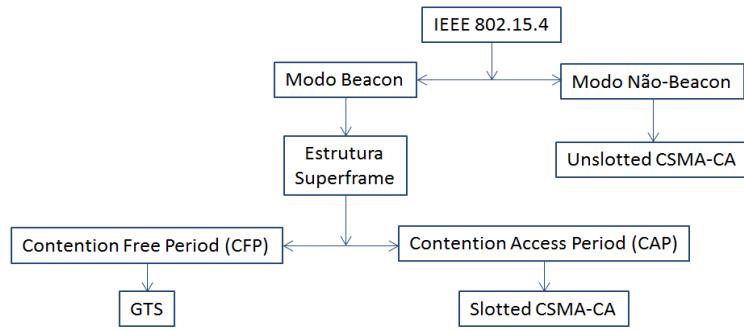
O **payload** é o campo que contém o *frame* MAC. Para um comprimento de 5 bytes, o PSDU será formado por um *frame* MAC de reconhecimento. Para um comprimento maior que 8 bytes, ele será formato por outros tipos de *frame* MAC (os frames MAC são mostrados no Apêndice [C] Camada MAC, NWK e APL: frame e comandos).

### 3.3.3. Modo Beacon e não-Beacon

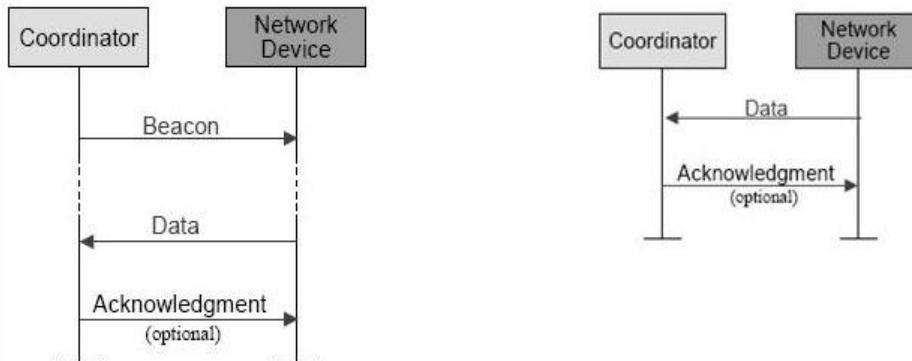
Uma rede Zigbee, funcionando no modo *beacon*, utiliza a estrutura de superquadro para o acesso dos dispositivos ao canal de transmissão. Nesta estrutura, o acesso é limitado dentro de períodos de tempo pré-determinados, parecido como o que ocorre em uma multiplexação por divisão do tempo. Um dispositivo desejando transmitir terá a oportunidade de fazê-lo somente dentro do período ativo do superquadro, que ocorre após a sinalização do coordenador.

O coordenador transmite um sinal *beacon* aos outros dispositivos, marcando o início do período de troca de dados dentro da rede. Ao receberem o *beacon*, os dispositivos estão livres para transmitir os seus dados, caso o tenham, durante o período ativo do superquadro. Os dispositivos sabem o intervalo de tempo que tem para troca de dados, pois após este intervalo eles entrarão em modo de suspensão para preservarem o máximo de energia possível.

Ao contrário, na rede *não-beacon* não existe o superquadro ou transmissão de um sinal *beacon* pelo coordenador, assim não há, também, um período específico para se transmitir. Qualquer dispositivo desejando transmitir o poderá fazer a qualquer momento, bastando que o canal esteja livre.



**Figura 6 - tipos de mecanismos de acesso ao canal - modo beacon e não beacon – retirada da fonte (NIEUWENHUYSE, ALVES e KOUBÂA, 2006)**



**Figura 7 - Transferência de dados entre coordenador e um dispositivo no modo beacon e no modo não-beacon – retirada da fonte (IEEE, 2003)**

### 3.3.4. Estrutura superquadro

O **superquadro** ou *superframe* é dividido em duas partes: ativa e inativa.

A parte ativa compreende o período do superquadro liberado para transmissões. O período ativo do superquadro começa a partir da emissão, pelo coordenador, de um *frame beacon* difundido na rede periodicamente pelo coordenador.

A parte inativa inicia-se após o término do período ativo, e é o momento em que a rede fica “silenciosa”, sem transações, e a maioria dos dispositivos aproveitam para entrar em modo de suspensão, reduzindo consideravelmente o consumo de energia.

O período de tempo,  $t$ , das partes ativa e inativa variam. Em redes com grande volume de tráfego a parte inativa do superquadro deve ser reduzida e nas redes com pouco

tráfego, em que há uma margem para a suspensão dos dispositivos, a parte inativa pode aumentar. O período total do superquadro e da parte ativa são fixados pelos atributos *macBeaconOrder* (BO) e *macSuperframeOrder* (SO), respectivamente. O período inativo corresponde ao período do superquadro, menos o período ativo.

Os períodos abaixo estão na unidade símbolos.

$$t_{superframe} = aBaseSuperframeDuration * 2^{BO} \quad 0 \leq BO \leq 14;$$

$$t_{ativa} = aBaseSuperframeDuration * 2^{SO} \quad 0 \leq SO \leq BO \leq 14;$$

$$t_{slot} = aBaseSlotDuration * 2^{SO}$$

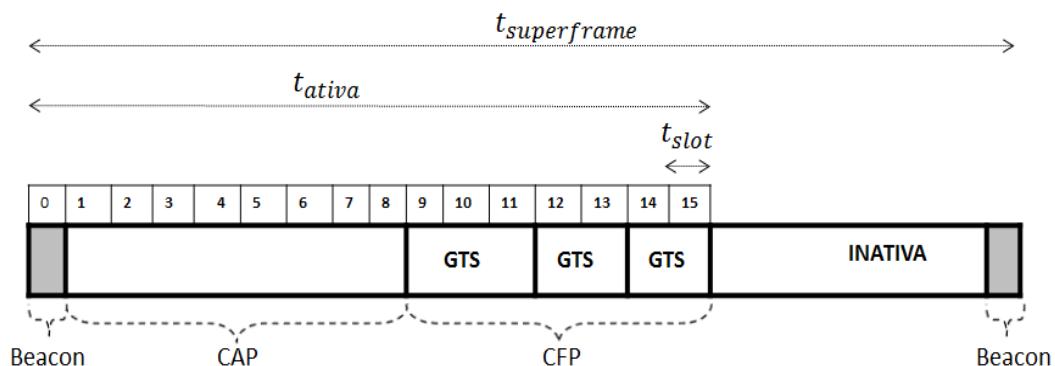
$$t_{inativa} = aBaseSuperframeDuration * (2^{BO} - 2^{SO})$$

O valor, em segundos, dos períodos acima para BO = 14, SO = 12 e 1 símbolo = 16µs (para rádio de 2,4 GHz) é:

$$t_{superframe} = 251,65s \quad t_{ativa} = 62,91s$$

$$t_{slot} = 3,93s \quad t_{inativa} = 188,74s$$

Para a rede no modo *não-beacon* BO e SO deverão ser iguais a 15. Sendo assim, o coordenador não transmitirá sinalizadores *beacons* e o acesso ao meio será feito através do *unslotted CSMA – CA* (explicado na subseção 3.3.5).



**Figura 8 - Estrutura superquadro**

Toda parte ativa é dividida em 16*slots* de iguais períodos,  $t_{slot}$ , que também variam de acordo com o valor de SO. O primeiro *slot* é reservado para o sinal *beacon* e os restantes são distribuídos entre o período de disputa pelo acesso (CAP - Contention Access Period) e o período livre de disputa (CFP- Contention Free Period).

**Sinal *beacon*:** é considerado pertencente à parte ativa do superquadro. Dentre outras coisas, o *frame beacon* contém informações de sincronização, de identificação da rede WPAN e de descrição da estrutura do superquadro.

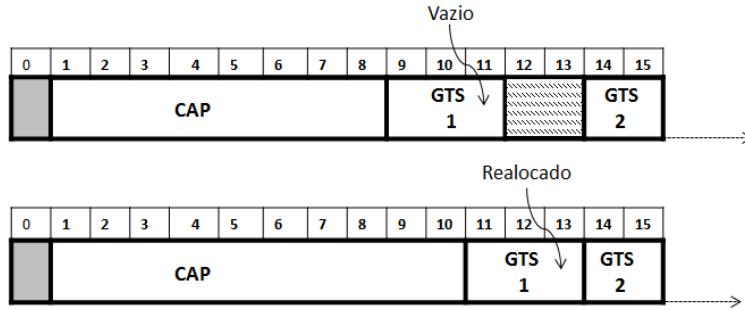
**CAP:** No período CAP, o canal não é reservado por um intervalo de tempo para um único dispositivo. Para acessar o canal será preciso, antes de tudo, verificar se ele está livre por meio do mecanismo *slotted CSMA-CA*. Um dispositivo transmitindo durante o período de disputa de acesso ao canal deverá certificar-se de que a transferência de dados seja finalizada antes do fim do período CAP.

**CFP:** No período CFP não há disputa pelo acesso do canal de comunicação, uma vez que são reservados períodos GTSs (*Guaranteed Time Slots*). Durante um GTS o canal será alocado exclusivamente para um dispositivo e, portanto, estes não precisarão disputar o acesso ao canal. Vale ressaltar que dispositivos que alocaram GTS também estão liberados para transmitir no período CAP.

Para gerenciar a alocação de espaços dentro do CFP, os coordenadores detêm informações como: slot de início do GTS, comprimentos do GTS, endereço dos dispositivos que alocam espaço, entre outros dados.

Se após alguns períodos de superquadro um dispositivo que aloca uma GTS não usá-lo para transmissão, terá seu espaço GTS liberado para outros dispositivos. Se após  $p$ superquadros não houver transmissões dentro do GTS, o coordenador retirará o GTS alocado para o dispositivo, liberando mais espaço para CAP, como mostrado na Figura 9.

$$p = \begin{cases} 2 * 2^{(8-macBeaconOrder)} & 0 \leq macBeaconOrder \leq 8; \\ 2 & 8 < macBeaconOrder; \end{cases}$$



**Figura 9 - Gerenciamento de espaços GTS**

### 3.3.5. CSMA – CA

O termo CSMA - CA significa *Carrier Sense Multiple Access with Collision Avoidance*. É um método usado para prevenir o uso simultâneo do canal por mais de um dispositivo, em um mesmo momento. O método é diferente do CSMA – CD (*Collision Detection*), que funciona quando a colisão já aconteceu.

No método CSMA-CA, será esperado um período aleatório antes de checar se o canal está livre. Somente após o término deste período aleatório, o dispositivo verificará se o canal está desocupado e, caso esteja, transmitirá. Entretanto, se o canal estiver ocupado, o dispositivo terá que aguardar novamente por um novo período aleatório e, só então, fazer a checagem de canal, novamente. O procedimento se repetirá até o dispositivo querendo transmitir conseguir encontrar o canal livre.

O período aleatório,  $t_{delay}$ , a se esperar é quantizado em unidades de  $aUnitBackoffPeriod$  igual a 20 símbolos e pode variar de:

$$0 \leq t_{delay} \leq aUnitBackoffPeriod * (2^{BE} - 1)$$

$$t_{delay} \leq 620 \text{ símbolos}$$

Para BE = 5e1 símbolo = 16 μs (em 2,4 GHz)

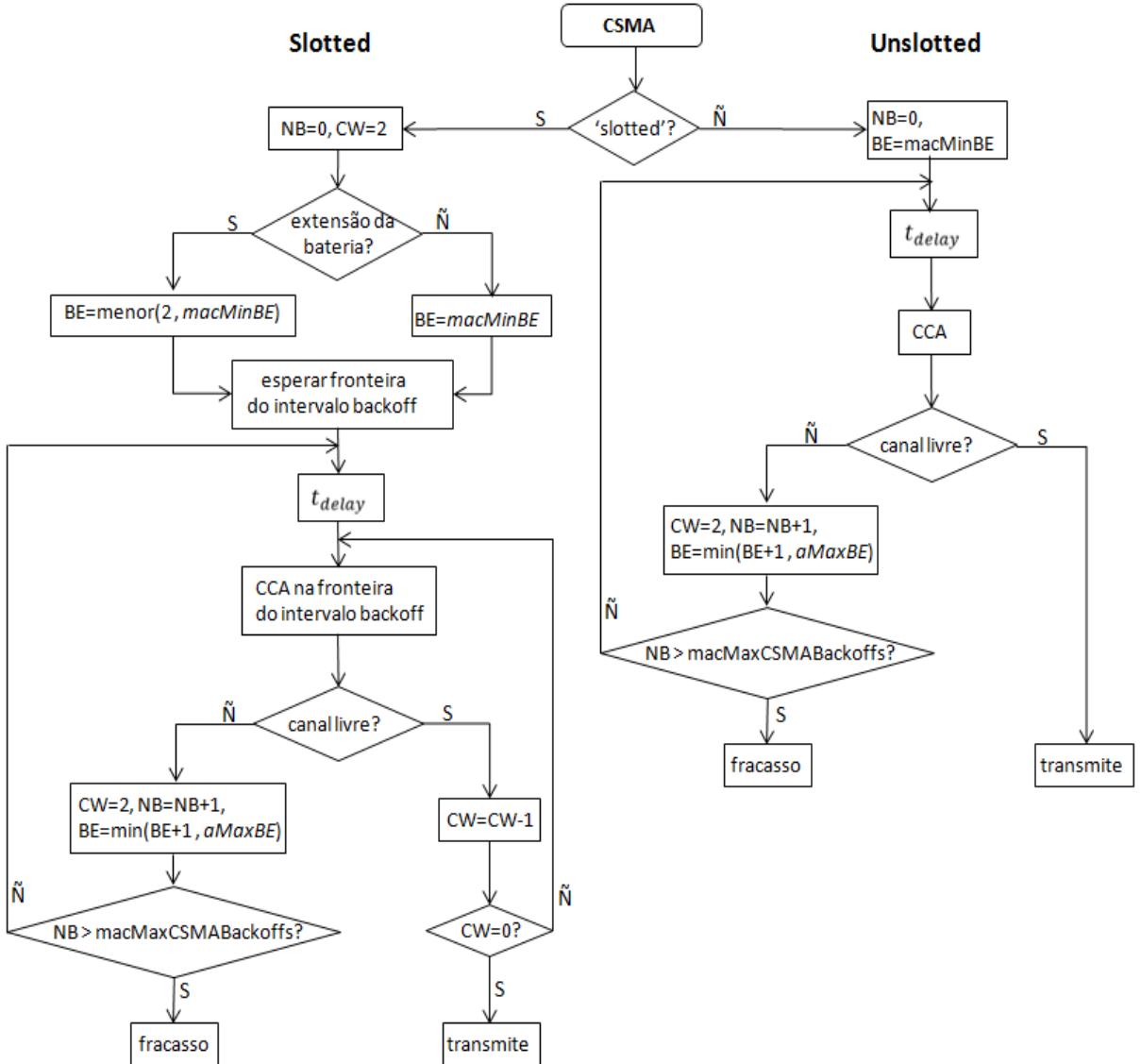
$$t_{delay} \leq 9,92 \text{ ms}$$

O valor mínimo de  $BE$  pode variar entre 0 a 3 e o valor máximo é 5. O mínimo é definido no atributo `macMinBE` e o máximo, na constante `aMaxBE`. O mecanismo de avaliação do canal será feito com o dispositivo no modo de recepção.

Ambos os modos, *beacon* e *não-beacon*, lançam mão do método CSMA para verificar se o canal está ocioso. Entretanto, cada modo utiliza algoritmos CSMA ligeiramente diferentes: *slotted* e *unslotted*, como mostrado no fluxograma da Figura 10

**Unslotted CSMA – CA:** usado para acessar o canal em redes no modo *não-beacon*, onde não há estrutura superquadro nem sinais *beacons* enviados pelo coordenador. Os dispositivos poderão tentar ocupar o canal de comunicação a qualquer momento, implementando o algoritmo *unslotted CSMA-CA*.

**Slotted CSMA – CA:** mecanismo empregado por dispositivos tentando acessar o canal no período CAP do superquadro, quando há disputa pelo acesso ao canal.



**Figura 10- Algoritmo CSMA-CA unslotted e slotted - retirada da fonte (IEEE, 2003)**

No método *slotted*, é preciso fazer uma checagem dupla da disponibilidade do canal, antes de se transmitir. O parâmetro CW é usado para fazer esta contagem. CW é iniciado em 2 e decrementado cada vez que o canal está livre. Quando CW torna-se 0, o dispositivo poderá transmitir. Caso CW já esteja em 1, e na segunda tentativa detecta-se que o canal está ocupado, o valor de CW será resetado, retornando ao valor 2. Já o parâmetro NB faz a contagem das tentativas de acesso ao canal e o número máximo permitido, *macMaxCSMABackoff*, é de 5 tentativas.

Ao contrário do *slotted*, o algoritmo *unslotted* poderá iniciar a qualquer momento sem precisar estar sincronizado ao intervalo *backoff*. Também não é necessário executar o mecanismo CCA (*Clear Channel Assessment*) pelo menos duas vezes antes de transmitir,

como ocorre no *slotted*. Se na primeira avaliação o canal estiver disponível, o dispositivo o ocupará.

### 3.3.6. Varredura de Canal

As varreduras ou escaneamento são artifícios que os dispositivos têm para verificar o ambiente ao seu redor. Elas são usadas por coordenadores para selecionar o canal mais adequado para iniciar uma nova rede PAN ou para escolher um identificador PAN único para a nova rede. Roteadores e end-devices as executam quando querem se associar a uma rede ou reencontrar o coordenador de sua rede PAN, em caso de falha de sincronização com ele.

As varreduras se iniciam com uma primitiva de pedido **MLME-SCAN.request**, da camada NWK para a MAC.

```
Primitiva MLME-SCAN.request (
    scanType,
    scanChannels,
    scanDuration
)
```

O primeiro parâmetro, *scanType*, é um parâmetro de dois bits com o tipo de varredura: detecção de energia – 00, ativa – 01, passiva – 10, órfão – 11. O segundo parâmetro, *scanChannels*, lista todos os canais que serão varridos. O terceiro parâmetro, *scanDuration*, determina a duração da varredura em cada canal.

Conforme o parâmetro *scanType*, há quatro opções de varreduras de canais.

A **varredura passiva** mapeará os coordenadores que transmitem *beacon* ao redor, permitindo que um dispositivo descubra os coordenadores e roteadores perto dele – dentro do seu raio de transmissão.

A **varredura de órfão** é utilizada por um dispositivo para tentar reencontrar seu coordenador após perda de sincronização com ele. O comando MAC de notificação de órfão faz parte do processo de varredura de órfão (ver comandos MAC no Apêndice [C] Camada

MAC, NWK e APL: frame e comandos). O coordenador deverá enviar ao órfão um comando MAC de realinhamento do coordenador.

A **varredura ativa** também é usada para mapear coordenadores e roteadores ao redor. Ao contrário da varredura passiva, a MAC lança mão de um *frame* com comando MAC de pedido de *beacon*. Todos os coordenadores que receberem o pedido deverão enviar um *frame beacon* de volta, com informações da rede que ele coordena.

A **varredura de detecção de energia (ED)** mede a energia dos canais disponíveis. Com a medição obtém-se informações sobre o uso de cada canal, que permitem ao coordenador escolher o canal mais adequado para operar. Durante este varredura será descartado qualquer tipo de *frame* que chegue a MAC.

Se possível, a energia do canal deverá ser medida várias vezes durante o intervalo da varredura. Para isso, a MAC deverá fazer várias requisições de detecção de energia para a PHY com a primitiva ***PLME-ED.request***.

A varredura de cada canal durará, em símbolos, por  $aBaseSuperframeDuration * (2^{ScanDuration} + 1)$ . A Tabela 2 abaixo mostra a duração, em segundos, da varredura para diferentes valores de *ScanDuration* e para o transceptor de rádio operando no espectro de 2,4 GHz ( $1\text{símbolo} = 16\mu\text{s}$ ).

**Tabela 2 - ScanDuration para 2,4 GHz**

<i>ScanDuration</i>	Duração (ms)	<i>ScanDuration</i>	Duração (ms)
0	31	8	3.948
1	46	9	7.880
2	77	10	15.744
3	138	11	31.473
4	261	12	62.930
5	507	13	125.844
6	998	14	251.674
7	1.981		

## 4. CAMADA NWK E REDE ZIGBEE

A camada de rede está entre a subcamada de suporte a aplicação (APS) e a camada MAC, e tem o papel de vincular corretamente uma aplicação aos níveis IEEE 802.15.4. As principais funções da camada NWK são:

**Começar uma nova rede:** habilidade de um coordenador de formar uma nova rede.

**Configuração da rede:** a camada NWK de um coordenador tem a permissão de reconfigurar a rede alterando, por exemplo, a profundidade da rede, número de nós “filhos” por roteador “pai”, se necessário.

**Juntar-se ou sair de uma rede:** a habilidade de um dispositivo decidir deixar ou entrar em uma rede, ou a de um coordenador ou roteador de pedir a saída de algum elemento da rede.

**Descoberta de vizinhos:** a habilidade de descobrir informações sobre nós vizinhos e armazenar estas informações na memória não volátil.

**Endereçamento:** a habilidade de um roteador ou coordenador em atribuir um endereço de rede para um novo dispositivo que acabou de se juntar na rede.

**Roteamento:** a habilidade de descobrir e armazenar diferentes rotas para a entrega de uma mensagem até o destinatário final. A camada de rede exibe mecanismos inteligentes de roteamento em malha, estendendo o alcance da rede.

**Transmissão multicast:** a camada NWK possibilita que uma mensagem seja enviada a apenas um grupo de nós dentro da rede. Isto é uma funcionalidade adicionada pelo Zigbee, já que no nível IEEE 802.15.4 só é possível à transmissão para um único dispositivo ou para todos (*broadcast*).

**Segurança:** a capacidade de adicionar segurança ao payload do *frame* NWK encriptando a mensagem.

## 4.1. Estrutura de Rede

### 4.1.1. Dispositivos Zigbee

Classificam-se dois tipos de dispositivos Zigbee, de acordo com a capacidade de processamento: FFD e RFD.

**FFD – Full Function Device:** Têm maior capacidade de processamento que o RFD. Ele consegue se comunicar com outros FFDs ou com RFDs e é o único que pode agir como um coordenador na rede Zigbee. Pode tanto funcionar como um coordenador, roteador ou *end-device*.

**RFD – Reduced Function Device:** No RFD os recursos e a memória são mais limitados e não apresenta toda a funcionalidade de um FFD. Por isso, ele é dedicado para aplicações simples como sensoriamento ou ligar/desligar lâmpadas remotamente. O RFD não tem a capacidade de se tornar um coordenador ou roteador e nem é capaz de se comunicar com outros RFD, por isso só se comunica diretamente com um FFD. Só pode se tornar um *end-device* em uma rede Zigbee

Quanto à função que desempenham dentro da rede, os dispositivos Zigbee podem ser classificados de três modos: coordenador, roteador ou *end-device*.

**Coordenador Zigbee:** gerencia a rede, sendo responsável pela maioria das atividades na rede. Ele seleciona o canal a ser usado, começa a rede, permite que novos dispositivos se associem a ele e também faz a transferência de mensagens como um roteador. Em redes no *modo-beacon*, também estabelece o momento em que os dispositivos podem transmitir.

As redes Zigbee têm, mandatoriamente, um coordenador, independente da topologia adotada e suas características são:

- Inicia uma rede.
- Retransmite mensagens de um nó a outro.
- Permite que outros nós se juntem a rede.
- Estabelece o período permitido para transmissão, no modo *beacon*

- Normalmente, não entra em modo de suspensão.
- Gerencia a rede.
- Armazena informações sobre os outros nós da rede
- Requer um espaço extra de memória.

**Roteador Zigbee:** o roteador tem a capacidade de repassar as mensagens de um dispositivo para o outro. Também tem a capacidade de associar outros dispositivos a ele, integrando-os a rede. Porém, não inicia uma nova rede. Ele é útil quando se deseja estender o tamanho da rede, criando ramificações e, também, pode funcionar como um *end-device*.

- Retransmite mensagens de um nó a outro.
- Junta-se a uma rede já existente, através de um coordenador ou um outro roteador.
- Permite que outros nós se juntem a rede.
- Pode, também, funcionar como um *end-device* se for necessário.

**End-Device Zigbee:** o *end-device* apenas transmite/recebe mensagens e não as reencaminha para outros dispositivos, como fazem os roteadores. Também não conecta novos dispositivos à rede. Geralmente é alimentado à bateria e costuma entrar em modo de suspensão, para conservar o máximo de energia.

- Junta-se a uma rede já existente.
- Normalmente é operado à bateria.
- Normalmente entra em modo de suspensão.
- Tem capacidade de processamento limitada.

#### 4.1.2. Topologias de rede

A topologia é a maneira em que os dispositivos se agrupam em uma rede para se comunicarem. As topologias possíveis no Zigbee são: em estrela, em árvore e em malha.

**Topologia Estrela:** nesta topologia um nó central (no Zigbee será o coordenador) é conectado com todos os outros nós da rede. Uma mensagem só chegará a um destinatário mediante passagem pelo coordenador. O esquema em estrela é mais simples, porém pouco dinâmico, no sentido em que todos os dados fluem para o nó central, muitas ve-

zes sobrecarregando-o. Uma falha no nó central fará todas as comunicações da rede serem cortadas, já que não há de rotas alternativas que não passem, necessariamente, pelo coordenador.

Na topologia estrela, os dispositivos podem funcionar somente com a pilha IEEE 802.15.4, já que mecanismos de roteamento inteligentes só são implementados na camada de rede.

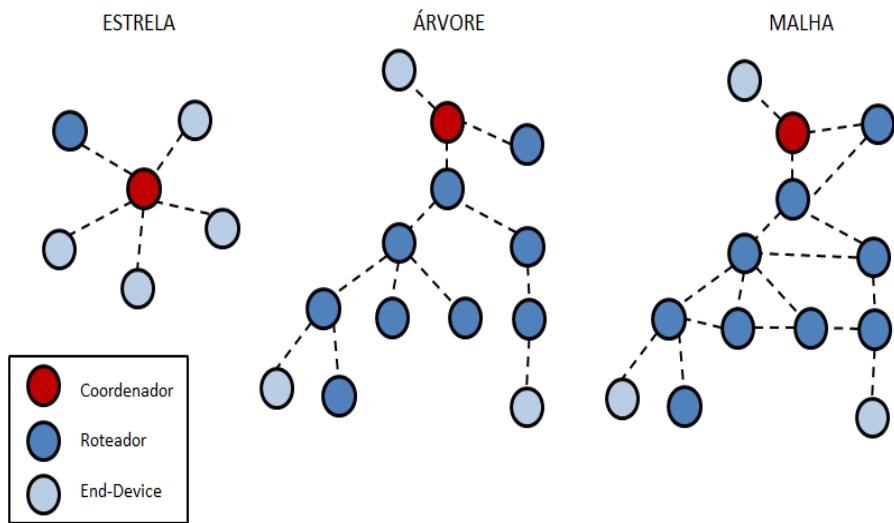
**Topologia em Árvore:** na topologia em árvore, o coordenador é conectado a vários outros roteadores e *end-devices* em uma topologia, inicialmente, em estrela. Mas estes roteadores também acrescentarão outros *end-devices* ou outros roteadores à rede, gerando diferentes níveis hierárquicos.

Usa-se as terminologias “pai” e “filho” para definir a relação entre dispositivos na estrutura em árvore. Um nó “pai” (coordenador ou roteadores) conecta vários nós “filhos” (outros coordenadores ou *end-devices*) a rede. Cada nó filho só pode se comunicar diretamente com o seu próprio pai. E o pai só pode se comunicar diretamente com os seus filhos ou com o seu pai em um nível hierárquico acima.

O esquema em árvore não é totalmente auto recuperável, no sentido que ainda só há uma rota entre quaisquer pares de dispositivos. Se um dispositivo falhar, algumas rotas serão interrompidas, mas nem todas elas, como ocorre na topologia em estrela.

**Topologia em Malha:** é estruturalmente similar à topologia em árvore, porém os roteadores podem estabelecer comunicações entre si. Assim, criam-se caminhos redundantes entre dois nós e a estrutura torna-se resiliente com rotas alternativas em caso de falha de algum dispositivo. O roteamento, porém, torna-se mais complexo.

A Descoberta de rota é uma das características da rede em malha, na qual é verificada a melhor rota entre dois dispositivos distantes.



**Figura 11 - Topologias de Rede**

#### 4.1.3. Endereçamento Zigbee

No Zigbee, tanto os dispositivos quanto a rede apresentam endereços que os identificam em um local.

**PAN ID:** Para cada rede PAN (*Personal Area Network*) atribui-se um número único de 16 bits chamado PAN ID. Esta identificação torna possível que várias redes coexistam em um mesmo local, sobre o raio de influência uma das outras, sem que haja problema nisto. O valor no identificador PAN ID varia entre 0x0000 a 0x3fff. Isto possibilita 1.6384 identificadores distintos, que corresponde ao número de redes que podem estar sobre o mesmo alcance. Duas redes podem ter o mesmo valor de PAN ID, desde que elas estejam utilizando canais distintos ou estejam fora do raio de alcance uma da outra.

- O valor do identificador PAN ID é armazenado no atributo *macPANID* na PIB.

**PAN ID Estendido:** é um número de 64 bits que identifica unicamente uma PAN globalmente e, por isso, um valor não pode ser repetido por nenhuma outra rede. Esse identificador é usado somente para ajudar um dispositivo a encontrar a rede certa a se juntar.

- O valor do identificador PAN ID estendido é armazenado no atributo *nwkExtendedPA-  
NId* na NIB.

**Endereço 64 bits - endereço MAC:** é o endereço exclusivo de cada dispositivo Zigbee no mundo. Também é chamado de endereço longo. Um conjunto de 64 bits possibilita 4 bilhões de equipamentos Zigbee por metro quadrado no planeta. Para cada dispositivo, este endereço será sempre o mesmo, independente da rede que o dispositivo se encontre.

- Este endereço é armazenado na constante *aExtendedAddress* na PIB.

**Endereço 16 bits - endereço NWK:** é o endereço exclusivo de cada dispositivo dentro da rede. Quando um dispositivo entra em uma rede, o seu endereço será escolhido pelo coordenador da rede ou pelo roteador que permitiu a associação deste novo dispositivo. Assim, se o dispositivo sair da rede ele deixará de ter um endereço NWK, e se ele se associar novamente a uma nova rede passará a ter um novo endereço NWK. Dois dispositivos Zigbee dentro de uma mesma rede não poderão ter o mesmo endereço de 16 bits. Os endereços atribuíveis variam de 0x0000 a 0xffff7.

Os coordenadores, por padrão, terão o endereço NWK nulo, 0x0000. Vale ressaltar que, dois coordenadores, pertencentes a diferentes PAN ID, poderão ter o mesmo endereço NWK.

- Este endereço de 16 bits é armazenado no atributo *nwkNetworkAddress* na NIB.

## 4.2. Atribuição de Endereços

O Zigbee usa dois esquemas para atribuir endereços NWK para os dispositivos na rede: Estocástico (aleatório) e Cskip.

### 4.2.1. Estocástico

No modo estocástico, um dispositivo que acabou de entrar na rede escolhe um endereço NWK aleatoriamente. Após a escolha, ele envia uma mensagem difundida para toda a rede para ter certeza que ninguém o está utilizando ainda. Se o endereço já estiver sendo usado, o dispositivo escolherá um outro endereço, até encontrar algum que esteja

disponível - o que não é muito difícil, pois há uma grande quantidade de endereços disponíveis para escolha. Este método é disponibilizado somente no *stack profile* 0x02<sup>1</sup> do Zigbee.

#### 4.2.2. Cskip

O Cskip é um outro método de atribuição de endereços para dispositivos na rede. Este método usa uma função com o mesmo nome para fazer a distribuição dos endereços de forma organizada. O valor da função Cskip é determinado por três atributos na base de dados NIB (*Network Information Base*):

*nwkMaxDepth(Lm)*: é a profundidade máxima permitida para um dispositivo da rede.

*nwkMaxChildren(Cm)*: é o máximo número de “filhos” que um roteador pode associar. *End-devices* terão sempre  $Cm = 0$ .

*nwkMaxRouters(Rm)*: é o máximo número de roteadores “filhos” que podem se associar a um roteador (um coordenador também entra como um roteador).

Esta função leva em conta mais uma variável, que é a profundidade, *depth*, de cada dispositivo da rede. Na rede Zigbee, o coordenador, localizado na raiz da rede, terá profundidade igual a 0.

A função *Cskip* é calculada para cada roteador e simboliza a distância do endereço de um dispositivo filho para outro. Por exemplo, se um roteador-pai tem um *Cskip* = 5 e endereço 0, os endereços de seus filhos poderão ser [1, 6, 11, 16 ...].

$$Cskip = \begin{cases} 1 + Cm * (Lm - d - 1), & \text{se } Rm = 1 \\ \frac{1 + Cm - Rm - Cm * Rm^{Lm - d - 1}}{1 - Rm}, & \text{caso contrário} \end{cases}$$

- Só se atribui valores *Cskip* para dispositivos do tipo coordenador ou roteador
- A ordem que os filhos se associam a um pai, influencia no valor de seus endereços de rede. Ou seja, o endereço de rede do primeiro a se associar será o endereço do seu pai

---

<sup>1</sup> Stack Profile 0x02 (Zigbee Pro Stack) e Stack Profile 0x01 (Zigbee 2006/2007)

somado a 1, e o endereço do segundo filho a se associar será o endereço do primeiro a se associar mais o valor de *Cskip* do pai.

- Roteadores com *Cskip* igual a zero não aceitarão que dispositivos se associem a ele, e se tornem seus filhos.
- Se um pai atingir o limite de endereços que ele pode distribuir a seus filhos, ele não poderá mais aceitar a associação de novos dispositivos.

O endereço, *Add*, do *n*-ésimo roteador a se associar a um mesmo pai será:

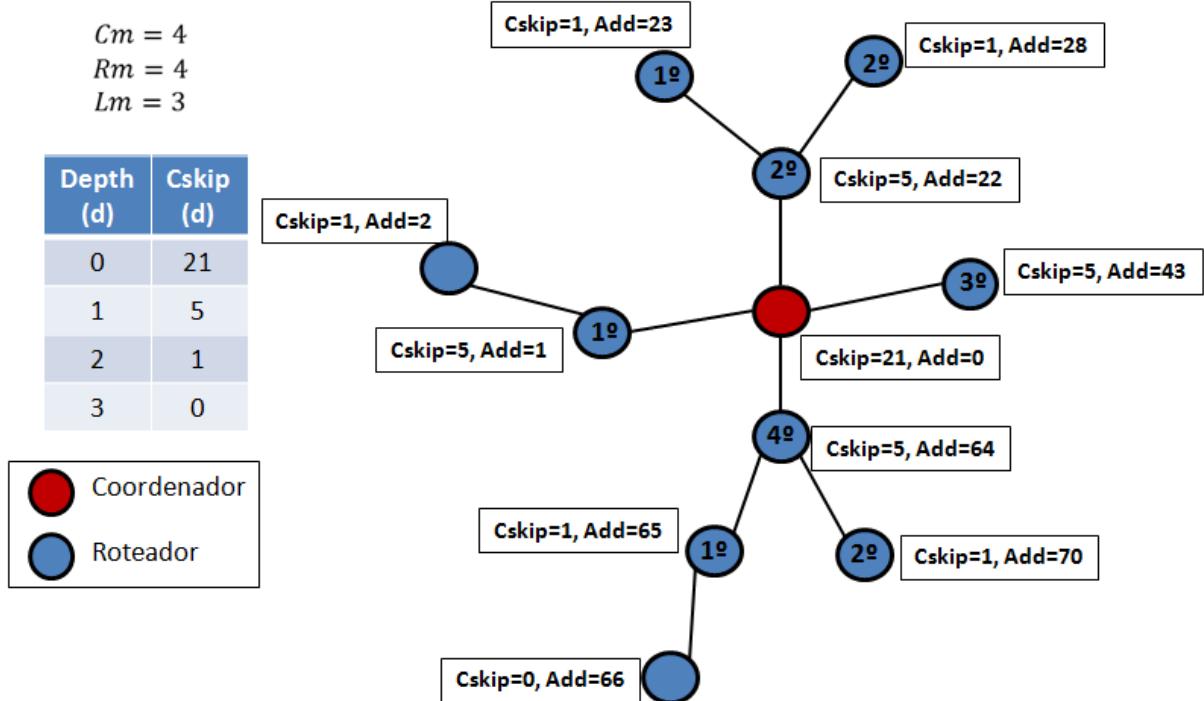
$$Add_n = Add_{n-1} + Cskip_{pai}$$

Onde *Add<sub>n-1</sub>* é o endereço do (*n* - 1)-ésimo roteador a se associar ao pai.

E o endereço do *n*-ésimo *end-device* a se associar a rede será:

$$Add_n = Add_{pai} + Cskip_{pai} * Rm + n$$

A Figura 12 mostra um exemplo de distribuição de endereços pelo método *Cskip*.



**Figura 12 - Exemplo cálculo da função Cskip - exemplo retirado da fonte (NIEUWENHUYSE, ALVES e KOUBÂA, 2006)**

Observa-se na Figura 12, que os dois filhos do roteador com  $Add = 22$  só podem ter endereços entre 22 e 43, ou seja, dentro de um intervalo  $Cskip = 21$  unidades de seu pai.

### 4.3. Dinâmica de uma rede Zigbee

#### 4.3.1. Formação de uma rede

Somente o coordenador pode formar uma nova rede e para formá-lo ele terá que encontrar um canal apropriado para operar e um identificador PANID para a sua nova rede. Inicialmente, o coordenador faz uma varredura de detecção de energia por vários canais. Todos os canais varridos que apresentarem um nível de energia acima de um patamar pré-determinado serão descartados, selecionando como opção de uso somente os canais com nível de energia abaixo do patamar - o Zigbee não especifica o valor deste patamar, ficando a cargo do desenvolvedor estabelecer o valor. Depois, o coordenador faz uma varredura ativa, e por meio dela é retornada uma lista com todos os PAN IDs vizinhos.

Dentre os canais pré-selecionados pela detecção de energia, o escolhido será aquele que apresentar o menor número de redes Zigbee operando sobre ele. Se em um canal não houver nenhuma rede Zigbee operando, será este o canal escolhido pelo coordenador. Decidido o canal, o coordenador escolherá uma PAN ID única e para isso levará em contas as outras PAN IDs já sendo usadas por outras redes naquele canal. Com a seleção do canal e PAN ID, o coordenador estará pronto para receber outros dispositivos na rede (roteadores e *end-devices*).

O momento em que o coordenador iniciará a formação de uma nova rede dependerá da aplicação rodando nele. Uma aplicação pode exigir que assim que o coordenador seja ligado pela primeira vez ele já estabeleça uma nova rede. Outra aplicação pode desejar esperar por um evento (aperto de um botão, etc.) ou esperar por um período determinado para formar a rede. Há diversas possibilidades.

#### 4.3.2. Entrada de novos dispositivos na rede

Roteadores e *end-devices* podem entrar em uma rede já existente. Para entrar, estes dispositivos precisam fazer uma varredura ativa ou passiva, para enxergar todos os coordenadores e roteadores ao redor capazes de recebê-lo como “filho”. Quando executando uma varredura ativa, o dispositivo transmite um pedido de *beacon* (comando MAC com identificador 0x07). Todos os roteadores que receberem o pedido – incluindo o coordenador, deverão enviar de volta um *frame beacon* com as propriedades da sua rede PAN. O processo de varredura ativa é repetido por vários canais. Com uma lista de roteadores possíveis para se associar, o dispositivo deverá escolher aquele que melhor o convém – a especificação Zigbee também não determina como será escolhido o roteador, ficando a cargo do desenvolvedor.

Após a escolha de um roteador em específico a se associar, o dispositivo enviará um pedido de associação (comando MAC com identificador 0x01) e receberá uma resposta de associação (comando MAC com identificador 0x02) deste roteador, com a aceitação ou rejeição desta associação.

## 4.4. Roteamento das Mensagens

A especificação Zigbee descreve diferentes modos de entregar uma mensagem de um nó a outro. Entre os mais importantes estão os roteamentos *broadcast*, em malha e em árvore.

### 4.4.1. Broadcast

Uma rota *broadcast* é nada mais do que o envio de uma mensagem a todos os dispositivos da rede. Entretanto, nem sempre todos os dispositivos da rede estão dentro do raio de alcance de transmissão. Então, para que a mensagem chegue aos dispositivos fora de alcance é preciso que os roteadores retransmitam-nas adiante, expandindo a difusão.

Inicialmente, é definido um raio de alcance para a mensagem a ser difundida, que é o número de retransmissões que esta mensagem sofrerá. Cada roteador deverá decrementar o valor do raio em 1 antes de transmiti-la. Quando o raio alcançar o valor de 0, o *broadcast* se encerrará e a mensagem não irá mais ser retransmitida pelo próximo roteador.

Na verdade, há quatro modos de *broadcast* identificados por diferentes endereços destino:

- Endereço NWK destino `0xffff`: difundido para toda a rede
- Endereço NWK destino `0xffffd`: difundido para todos os dispositivos acordados (que não estão em modo de suspensão)
- Endereço NWK destino `0xffffc`: difundido para todos os roteadores.
- Endereço NWK destino `0xffffe`: difundido somente para roteadores com baixa potência.

É importante ressaltar que apenas roteadores e coordenadores têm a capacidade de retransmitir uma mensagem adiante, agindo como um Repetidor de Sinal.

### 4.4.2. Roteamento em malha

O roteamento em malha usa roteadores intermediários para criar uma rota por onde a mensagem possa passar até chegar ao dispositivo alvo, que está fora do alcance de transmissão do transmissor. Para saber quais roteadores deverão ser usados na rota, o Zigbee apresenta o conceito de tabelas de roteamento ou, em inglês, *Routing Tables*.

#### 4.4.3. Tabelas de Roteamento

Basicamente, as tabelas de roteamento são armazenadas pelos roteadores e compostas pelo endereço de destino da mensagem, pelo endereço do próximo roteador na rota e por *flags* de controle.

A tabela de roteamento não armazena o endereço de todos os roteadores dentro da rota, mas somente do próximo roteador na lista da rota. Armazenar todos os roteadores pertencentes à rota significa perder espaço da memória desnecessariamente, pois basta se saber o próximo roteador que deverá repetir a mensagem adiante.

Por exemplo, se um *end-device* quer enviar uma mensagem, ele repassa-a para o seu roteador-pai, que mantém uma tabela de roteamento. O roteador-pai verifica se o endereçamento destino, junto com o próximo roteador, é encontrado dentro da sua tabela. Se sim, a mensagem é retransmitida para o próximo roteador, que fará o mesmo processo até a mensagem chegar ao alvo. Desta maneira, cada roteador não tem conhecimento de todos os dispositivos que compõem a rota, mas somente o endereço do próximo roteador neste caminho.

Caso não haja nenhuma referência sobre o endereço destino dentro de sua tabela, o roteador terá que executar a técnica descoberta de rota para encontrar uma rota até o destino e adicioná-la em sua tabela de roteamento.

Destaca-se que somente roteadores mantêm tabelas de roteamento e que *end-devices* sempre enviarão mensagens diretamente para seus roteadores-pai, com o endereço destino da mensagem, respectivamente.

#### 4.4.4. Descoberta de rota e Custo de Link

No roteamento em malha, na primeira vez que dois dispositivos estabelecem comunicação, é preciso descobrir uma rota entre eles por meio de um processo chamado Descoberta de Rota. A criação de uma rota, usando roteadores, só é necessária se o receptor está fora do alcance de transmissão do transmissor.

Inicialmente, o emissor não sabe como transmitir até o destino, e por isso difunde pela rede uma mensagem com pedido-de-rota (comando NWK com identificação 0x01). Esta mensagem de pedido-de-rota contém uma variável **custo** que é, inicialmente, igual a 0.

Os roteadores que receberem o pedido-de-rota adicionam um custo a ela e retransmitem-na. Assim, a cada repetição do pedido-de-rota o custo é incrementado, até o pedido chegar ao destinatário final.

O custo indica a qualidade de *link*, ou seja a qualidade da conexão entre dois dispositivos. Quanto menor é este custo melhor é a qualidade de conexão entre dois dispositivos.

A fórmula do custo  $C(l)$  para um *link* entre dois dispositivos:

$$C(l) = \begin{cases} 7 \\ \min[7, \text{round}\left(\frac{1}{P_l^4}\right)] \end{cases}$$

A variável  $P_l$  é a probabilidade de um pacote ser entregue no *link*,  $l$ . O valor  $C(l)$  é uma função definida dentro do intervalo de [0 ... 7], com 7 indicando a pior qualidade de link possível.

A função `min()` seleciona no menor número dentro de um conjunto e a função `round()` arredonda um número real para o número inteiro mais próximo.

O Zigbee não especifica como será medida e estimada a probabilidade  $P_l$ . A especificação apenas cita que uma forma de se estimar esta probabilidade é usar a média dos LQI recebidos em cada pacote de dados.

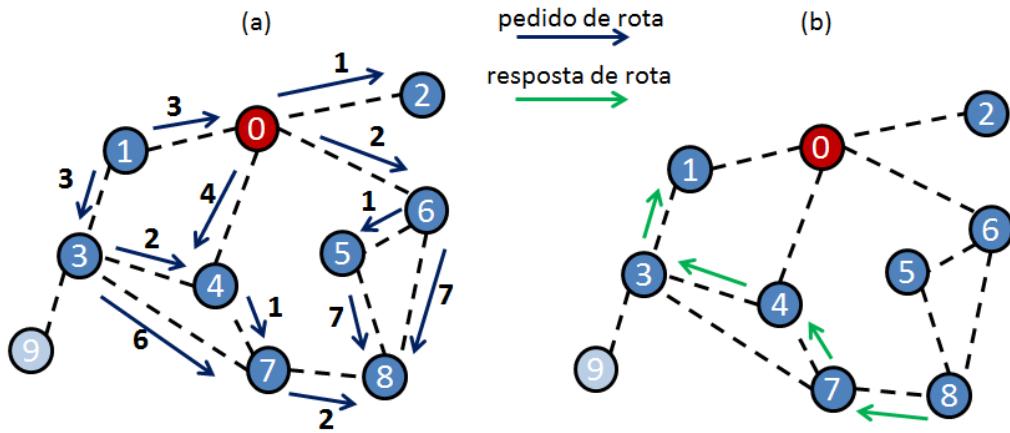
O custo total para um caminho (rota),  $C(P)$ , formado por  $n$  *links*  $[l_1, l_2, \dots, l_n]$  é:

$$C(P) = C[l_1, l_2, \dots, l_n] = \sum_{i=1}^n C(l_i)$$

É provável que vários pedidos-de-rota, oriundos de diferentes retransmissões, cheguem a um roteador. Caso isto ocorra, o roteador deverá retransmitir somente aquele que apresentar o menor custo. Também é provável que diferentes pedidos-de-rota cheguem ao destinatário final, em diferentes instantes de tempo. Por isso, é importante que o destinatário final espere um tempo antes de escolher o pedido-de-rota com menor custo total. O pedido-de-rota que chegar ao destinatário com menor custo final representará a melhor rota, já que significa em uma qualidade melhor de conexão entre os dispositivos.

Ao final da descoberta de rota, o destinatário deverá enviar uma resposta-de-rota (comando NWK número 0x02) de volta ao emissor, através da rota escolhida.

A Figura 13 ilustra o procedimento de descoberta de rota com o custo em cada *link* (representado pelas linhas tracejadas).



**Figura 13–Descoberta de Rota com os custos de link – figura (a): pedido-de-rota; figura (b): resposta-de-rota**

No exemplo da Figura 13, o dispositivo número 1 deseja transmitir para o número 8. Foram encontradas possíveis rotas com os custos totais de cada uma.

$$1 \rightarrow 3 \rightarrow 7 \rightarrow 8 \text{ com } C(P) = 11$$

$$1 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 8 \text{ com } C(P) = 8$$

$$1 \rightarrow 0 \rightarrow 4 \rightarrow 7 \rightarrow 8 \text{ com } C(P) = 10$$

$$1 \rightarrow 0 \rightarrow 6 \rightarrow 5 \rightarrow 8 \text{ com } C(P) = 13$$

$$1 \rightarrow 0 \rightarrow 6 \rightarrow 8 \text{ com } C(P) = 12$$

O melhor caminho será por  $1 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 8$  que tem o menor custo total  $C(P) = 8$ , e é por ele que o dispositivo número 8 enviará uma resposta-de-rota para o dispositivo número 1.

#### 4.4.5. Roteamento em árvore

De acordo com a especificação, se um dispositivo não tem capacidade de roteamento em malha ele deve rotear através da estrutura em árvore usando um roteamento hierárquico.

No roteamento em árvore ou hierárquico, um emissor precisa descobrir se a mensagem que ele está a enviar, para chegar ao destinatário final, precisa descer a estrutura em árvore (ir em direção a uma profundidade maior na rede) ou subir a árvore em direção à raiz (ir em direção a uma profundidade menor). Esta decisão é tomada lançando mão do método *Cskip* de endereçamento (visto na Subseção 4.2.2).

Um roteador na árvore, com endereço  $Ae$  na profundidade  $d$ , deverá enviar a mensagem para baixo se a seguinte expressão lógica for verdadeira:

$$A < D < A + Cskip(d - 1)$$

Onde  $D$  é o endereço destino da mensagem e  $Cskip(d - 1)$  o valor *Cskip* do pai do roteador  $A$ .

Se a expressão for falsa, o roteador deverá repassar a mensagem para o seu pai, em um nível acima. Mas, caso a expressão seja provada correta, o roteador deverá mandar a mensagem para um nível abaixo, mesmo que este nível ainda não seja o do destinatário final.

O endereço,  $N$ , do próximo dispositivo a receber a mensagem, no nível abaixo, pode ser encontrado como uma função de outros parâmetros:

$$N = A + 1 + \frac{[D - (A + 1)]}{Cskip(d)} * Cskip(d)$$

Se o destinatário final já estiver no próximo nível abaixo do roteador, então:

$$N = D$$

#### 4.4.6. Outros modos de Roteamento

**Roteamento para um vizinho:** Se o endereço do destino é marcado na tabela de vizinhança (*neighbor table*) do transmissor, não é preciso fazer nenhum esquema de roteamento. Basta que a mensagem será enviada diretamente ao vizinho, sem precisar de rotas com roteadores – só não é possível se um *end-device* quer transmitir para outro *end-device*.

Um dispositivo armazena informações sobre seus vizinhos e a qualidade de conexão com ele na sua tabela de vizinhança. Algumas destas informações armazenadas são: o endereço IEEE, o endereço NWK, a qualidade LQI, o custo de *link*, o tipo de dispositivo que o vizinho é (coordenador, roteador, *end-device*) e o tipo de parentesco com o vizinho (o vizinho é seu pai, o vizinho é seu filho, o vizinho é seu irmão).

As terminologias pai-filho-irmãos são utilizadas para designar a relação entre os dispositivos da rede. Quando o roteador tem outros dispositivos associados a ele, ele é chamado de pai, e os dispositivos a ele associados são seus filhos. Já a relação entre os filhos de um mesmo pai é de irmãos – dispositivos que se associam a rede através de um mesmo pai.

## 5. SIMULAÇÕES

Neste capítulo serão feitas algumas simulações sobre o Zigbee, usando ou ferramentas disponíveis online ou construindo o próprio modelo em Matlab/Simulink. As simulações remetem a redes Zigbee na topologia em árvore; ao método de checagem de erro por CRC; e às funções de modulação e espalhamento espectral de um transceptor de 2,4 GHz Zigbee.

### 5.1. Teoria a respeito das simulações

#### 5.1.1. Modo Beacon para rede na topologia em árvore

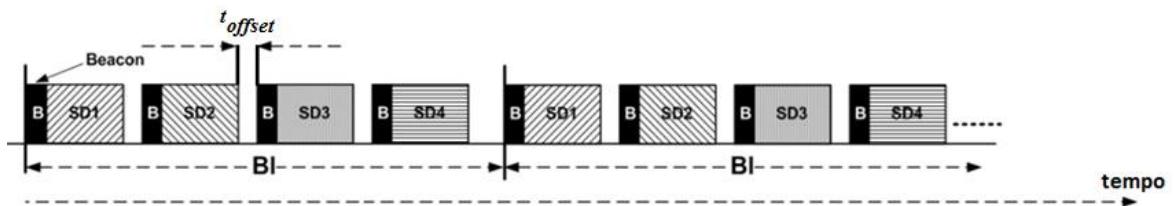
Em redes organizadas na topologia em árvore e operando no modo *beacon*, roteadores e o coordenador difundem sinais *beacons* periodicamente para trocar dados com seus “filhos”. Os “filhos” monitoram os sinais *beacons* de seu “pai” para manter a sincronização temporal e trocar dados com ele. Com vários roteadores na rede, é provável que haja a transmissão simultânea de sinais *beacons*, que podem colidir uns com os outros ou com frames de dados, causando perda de sincronização na rede. Este problema de perda de sincronismo entre pai e filho pode ser resolvido de vários modos.

Em um dos modos, o roteador deve evitar transmitir seu sinal *beacon* no mesmo momento em que seus vizinhos ou sobre o período ativo do superquadro de um deles. Assim, roteadores vizinhos devem intercalar os seus períodos ativos de superquadro. Por exemplo, digamos que um novo roteador começou a participar de uma rede. Primeiramente, ele deverá determinar o momento em que ocorrem os *beacons* e os períodos ativos do superquadro de seus vizinhos (listados na tabela de vizinhança). O roteador deverá, a partir de então, transmitir seus sinais *beacon* sem um momento em que não esteja ocorrendo nenhum outro período ativo de seus vizinhos. À vista disto, os roteadores só precisam se preocupar com o período ativo de seus vizinhos e não de todos os roteadores da rede. Deste modo, é permitido que ocorra a sobreposição de períodos ativos entre roteadores não-vizinhos. Mas(YEN, LAW e PALANISWAMI) apontaram um possível problema ao fazer isto, pois mesmo que dois roteadores não sejam vizinhos (dentro do raio de alcance um do outro) eles podem compartilhar vizinhos em comum ou vir a compartilhar no futuro. E caso os perí-

odos ativos destes dois roteadores ocorram simultaneamente, este vizinho em comum seria dessincronizado.

Um segundo modo, mais fácil e direto, é evitar a sobreposição dos períodos ativos entre quaisquer pares de roteadores da rede, independentemente de serem vizinhos ou não, e mesmo que uma sobreposição não levasse a uma colisão de *beacons*. Este método é bem seguro, entretanto não é viável quando há um grande número de roteadores, já que o intervalo de tempo reservado para cada período ativo seria muito pequeno. A Figura 14 mostra o funcionamento deste esquema.

O simulador “Worst-Case Dimensioning of Cluster-Tree Wireless Sensor Networks” (que será detalhado na Subseção 5.2) intercala os períodos ativos usando o segundo modo.



**Figura 14 - Esquema para intercalar transmissões beacon em redes cluster-tree da maneira fácil – modificada da fonte (KOUBAAA, ALVES e TOVAR)**

Neste segundo modo, os sinais *beacons* são iniciados em sequência. Após o término do primeiro período ativo, um segundo roteador transmite um sinal *beacon*, e após o término do segundo período ativo, um terceiro roteador transmite seu sinal *beacon*, e assim por diante. Para que este sequenciamento funcione em uma rede com  $N$  roteadores, todos eles com mesmos valores de *Superframe Order*,  $SO$ , e *BeaconOrder*,  $BO$ , a seguinte afirmação deve ser verdadeira:

$$BI \geq N * SD$$

Onde para um roteador que use um transceptor de 2,4GHz (1símbolo = 16μs):

$$BI = aBaseSuperframeDuration * 2^{BO} * 16 \mu s$$

$$SD = aBaseSuperframeDuration * 2^{SO} * 16 \mu s$$

Para uma rede onde  $BI > N * SD$ , os sinais *beacons* são separados em  $SD + t_{offset}$ , como ilustrado na Figura 14.

### 5.1.2. Técnica CRC de checagem de erros

O CRC-16 (*Cyclic Redundancy Check – 16 bits*) é um método para detectar erros em transmissões. Neste método, os bits CRC calculados pelo transmissor são adicionados ao campo FCS de um *frame MAC*, antes da transmissão.

Para se calcular os bits CRC-16 no transmissor deve seguir os seguintes passos:

1. Transformar os bits de dados (cabecalho + payload MAC) no polinômio  $B(x) = b_0x^{n-1} + b_1x^{n-2} + \dots + b_{n-1}$  de grau  $n - 1$ . Os coeficientes  $[b_0, b_1, \dots, b_{n-1}]$  do polinômio são os bits de dados, portanto, os valores possíveis serão apenas 0 ou 1.
2. Multiplicar  $B(x)$  por  $x^{16}$ . Tem-se assim um polinômio resultante  $B(x) * x^{16}$ .
3. Fazer a divisão polinomial deste polinômio resultante  $B(x) * x^{16}$  pelo polinômio gerador  $G(x) = x^{16} + x^{12} + x^5 + 1$  (Este polinômio gerador é específico para o CRC-16).
4. Os coeficientes do polinômio resto  $R(x) = r_0x^{15} + r_1x^{14} + \dots + r_{15}$  da divisão são os 16 bits CRC. Como o campo FCS do frame MAC tem 16 bits, então o polinômio resto sempre terá 16 coeficientes.

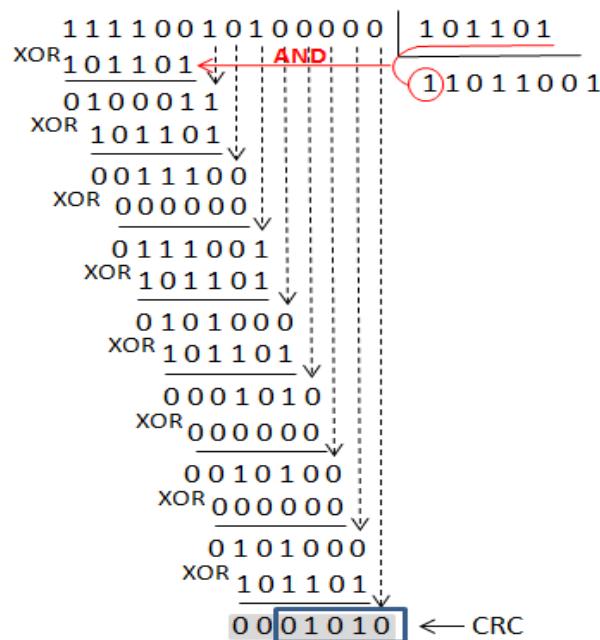
Ao receber o *frame MAC*, o receptor também deve executar o cálculo CRC seguindo os mesmos quatro passos no transmissor. Entretanto os bits de dados recebidos corresponderão ao (cabecalho + payload + campo FCS). Isto resultará em um polinômio  $B(x)$  igual a  $B(x) = b_0x^{n-1} + b_1x^{n-2} + \dots + b_{n-1}x^{16} + r_0x^{15} + r_1x^{14} + \dots + r_{15}$

Se no receptor, o passo 4 resultar no polinômio resto nulo,  $R(x) = 0$ , os dados recebidos são exatamente iguais aos dados transmitidos e não ocorreram erros durante a transmissão. Caso  $R(x) \neq 0$ , o receptor deverá pedir a retransmissão dos dados, pois eles foram corrompidos.

O polinômio gerador,  $G(x)$ , é escolhido de modo a aumentar a eficiência na detecção de erros e, para o CRC-16, ele será de ordem 16.

A Figura 15 ilustra um exemplo de como calcular os bits para o CRC-5 - um exemplo para o CRC-16 seria inconveniente, pois a operação seria muito grande. No exemplo, o CRC foi calculado através de divisão binária ao invés de divisão polinomial, não tendo, assim, que converter toda a sequência binária em um polinômio numérico. O resultado, porém, é o mesmo. Esta divisão binária usa as operações de AND e ou-exclusivo (XOR).

Para os bits de dados  $B = [1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1]$  têm-se o polinômio equivalente:  $B(x) = x^8 + x^7 + x^6 + x^5 + x^2 + 1$ . E, considerando, um polinômio gerador  $G(x) = x^5 + x^3 + x^2 + 1$  para o CRC-5, o cálculo seria:



**Figura 15 – Exemplo de cálculo CRC**

No dividendo, foram acrescidos cinco zeros aos bits de dados, que representam a multiplicação do polinômio  $B(x)$  por  $x^5$ , como deve ser feito.

### 5.1.3. Modulações e técnica DSSS de espalhamento espectral

A **modulação BPSK**<sup>2</sup> é o processo na qual a fase angular da portadora é alterada em 180 graus (e.g., 0° e 180°) quando há uma troca de bits do sinal modulante (neste caso o sinal modulante digital são os próprios bits de um pacote de dados). Cada uma das fases representará um valor de um bit: 0 ou 1, assim, quando o sinal for demodulado no receptor, a fase 0° será traduzida como um bit 0, e uma fase de 180° será traduzida como um bit 1, por exemplo. No BPSK cada fase do sinal modulado corresponde a apenas um bit, ou seja, há apenas duas fases, separadas por veia volta.

Uma variação da modulação BPSK é a QPSK (*Quadrature-PSK*) que aumenta a eficiência da modulação em fase, ao possibilitar que o sinal modulado varie em quatro fases separadas por 90 graus (e.g., -135°, +135°, -45° e +45°), cada uma representando dois bits do sinal modulante: 00, 01, 10 ou 11. Assim neste sistema, uma fase corresponde a dois bits.

Para produzir dois bits por fase, modulador QPSK separa os bits de dados em duas sequências distintas: a dos bits pares e a dos bits ímpares. As duas sequências são igualmente temporizadas no circuito modulador, e não há nenhum atraso entre elas. Cada sequência modula uma das duas portadoras que estão separadas em 90° (sinais seno e cosseno), produzindo dois sinais modulados distintos. Ao final do processo de modulação, o modulador soma os dois sinais modulados, formando um sinal modulado único que será transmitido ao meio. Neste esquema, o sinal modulado pode ter uma variação repentina de fase de até 180°, causando uma grande flutuação de amplitude do sinal no momento de transição de fase - se A é a amplitude do sinal, a flutuação pode ser de até 2\*A.

Ao contrário da QPSK, na **modulação O-QPSK**<sup>3</sup> as sequências de bits pares e ímpares não estão alinhados, mas sim atrasados em meio período de bit ou T/2, se T é o período de um bit. O restante das operações do modulador O-QPSK continua igual ao do modulador QPSK, com duas portadoras separadas em 90° sendo moduladas pelas sequências par e ímpar. Se for definido um esquema em que as fases que representam os pares 00 e 11 estejam separadas de 180° (Exemplo: o par de bits 00 seja representado pela fase -

---

<sup>2</sup> BPSK: Binary Phase Shift Keying

<sup>3</sup> O-QPSK: Offset Quadrature Phase Shift Keying

135° e 11 seja representado pela fase +45°), a variação de fase do sinal modulado será de no máximo 90°, devido a defasagem entre as sequências par e ímpar. Ocorre que na modulação a variação de par 00 para 11 nunca ocorrerá, pois a transição ocorrerá em somente um dos bits. Se a variação de fase do sinal modulado for de no máximo 90°, a diferença de amplitude do sinal será metade daquela produzida no QPSK - se A é a amplitude do sinal, a flutuação pode ser de até A, também, como mostrado na Figura 16 - modulação O-QPSK e BPSK - retirada da fonte .

Na modulação O-QPSK uma fase do sinal modulado corresponde a dois bits.

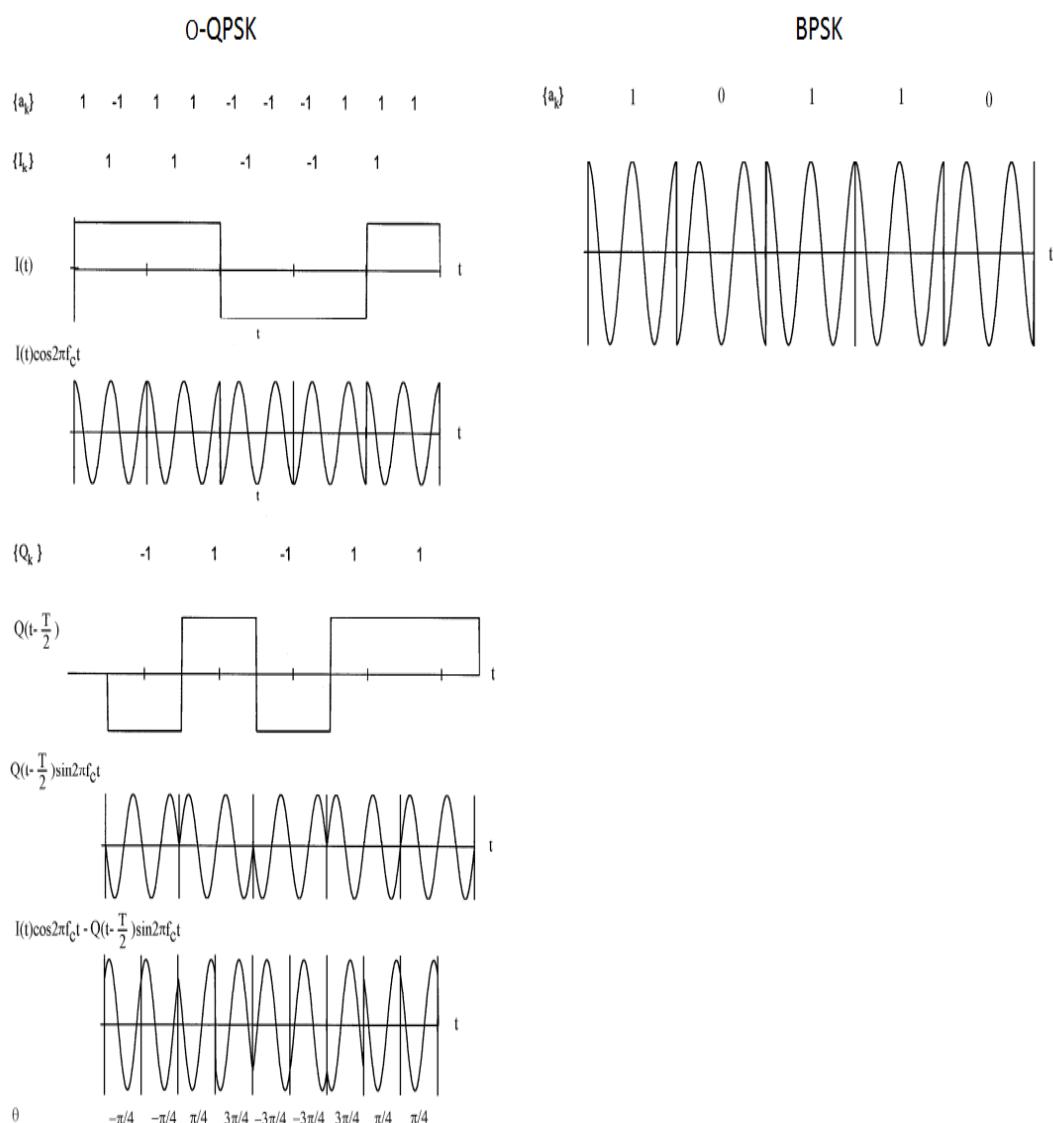
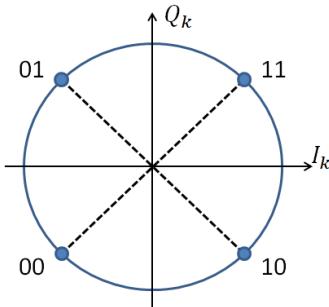


Figura 16 - modulação O-QPSK e BPSK - retirada da fonte (XIONG, 2006)

Como mostrado na Figura 16, o modulador O-QPSK separa os bits de entrada  $\{a_k\}$ , nas sequências de bits ímpar  $\{I_k\}$  e de bits par  $\{Q_k\}$ , para poder gerar dois bits por fase. Já na BPSK isto não ocorre, pois apenas uma fase é gerada por bit.

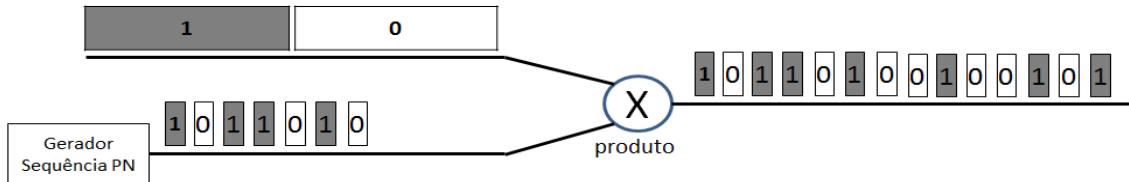
A modulação BPSK é usada em transceptores de rádio que operam em 868/915 MHz, enquanto a modulação O-QPSK é usada em transceptores de rádio que operam no espectro de 2,4 GHz.



**Figura 17 – Diagrama de fase da modulação O-QPSK**

A **técnica DSSS** (*Direct Sequence Spread Spectrum*) é uma técnica de espaçamento espectral que consiste em codificar cada bit de um sinal binário em conjunto de bits com menor período de amostragem. Estes bits “menores” são chamados de *chips*, para diferenciá-los da unidade básica de informação: o bit. A codificação é feita multiplicando-se cada bit por uma sequência de chips chamada sequência PN (do inglês, *Pseudo Noise Sequence*). Quanto maior o tamanho da sequência PN, maior é o código representativo de um bit.

A taxa de transmissão de *chips/s* será equivalente à taxa *bits/s* vezes o tamanho (número de chips) da sequência PN. Este processo de troca de bits para chips é chamado de *codificação de chip*.



**Figura 18 - Exemplo do funcionamento da técnica de multiplicação dos bits por uma sequência PN**

No IEEE 802.15.4, nas frequências de 868/915 MHz, cada bit é codificado por uma sequência PN de 15 *chips*. O código gerado para um bit 1 será, binariamente, complementar ao código gerado para o bit 0 e, por isso, só há dois códigos, um para cada valor de bit. Esta técnica é chamada apenas de DSSS, pois há apenas dois códigos complementares de 15 chips. Neste caso, a taxa *chips/s* é quinze vezes maior que a taxa de *bits/s*.

Já em 2,4 GHz, quatro bits são codificados por 32 chips. Um conjunto de quatro bits possibilita até  $2^4 = 16$  sequências PN diferentes. Esta técnica que estabelece 16 sequências PNs diferentes é chamada de 16 *quasi-orthogonal* DSSS.

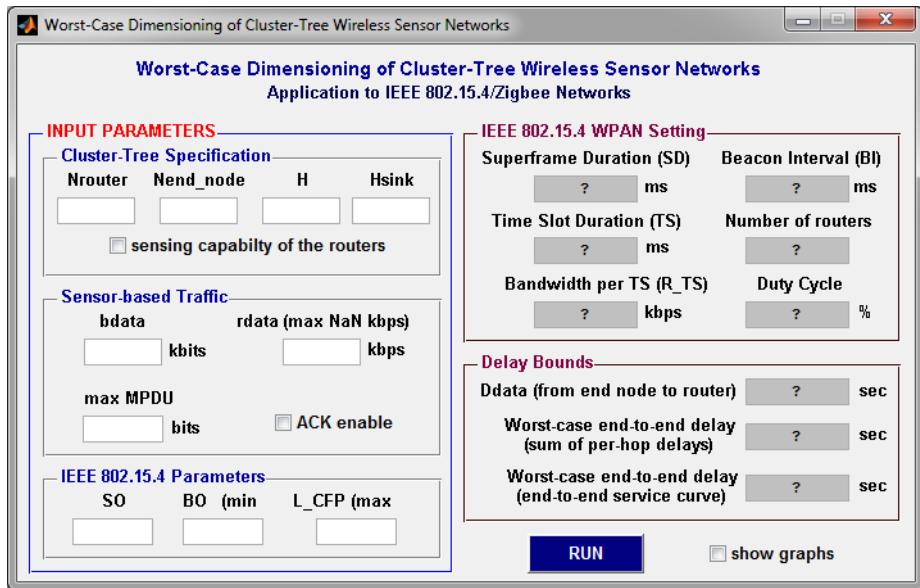
As técnicas de espalhamento espectral aumentam a resistência do sinal a interferências e a interceptação – usadas principalmente pelos militares. A codificação de chips diminui a probabilidade de erros de recepção, em virtude de uma quantidade redundante de chips. Mesmo que haja alteração de um ou outro chip, o receptor ainda conseguirá interpretar e recuperar os bits originais, como será mostrado na simulação de um transceptor de 2,4 GHz (ver Subseção 5.4). Outras vantagens da técnica de espalhamento espectral são: menor densidade espectral, em consequência do espalhamento do sinal ao longo de uma ampla banda e compartilhamento da mesma banda de frequência por vários dispositivos.

Deste modo, o transceptor Zigbee condiciona o sinal binário codificando-o em chips e, posteriormente, modulando os chips na modulação BPSK ou O-QPSK.

## 5.2. Simulação de uma rede WSN Cluster-Tree em modo beacon

Será feita uma simulação de uma rede WSN/IEEE 802.15.4, organizada na topologia em árvore, que usa o modo *beacon* para trocas de dados entre os dispositivos.

O simulador escolhido para simular tal rede foi o “*Worst-Case Dimensioning of Cluster-Tree Wireless Sensor Networks*”, disponível para download em [www.open-zb.net](http://www.open-zb.net) na versão V1.0. Ele roda sobre o MATLAB 7.3 ou posteriores e oferece uma interface gráfica GUI (*Graphical User Interface*). A GUI do MATLAB permite que as tarefas sejam executadas interativamente em uma janela, sem a necessidade do usuário usar comandos ou ambientes de programação.



**Figura 19 – Janela do simulador Worst-Case Dimensioning of Cluster-Tree Wireless Sensor Networks**

Com a interface gráfica o usuário configura a rede através dos parâmetros de entrada, localizado na caixa *Input Parameters* da janela do simulador.

$N_{router}$ : máximo número de roteadores que podem se associar a um roteador-pai.

$N_{end-node}$ : máximo número de end-nodes que podem se associar a um roteador-pai. O simulador considera que os *end-nodes* como dispositivos do tipo *end-device* e são eles que possuem sensores para fazer as medições de sensoriamento.

$Height(H)$ : altura máxima da rede. A contagem da profundidade inicia-se em 0, na raiz da estrutura em árvore, onde está localizado o coordenador. Os roteadores localizados no nível mais baixo da rede têm uma profundidade igual ao valor de  $Height$ , e os end-nodes associados a estes roteadores estarão na profundidade  $Height + 1$ .

$Hsink$ : profundidade do roteador que abriga o dispositivo *sink*. O *sink* é um dispositivo autônomo e móvel pela rede responsável por reunir/coletar todos os dados medidos pelos nós-sensores (*end-nodes*). Ele se associa a um roteador e é o dispositivo que agrupa todas as medições para posterior processamento dos dados.

*Sensing capability of the routers:* se os roteadores também possuem sensores, eles são ditos com capacidade de sensoriamento. Caso contrário, eles só retransmitirão as medições feitas pelos *end-nodes*.

$r_{data}$ : taxa média de transferência das medições dos sensores, que chega até os *end-nodes*.

$máxMPDU$ : máximo tamanho de um frame MAC configurado pelo usuário.

*Ack enable*: configuração que indica se os dispositivos usarão uma mensagem de reconhecimento, *Ack*, a cada conjunto de dados recebido.

$SO$ : expoente usado para calcular o período ativo do superquadro. O expoente  $SO$  corresponde ao valor no atributo *macSuperframeOrder*.

$BO$ : expoente usado para calcular o período superquadro (período entre dois *beacons*). O expoente  $BO$  corresponde ao valor no atributo *macBeaconOrder*. A seguinte relação deve ser sempre mantida:  $0 \leq SO \leq BO \leq 14$

$L_CFP$ : número de slots reservados na parte CFP (*Contention Free Period*) dentro do período ativo do superquadro. O período ativo do superquadro é subdividido em 16 slots com mesmo tamanho. A especificação IEEE 802.15.4 relata que pelo menos um slot deverá ser deixado para o período CAP (*Contention Access Period*), apesar que reservar apenas um slot não é viável na prática.

Na caixa IEEE 802.15.4 WPAN *Setting*, no lado direito da janela da Figura 19, é exibida a configuração do superquadro para os parâmetros de entrada escolhidos.

$SD$ : intervalo de tempo, em milissegundos, entre dois sinais *beacons* de um roteador.

$BI$ : duração, em milissegundos, do período ativo do superquadro. O período ativo é compreendido pelo *beacon*, CAP mais CFP.

*Duração do time slot*: duração, em milissegundos, de um slot.

*Duty Cycle*: é dado pela razão entre  $SD$  e  $BI$ , indicando a relação entre período ativo e período total do superquadro.

Após rodar a simulação, o simulador usa o modelo proposto nos trabalhos de (KOUBAA, ALVES e TOVAR)e (LE BOUDEC e THIRAN, 2012) para dimensionar os requisitos de recurso da rede e analisar o desempenho de tempo. A explicação do modelo ficará fora do escopo deste trabalho.

### 5.2.1. Procedimentos para Simulação

Foram simulados três cenários diferentes no simulador “Worst-Case Dimensioning...”. O primeiro cenário envolve a variação do número de end-nodes por pai,  $N_{end-node}$ . O segundo cenário envolve a variação do número de roteadores por pai,  $N_{router}$ . E o terceiro cenário refere-se a variação sobre o parâmetro altura,  $Height$ , da rede. Todos os três parâmetros,  $N_{end-node}$ ,  $N_{router}$  e  $Height(H)$ , são variáveis utilizadas pelo simulador para configurar a rede em árvore.

**Primeiro Cenário:** O primeiro cenário simula a aplicação de uma rede de sensoriamento, em que a quantidade de *end-nodes* pode variar, dependendo da necessidade de mais ou menos sensores. Somente o número de *end-nodes*,  $N_{end-nodes}$ , será alterado, assim o número de roteadores por pai,  $N_{router}$ , e a altura da rede,  $Height$ , não serão modificados.

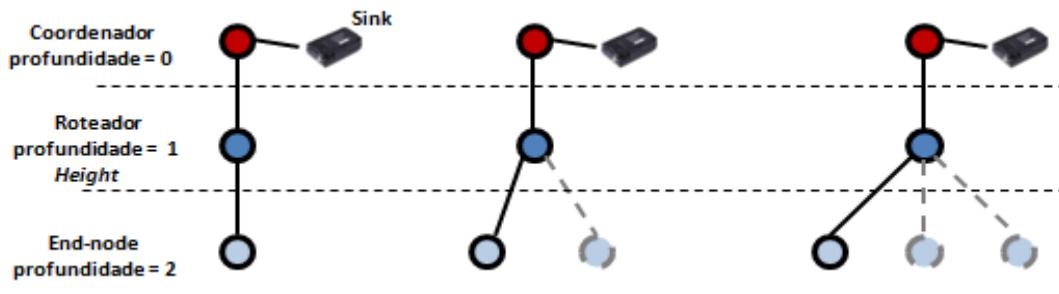


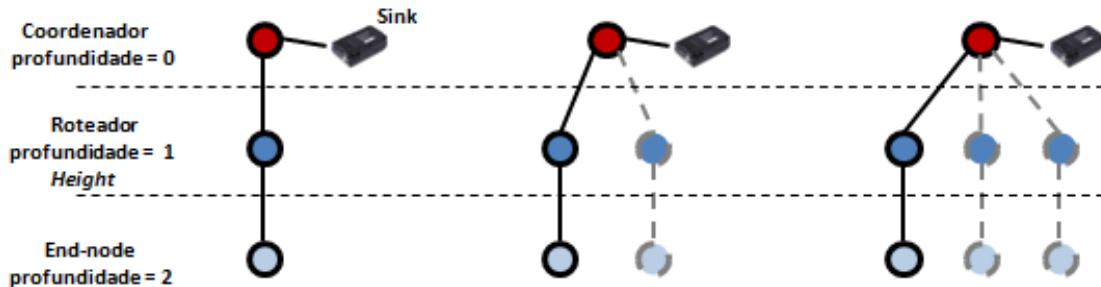
Figura 20 - Cenário 1 da simulação de uma rede cluster-tree no modo beacon

**Tabela 3 - Parâmetros da rede para o Cenário 1**

Parâmetros de Entrada	
Nend_nodes	1,2 e 3
Nrouter	1
H	1
Hsink	0
bdata [kbytes]	0
rdata [kbps]	1
máxMPDU [bits]	1016
SO	6
BO	8
L_CFP	8

Como mostrado na Figura 20, o dispositivo *sink* está associado ao coordenador e, portanto, Hsink é igual a 0.

**Segundo Cenário:** No segundo cenário, o único parâmetro variável será o número de roteadores por pai,  $N_{router}$ , e com variação deste parâmetro a quantidade de nós-sensores mudará juntamente conforme mostrado na Figura abaixo. Os parâmetros  $N_{end-node}$  e Height serão mantidos inalterados.

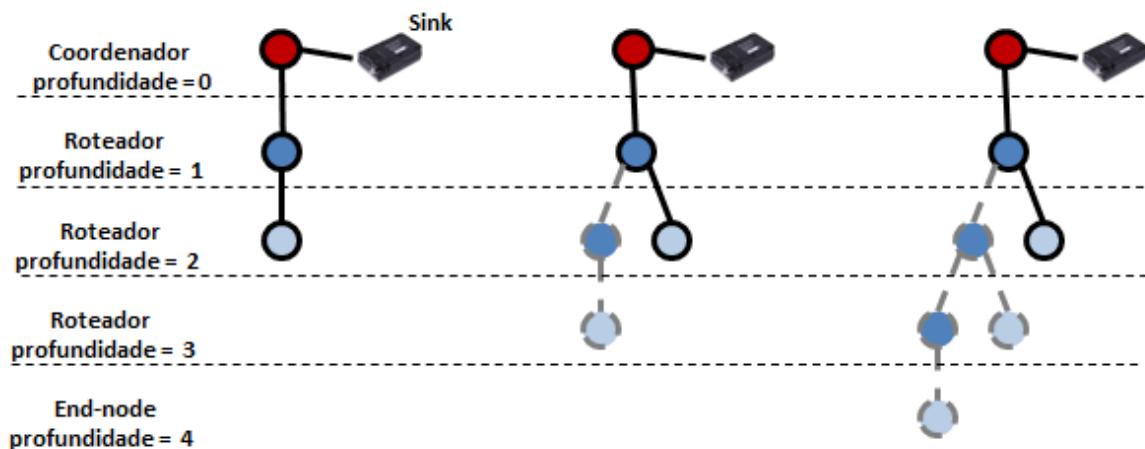


**Figura 21 - Cenário 2 da simulação de uma rede cluster-tree no modo beacon**

**Tabela 4 - Parâmetros da rede para o Cenário 2**

Parâmetros de Entrada	
Nend_node	1
Nrouters	1,2 e 3
H	1
Hsink	0
bdata [kbytes]	0
rdata [kbps]	1
máxMPDU [bits]	1016
SO	6
BO	8
L_CFP	8

**Terceiro Cenário:** O terceiro cenário estipula variações no alcance da rede, por meio de modificações na altura da rede, *Heigth*. Os valores de  $N_{end-node}$  e  $N_{router}$  serão constantes.



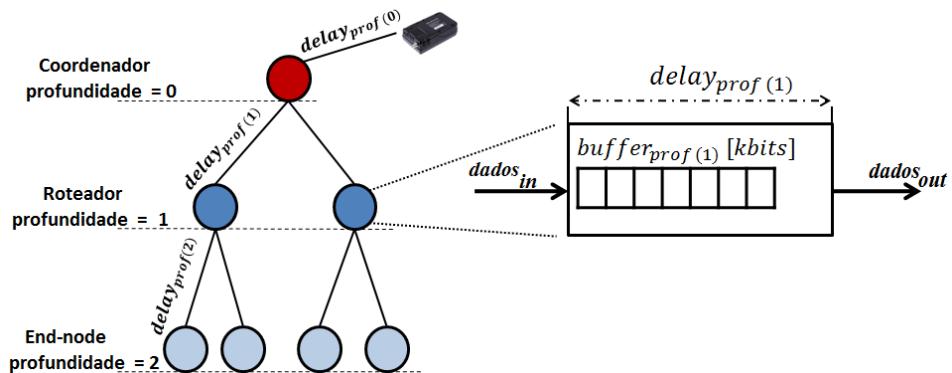
**Figura 22 - Cenário 3 da simulação de uma rede cluster-tree no modo beacon**

**Tabela 5 - Parâmetros da rede para o Cenário 3**

Parâmetros de Entrada	
Nend_node	1
Nrouters	1
H	1, 2 e 3
Hsink	0
bdata [kbytes]	0
rdata [kbps]	1
máxMPDU [bits]	1016
SO	6
BO	8
L_CFP	8

O objetivo nesses três cenários foi verificar a influência destes três parâmetros sobre o comportamento da rede em relação ao: *delay* dos dados trafegados pela rede e *buffer* necessário para os roteadores, para que não haja estouro de memória.

A Figura 23 mostra um exemplo dos parâmetros *buffer* e  $t_{delay}$  calculados pelo simulador.



**Figura 23 - Delay e Buffer do roteador na profundidade 1**

### 5.2.2. Resultados

Para os valores  $BO = 8$  e  $SO = 6$ , os parâmetros do superquadro, para os três cenários, são:

$$BI = aBaseSuperframeDuration * 2^{BO} * 16us = 960 * 2^8 * 16us = 3932,16 ms$$

$$SD = aBaseSuperframeDuration * 2^{SO} * 16us = 960 * 2^6 * 16us = 983,04 ms$$

$$Timeslot = \frac{aBaseSuperframeDuration * 2^{SO}}{16} * 16us = \frac{960 * 2^8}{16} * 16us = 61,44 ms$$

$$DutyCycle = \frac{SD}{BI} = \frac{2^{SO}}{2^{BO}} * 100 = 25 \%$$

**Tabela 6 - Valores dos parâmetros WPAN de saída**

<b>Configurações WPAN (saída)</b>	
SD [ms]	983,04
BI [ms]	3932,16
Time slot [ms]	61,44
Núm. Total de Rotead.	?
DutyCycle [%]	25

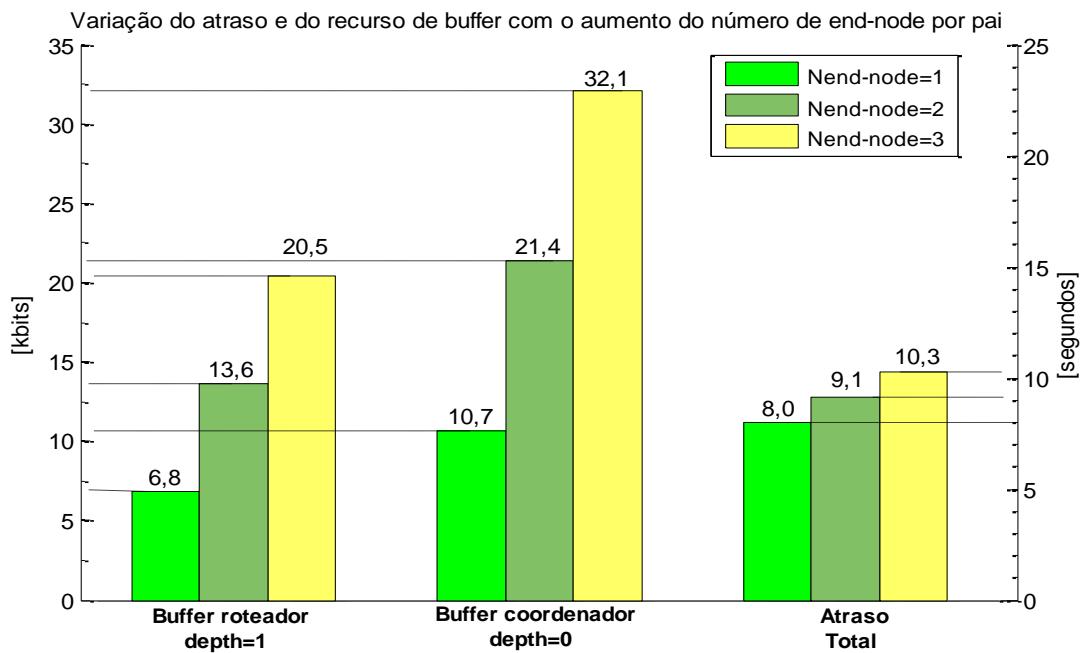
Com os dados da simulação foram construídos três gráficos (Figura 24, Figura 25 e Figura 26) - um para cada cenário.

Cada gráfico apresenta três agrupamentos. O primeiro agrupamento mostra o mínimo buffer necessário para o roteador na profundidade 1,  $buffer_{prof(1)}$ . O segundo mostra o mínimo buffer necessário para o coordenador,  $buffer_{prof(0)}$ . E o terceiro agrupamento mostra o *atraso total*, que o tempo mais longo que um dado de medição do sensor levará para chegar até o dispositivo *sink*. Os dados do(s) *end-node(s)* mais afastado do sink será o que levaram mais tempo para ser recebido pelo *sink*. Este tempo mais longo será calculado a partir do(s) *end-node(s)* mais afastado do sink. Nos três cenários o(s) *end-node(s)* mais afastado será aquele localizado na maior profundidade da rede.

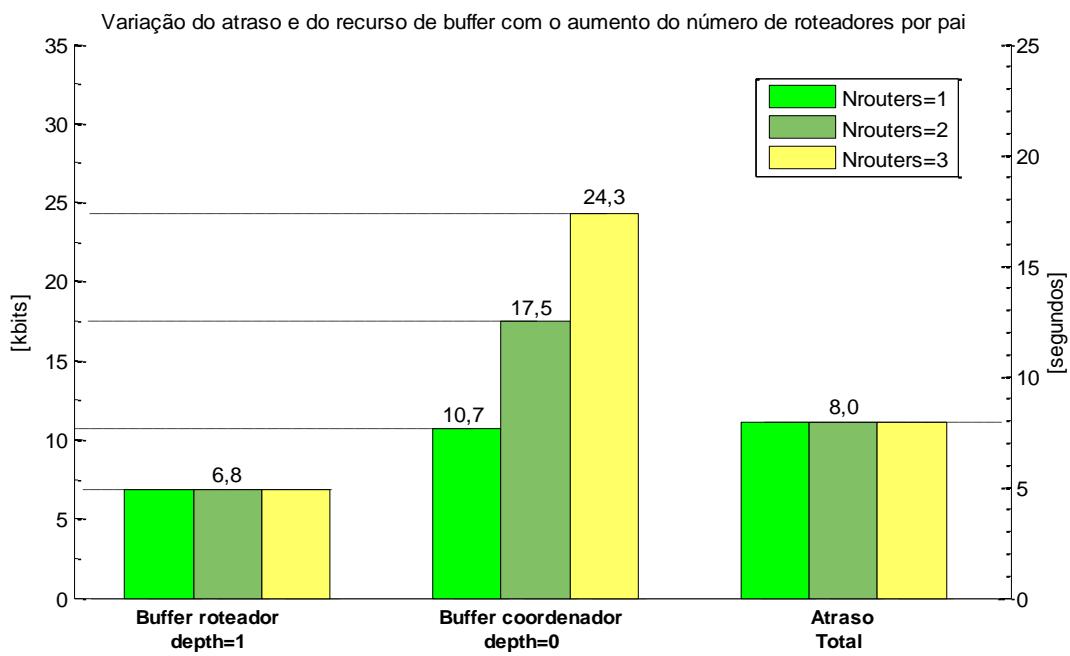
No simulador, o *atraso total* é calculado como a soma dos atrasos individuais provocados por cada dispositivo no caminho entre o *end-node* e o sink. O atraso do *end-node* também é incluído no somatório. Assim, para uma rede com altura  $H$ , o *atraso total* é:

$$AtrazoTotal = buffer_{prof\ (H+1)} + buffer_{prof\ (H)} + buffer_{prof\ (H-1)} \dots + buffer_{prof\ (0)}$$

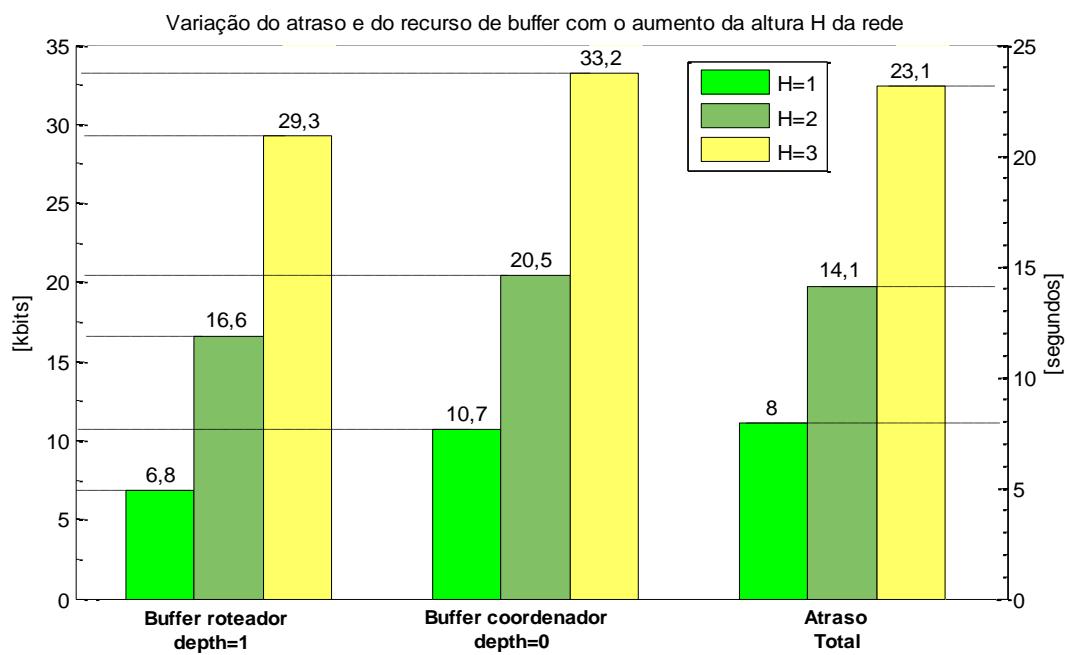
Onde  $buffer_{prof\ (H+1)}$  é o buffer do(s) *end-node*(s) localizados na máxima profundidade da rede.



**Figura 24 - Resultado da simulação para o Cenário 1**



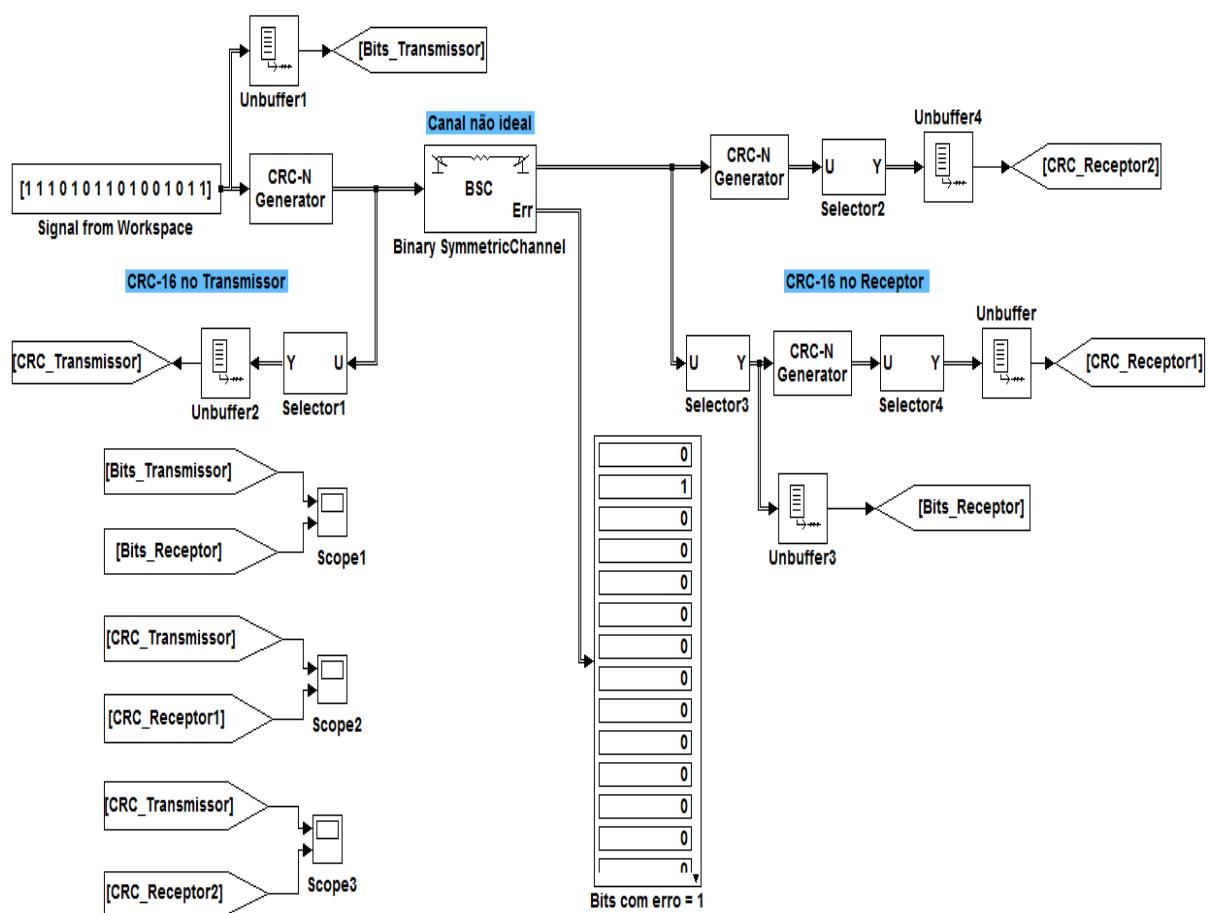
**Figura 25 - Resultado da simulação para o Cenário 2**



**Figura 26 - Resultado da simulação para o Cenário 3**

### 5.3. Simulação do algoritmo CRC-16 com Simulink

Com o sistema Simulink da Figura 27, pretendeu-se verificar o funcionamento da técnica CRC, método este utilizado para se detectar erros em transmissões de dados. O sistema simulou a transmissão de uma palavra binária sobre um canal de comunicação - aqui representado pelo bloco *Binary Symmetric Channel*. Ao lado esquerdo do bloco canal considera-se as operações CRC de um transmissor Zigbee, e sobre o lado direto considera-se as operações CRC de um receptor Zigbee.



**Figura 27 – Simulação em Simulink do funcionamento da técnica CRC**

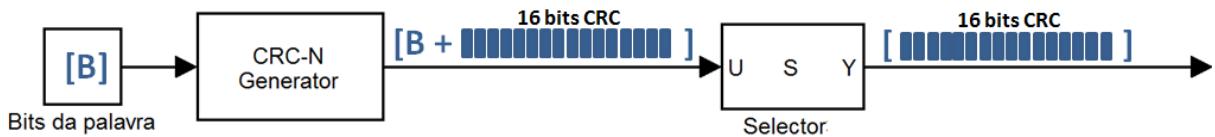
O sistema é formado pelo bloco *Signal from Workspace* que gerará uma palavra de 16 bits, simulando os dados que um dispositivo transmitirá; por blocos Geradores CRC-N que geram os bits CRC-16; por Seletores que extraem apenas determinados bits de uma sequência de bits maior; por um Display que permite visualizar quais bits da palavra binária foram alterados ao passar pelo canal; e pelo Bloco *Binary Symmetric Channel* que simula um canal estimulador de erros, com determinada probabilidade de erro de bits.

**Bloco Signal from Workspace:** bloco de entrada do sistema e que contém a palavra binária a ser enviada pelo transmissor.

**Bloco Binary Symmetric Channel:** este bloco simula um canal ruidoso, que gera uma probabilidade de erro determinística especificada no parâmetro **Error Probability** do bloco. O bloco provoca um erro mudando um bit 0 para um bit 1, ou vice-versa. O bloco apresenta uma porta de saída (**Err**) onde é possível verificar quais bits da palavra binária foram modificados ao passar pelo canal. Para um bit alterado, o Display mostrará o valor 1.

**Bloco CRC-N Generator:** no gerador de verificação de redundância cíclica, o tipo de método CRC é escolhido através do parâmetro **CRC method**. Como saída do bloco, tem-se um *frame* contendo a palavra binária mais os bits CRC. Deste modo, para retirar deste *frame* apenas os 16 bits CRC ou apenas a palavra binária é preciso usar o bloco *Selector*.

A ação dos blocos *Selector* e *CRC-N Generator* é exemplificada na Figura 28.



**Figura 28 - Exemplo de como funciona os blocos CRC-N Generator e Selector**

### 5.3.1. Procedimento para Simulação

Usando o sistema da Figura 27 foi verificado os erros causados por um canal ruidoso com probabilidade de 5% de gerar erros. A simulação representa um caso real de uma transmissão não ideal, onde erros são passíveis de ocorrer em decorrência de ruídos presentes no meio, atenuação/distorção do sinal ou interferências com outros sinais eletromagnéticos.

Com o bloco *Signal from Workspace* foi gerado uma palavra binária de 16 bits:  $[1\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1] = [b_{15}b_{14} \dots b_0]$ . Foi gerada uma palavra binária pequena, pois uma palavra formada por um grande número de bits atrapalharia a visualização no osciloscópio.

cópia, mas o Zigbee permite que até 127 bytes sejam transmitidos de uma só vez, em um pacote de dados.

Antes de passar pelo canal, o bloco *CRN-N Generator* gerou e acrescentou 16 bits CRC,  $[c_{15} c_{14} \dots c_0]$ , a palavra binária, formando a sequência  $[b_{15} b_{14} \dots b_0 + c_{15} c_{14} \dots c_0]$  na saída do bloco. O acréscimo dos bits CRC confere segurança e confiabilidade aos dados transmitidos ao meio.

### 5.3.2. Resultados

Simulado o sistema, verificou-se os sinais binários gerados em cada um dos três osciloscópio.

- **Osciloscópio 1:**

- Palavra,  $[b_{15}^t b_{14}^t \dots b_0^t]$ , de 16 bits antes de passar pelo canal ruidoso. Onde  $b_n^t$  representa o  $n - \text{éssimo}$  bit da palavra no lado do transmissor.
- Palavra,  $[b_{15}^r b_{14}^r \dots b_0^r]$ , após a palavra original passar pelo canal ruidoso. Onde  $b_n^r$  representa  $n - \text{éssimo}$  bit da palavra no lado do receptor.

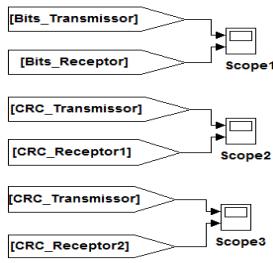
- **Osciloscópio 2:**

- 16 bits CRC no lado do transmissor, calculados somente sobre a palavra  $[b_{15}^t b_{14}^t \dots b_0^t]$ .
- 16 bits CRC no lado do receptor, calculados somente sobre a palavra  $[b_{15}^r b_{14}^r \dots b_0^r]$ .

- **Osciloscópio 3:**

- 16 bits CRC no lado do transmissor, calculados somente sobre a palavra  $[b_{15}^t b_{14}^t \dots b_0^t]$ . Mesmo sinal visualizado no osciloscópio 2.
- 16 bits CRC no lado do receptor, calculados sobre a sequência  $[b_{15}^r b_{14}^r \dots b_0^r + c_{15}^r c_{14}^r \dots c_0^r]$  que é a sequência binária presente na saída do canal ruidoso. Neste caso, em uma transmissão correta e livre de erros os bits CRC serão todos nulos:  $[00 \dots 0]$ .

Nota-se que no receptor foram feitos dois cálculos CRC diferentes, representados pelos sinais  $[CRC_{Receptor1}]$  e  $[CRC_{Receptor2}]$ , para diferenciá-los.



**Figura 29 - Três Osciloscópio da simulação CRC**

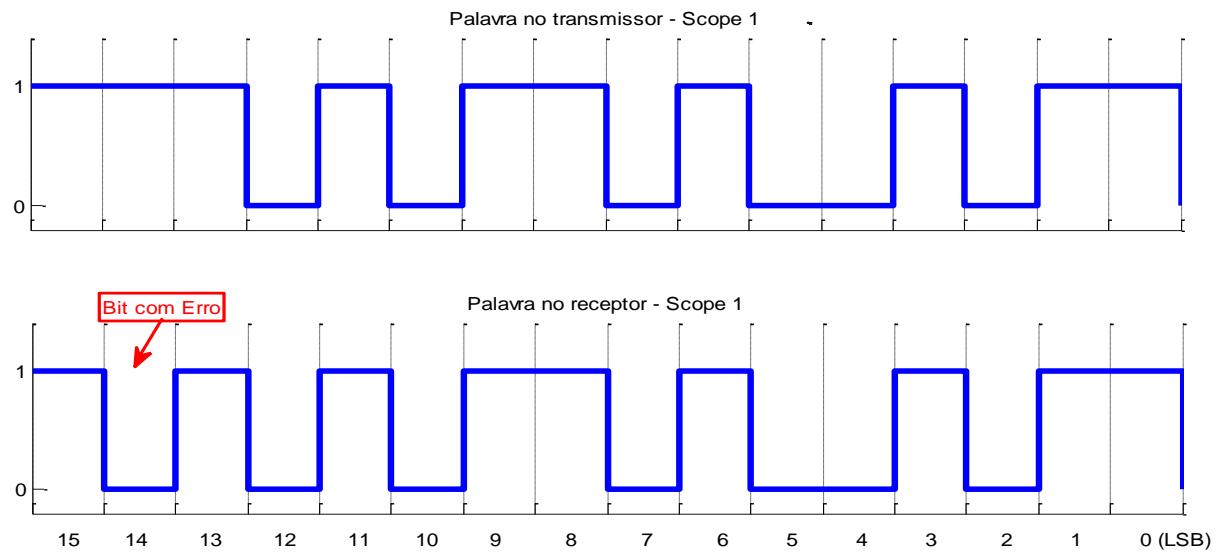
Para as modulações O-QPSK e BPSK a probabilidade de erro em um bit é:

$$P_b = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$$

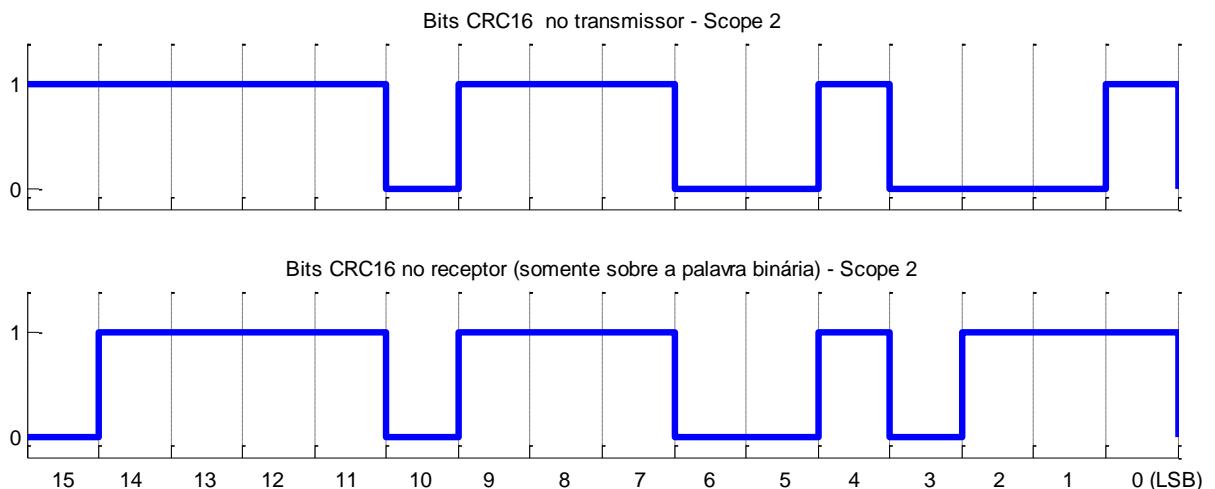
Tal que  $\operatorname{erfc}()$  é a função complementar-do-erro e  $E_b/N_0$  é a relação sinal/ruído.

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-u^2} du$$

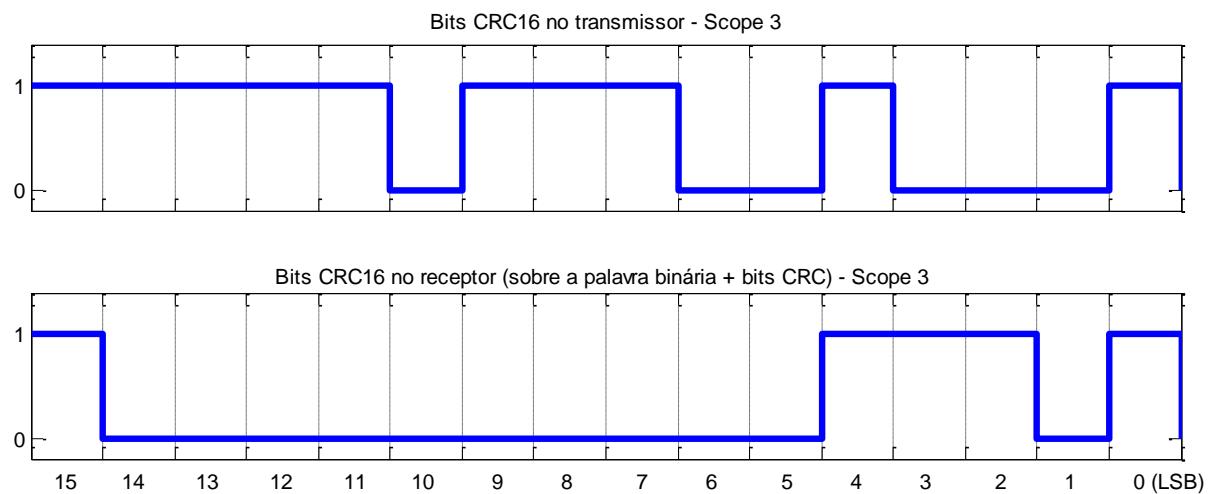
Para uma probabilidade de erro de 5%, a relação entre energia do bit e densidade espectral de potência do ruído é  $E_b/N_0 = 1,31 \text{ dB}$ .



**Figura 30 - Palavra binária transmitida e recebida em um canal com 5% de probabilidade de erro de bit**



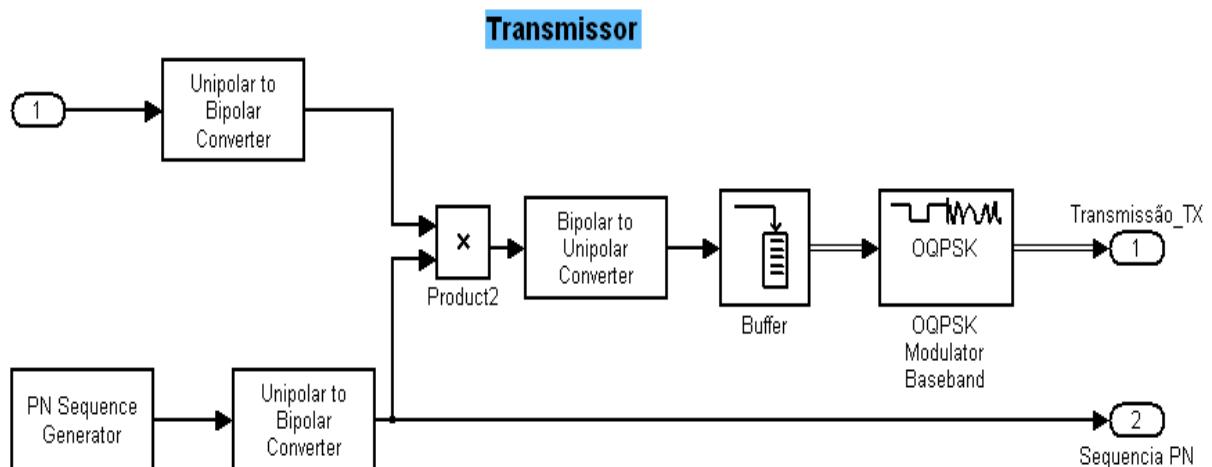
**Figura 31 - Bits da CRC no receptor calculados sobre a sequência  $[b_{15}^r \ b_{14}^r \ \dots \ b_0^r]$  no receptor.**  
Transmissão sobre um canal com 5% de probabilidade de erro de bit



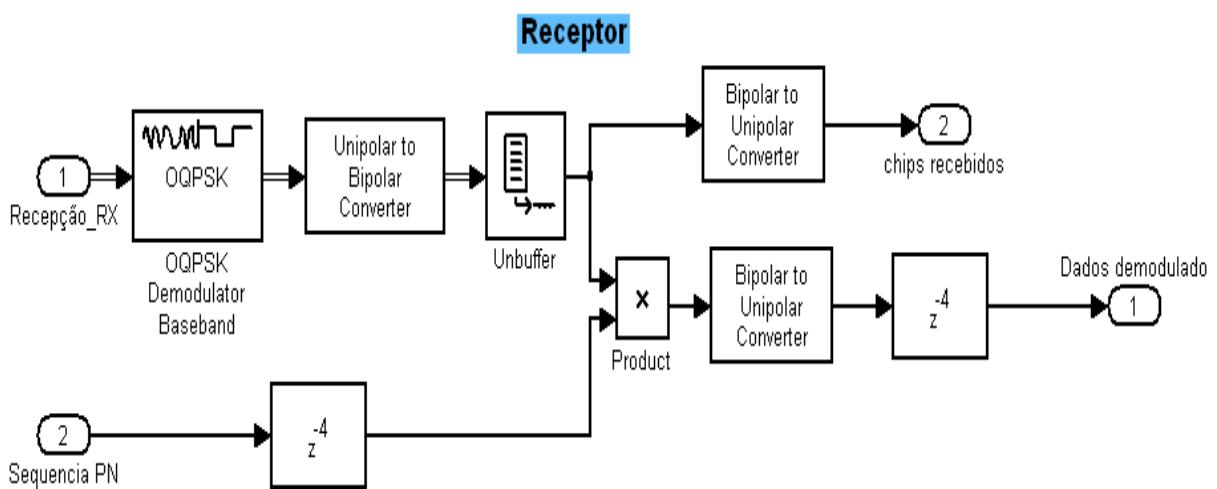
**Figura 32- Bits da CRC no receptor calculados sobre a sequência  $[b_{15}^r \ b_{14}^r \ \dots \ b_0^r + c_{15}^r \ c_{14}^r \ \dots \ c_0^r]$  no receptor.** Transmissão sobre um canal com 5% de probabilidade de erro de bit

## 5.4. Simulação Transceptor 2.4 GHz IEEE 802.15.4 com Simulink

No Simulink, foram modelados um transmissor e um receptor Zigbee, que, conjuntamente, compõem a funcionalidade de um transceptor Zigbee. Um transceptor é tanto capaz de transmitir dados como recebê-los.



**Figura 33 - Modelo em Simulink de um Transmissor Zigbee**



**Figura 34 - Modelo em Simulink de um Receptor Zigbee**

No transmissor, os bits que chegam pela porta 1 (*Dados de entrada*) são multiplicados pela sequência PN, codificando-os em chips. Este processo faz parte da técnica de espalhamento espectral.

Porém, antes desta multiplicação ocorrer, o transmissor converte os dados de entrada e a sequência PN de unipolar para bipolar, ou seja, de um esquema binário de 0/1 para um esquema -1/1. Esta mudança é necessária para que o resultado da multiplicação esteja de acordo com o que é descrito pela especificação IEEE 802.15.4. Se não houvesse a mudança de unipolar para bipolar, a multiplicação entre um bit nulo com uma sequência PN resultaria em chips também nulos, e este não é o propósito.

Depois da codificação de cada bit, o transmissor modula os chips sobre uma portadora no esquema O-QPSK. Com já explicado, nesta modulação cada par de chips é representado por uma das quatro fases do sinal modulado.

No transmissor, alguns blocos Simulink que merecem destaque são:

**Bloco Product:** bloco que faz a multiplicação entre dois escalares ou vetores. No transmissor, o bloco foi usado para codificar os bits e no receptor foi usado para decodificá-los. As operações possíveis serão:

$$1 * (-1) = -1$$

$$(-1) * 1 = -1$$

$$1 * 1 = 1$$

$$(-1) * (-1) = 1$$

**Bloco PN Sequence Generator:** Este bloco está disponível na biblioteca *Communications Blockset* e gera sequências PN. Um dos parâmetros deste bloco é o **Generator Polynomial** que deve ser preenchido com um vetor numérico e cujo primeiro elemento do vetor indica o comprimento da sequência PN. Já o parâmetro **Sample Time** determina a frequência em que os chips da sequência PN são gerados.

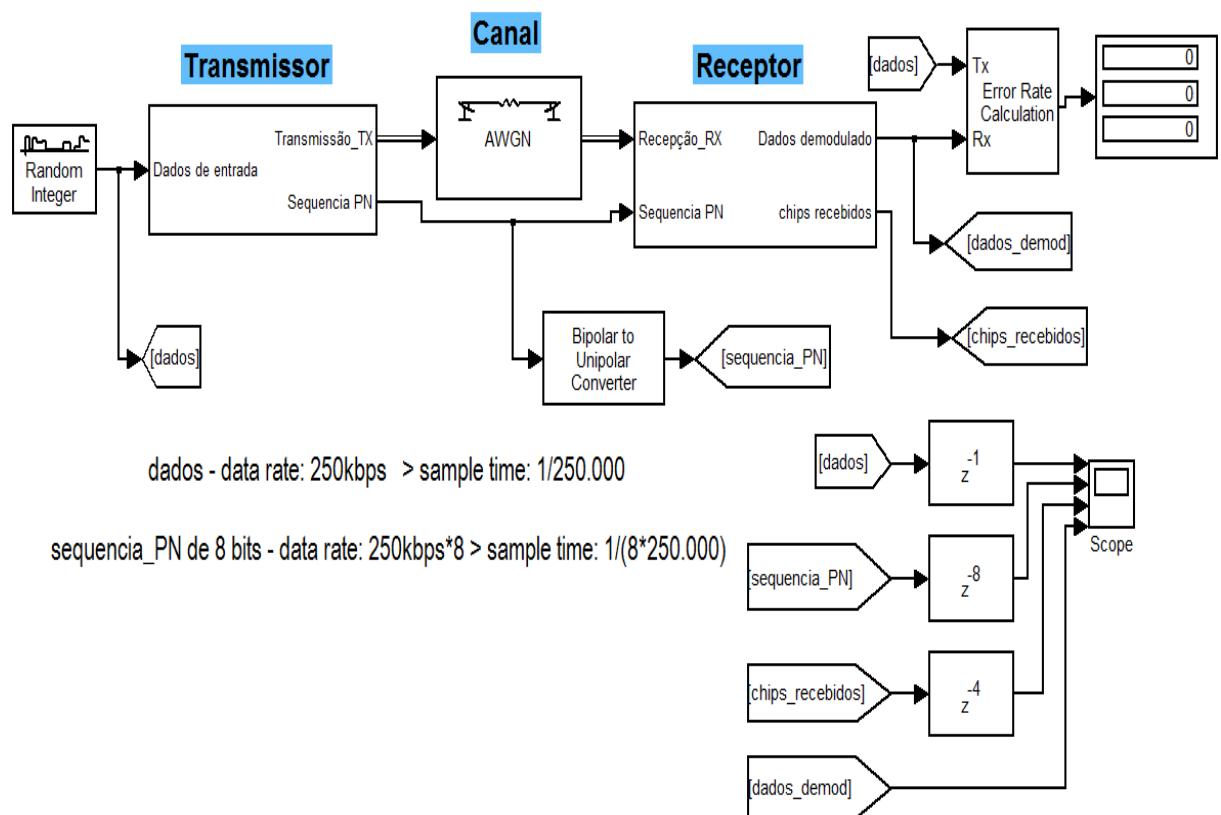
No receptor ocorre o processo inverso do que ocorre no transmissor. O sinal modulado é demodulado pelo bloco demodulador O-QPSK e os chips são decodificados formando os bits, novamente. Para que ocorra a decodificação correta no modelo de recep-

tor Simulink, os chips devem passar para bipolar novamente e, só então, serem multiplicados pela sequência PN.

No modelo do receptor vale explicar o uso de bloco de atraso  $z^{-n}$ :

**Blocoz<sup>-n</sup>:** Os chips e a sequência PN foram dessincronizados pelo bloco *buffer*, colocado no transmissor. Para uma decodificação correta,, a sequência PN teve que ser atrasada em 4 períodos de amostragem no modelo do receptor, sincronizando-a novamente com os chips. O valor do atraso provocado pelo *buffer* foi descoberto utilizando a função *rebuffer delay()*, disponível na biblioteca de funções MATLAB.

Desenvolvido os modelos de transmissor e receptor, simulou-se a transmissão de uma sequência binária sobre um canal AWGN (*Additive White Gaussian Noise*).



**Figura 35 - Simulação em Simulink de uma transmissão Zigbee sobre um canal AWGN**

Vale destacar a função de alguns blocos do sistema da Figura .

**Bloco Random Integer Generator:** bloco que gera números inteiros aleatórios entre 0 e  $M - 1$  em uma distribuição normal. O valor de  $M - 1$  é escolhido por meio do parâmetro **M-ary Number** que, para números binários, deve ser igual a 2. Outro parâmetro do bloco é o **Sample Time**, que determina a frequência de geração de números aleatórios.

**Bloco AWGN:** simula um canal não ideal com ruído aditivo gaussiano na razão  $Eb/No$ , determinado pelo parâmetro **Eb/No (dB)**, onde  $Eb$  é a energia de cada bit e  $No$  é a densidade de potência espectral do ruído.

**Bloco Error Rate Calculation:** este bloco determina a taxa de erros produzida pelo canal. O bloco tem duas portas de entrada:  $Tx$ , para o sinal transmitido e  $Rx$ , para o sinal recebido. A função do bloco é comparar os dois sinais e calcular a porcentagem de erros. A saída do bloco é um vetor com três elementos: *número total de erros*, *número total de bits transmitidos* e *taxa de erro*se a relação entre eles é:

$$\text{taxa de erros} = \frac{\text{número total de erros}}{\text{número total de bits transmitidos}}$$

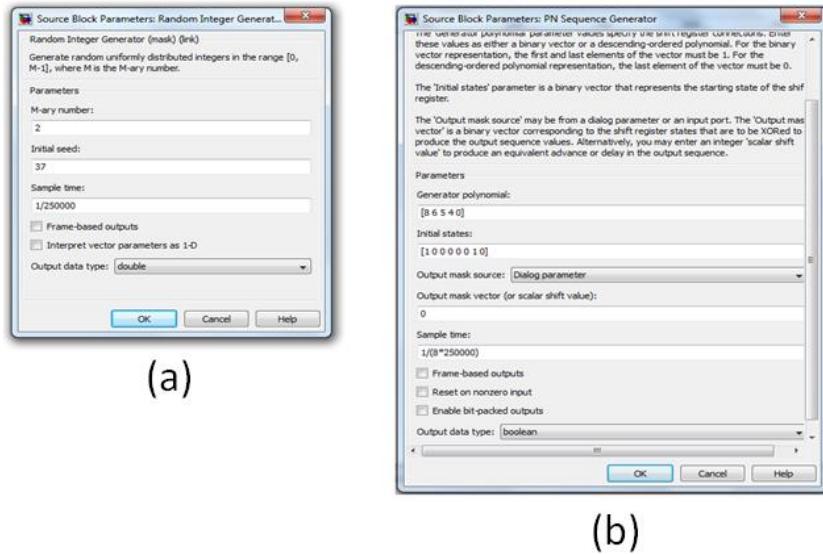
**Blocos  $z^{-n}$ :** antes de serem mostrados no osciloscópio, os sinais [*dados*], [*sequencia\_PN*] e [*chips\_recebidos*] foram atrasados em algumas amostras, para visualizá-los em sincronia como sinal [*dados\_demod*]. O sinal [*dados*] foi atrasado em uma amostra, considerando um período de amostragem de 4  $\mu\text{s}$ . Já o sinal [*sequencia\_PN*] foi atrasado em 8 amostras e o sinal [*chips\_recebidos*] foi atrasado em 4 amostras, considerando um período de amostragem de 0,5  $\mu\text{s}$ .

#### 5.4.1. Procedimento para Simulação

Com o auxilio do bloco *Error Rate Calculation*, foi analisada a influênciada relação sinal/ruído ( $Eb/No$ ) do canal sobre a taxa de erro de bits ( $Pb$ ).

O bloco *Random Integer Generator* foi ajustado para produzir uma sequência binária com **Sample Time** de 4  $\mu\text{s}$ , o que corresponde a 250 kbits/s. Em contrapartida, o bloco *PN Sequence Generator* foi ajustado para produzir sequências PN com **Sample Time** de 0,5  $\mu\text{s}$ , o que corresponde a 2.000 kchips/s. A especificação IEEE802.15.4 determina que o *chip-rate* seja quatro vezes maior que o *bit-rate* em transceptores de rádio em 2,4 GHz.

A Figura 36 mostra as configurações para os blocos *Random Integer Generator* e *PN Sequence Generator*.



**Figura 36 - Janela de Configuração dos blocos Random Integer Generator (a) e PN Sequence Generator (b)**

#### 5.4.2. Resultados

Para gerar o gráfico da Figura 37, rodou-se dez simulações, cada uma com um valor  $E_b/N_0$  diferente - todos os valores estão listados na Tabela 7. Apenas dez valores já foram suficientes para se verificar o comportamento da curva  $P_b \times E_b/N_0$ , mostrando que o modelo produzido em Simulink conseguiu aproximar-se bem do comportamento teórico.

**Tabela 7 - Valores levantados com a simulação para o gráfico  $P_b \times E_b/N_0$**

Eb/No [dB]	Taxa de erro [%]	Eb/No [dB]	Taxa de erro [%]
0	8,08	3	2,32
0,2	7,52	4	1,32
0,4	7,12	5	0,56
0,6	6,6	6	0,2
0,8	6	7	$6,40 \times 10^{-2}$
1	5,76	8	$2,4 \times 10^{-2}$
1,5	4,64	9	$4,80 \times 10^{-5}$
2	3,92	10	$1,20 \times 10^{-5}$

Na Figura 37, os pontos em verde no gráfico representam os valores da tabela, e a curva contínua em azul representa a curva teórica, que segue a equação:

$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)$$

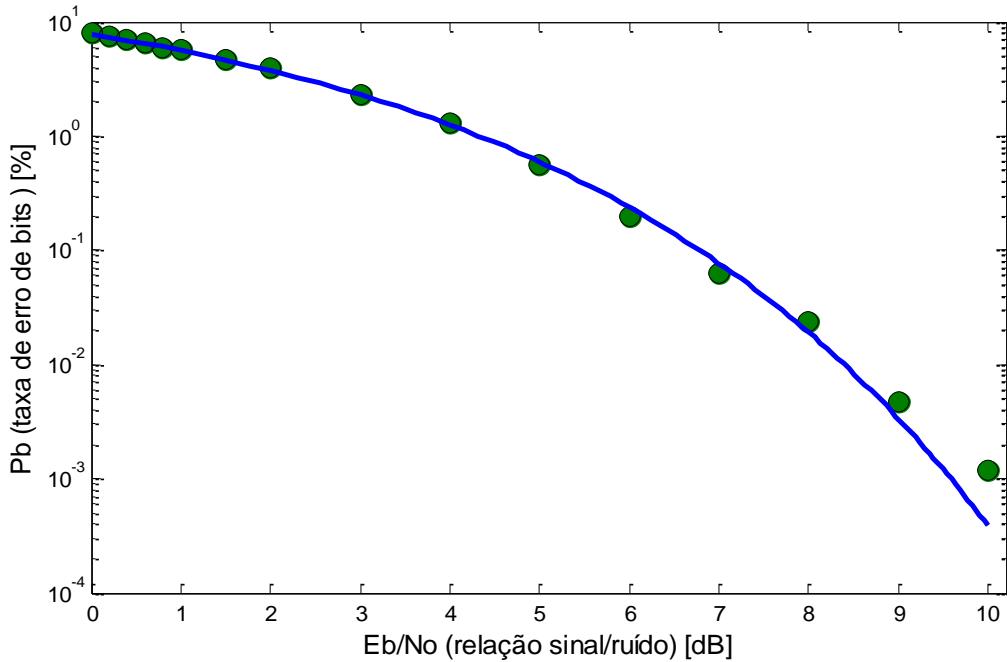


Figura 37 - Influência da relação sinal/ruído sobre a probabilidade de erro dos bits

Com o osciloscópio foram visualizados parte dos sinais binários para duas situações: uma para um canal ideal (Figura 39) e outra para um canal não-ideal com  $\frac{E_b}{N_0} = 0 \text{ dB}$  (Figura 40).

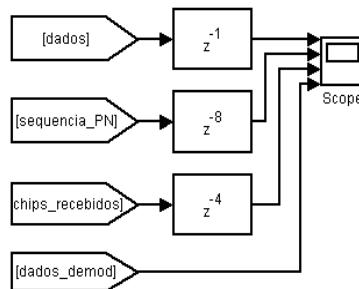
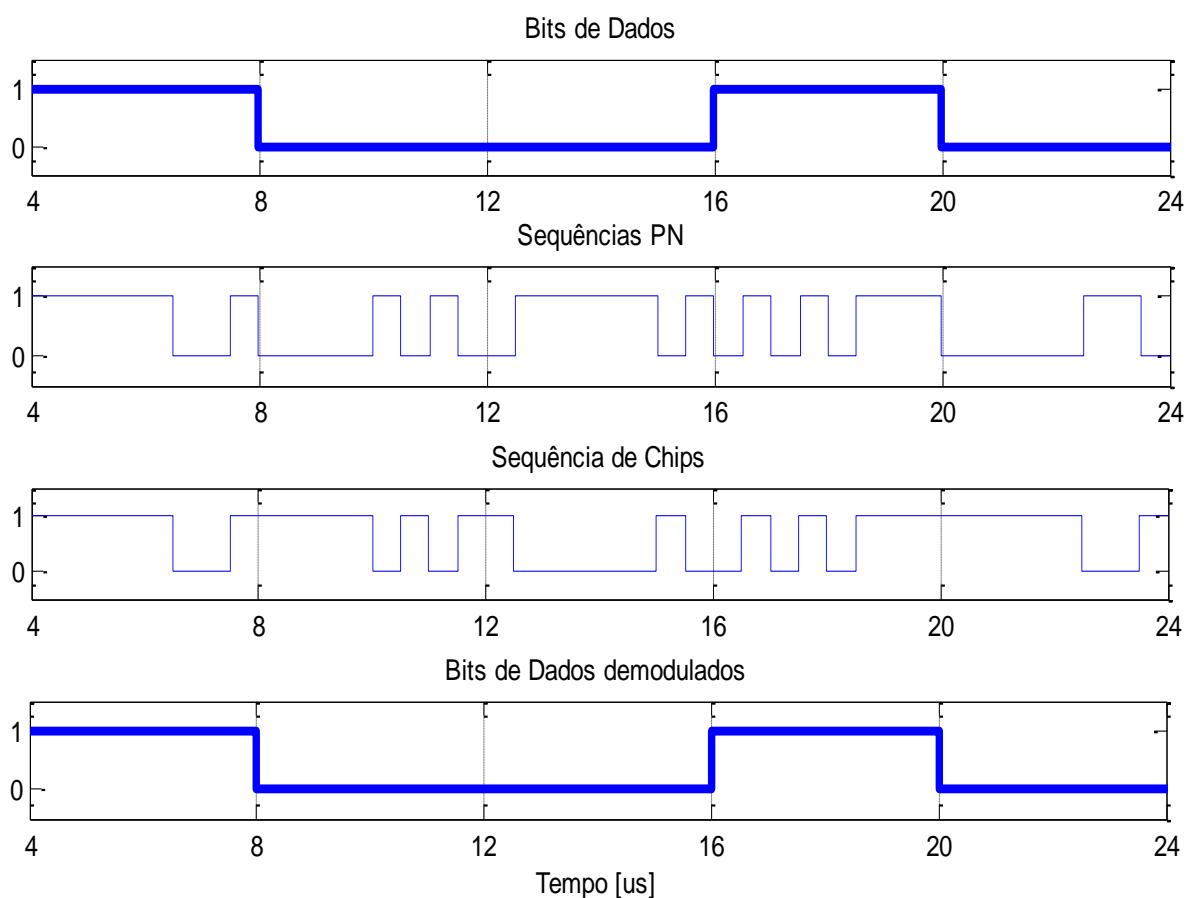


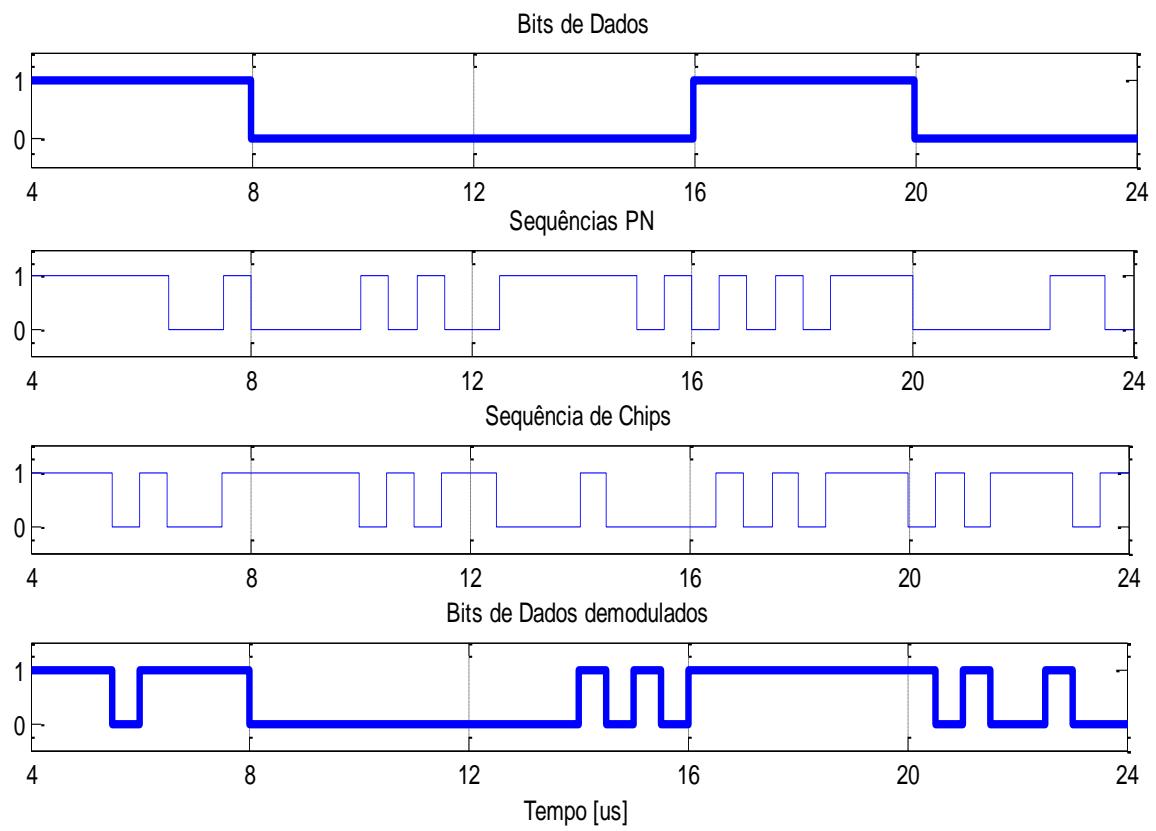
Figura 38 – Osciloscópio e sinais da simulação do transceptor

Na primeira situação (situação ideal) era esperado no Osciloscópio:

- Que os bits de dados transmitidos (primeiro sinal da Figura 39) fossem similares aos bits de dados demodulados no receptor (quarto sinal da Figura 39).
- Que para um bit 1, a sequência de *chips* recebidos (terceiro sinal da Figura 39) deveria ser igual sequência PN (segundo sinal da Figura 39). Enquanto que para um bit 0, a sequência de *chips* recebidos deveria ser complementar a sequência PN.



**Figura 39 - Sinais visualizados no Osciloscópio para um canal com uma relação sinal/ruído alta**



**Figura 40 - Sinais visualizados no Osciloscópio para um canal com uma relação sinal/ruído muito baixa**

## 6. CONCLUSÃO E TRABALHOS FUTUROS

Tecnologias de redes e dispositivos sem fio têm um futuro promissor em varias áreas, por serem visualmente limpos, pela mobilidade alcançada e por eliminarem os “desagradáveis” cabos. Uma destas áreas promissoras é a de sensoriamento remoto, em que o Zigbee e o IEEE 802.15.4 são protocolos em destaque, por cobrirem muitos dos requisitos exigidos: baixo consumo alcançado com eficiência energética; redes dinâmicas e flexíveis; e simplicidade protocolar em dispositivos com memória limitada.

Sabendo da relevância do Zigbee no cenário de padrões sem fio, este trabalho abordou uma variedade de temas relacionados à sua arquitetura protocolar, fazendo um retrato das funções e características principais do Zigbee.

Antes de descrever a pilha protocolar, o trabalho contextualizou o Zigbee dentro do universo de redes sem fio, fornecendo conceitos gerais sobre redes sem fio, redes *Wireless Sensor Network* (WSN) e aplicações para o Zigbee.

Ao final do texto teórico, foram executadas algumas simulações com o intuito de reforçar e consolidar conceitos ligados ao IEEE 802.15.4/Zigbee: modulação O-QPSK, espaçamento espectral com sequência PN, método CRC de verificação de erros e rede WSN na topologia em árvore.

Por fim, vale destacar que os autores utilizados muito acrescentaram para o desenvolvimento das simulações, bem como para a ampliação do conhecimento referente ao estudo e pesquisa realizados. Contudo, não se pode esquecer que a ciência é, também, uma construção histórica, e, por isso, pode tornar-se obsoleta em um curto período do tempo em decorrência das transformações tecnológicas e da produção da vida humana no mundo moderno. Neste sentido, do ponto de vista teórico, podemos considerar o aspecto temporal.

## **6.1. Considerações acerca de trabalhos futuros**

Trabalhos futuros recorreriam a ferramentas computacionais específicas para redes, que poderiam fornecer uma análise mais completa do desempenho e funcionamento de redes sem fio. Uma destas ferramentas é o OPNET Modeler, *software* pago com um ambiente completo para modelar e simular redes, dispositivos e protocolos. Outros simuladores seriam o NS-2 e o J-SIM. O primeiro é um dos simuladores mais usados, sendo implementado em C++ e isento de licença. O segundo é implementado totalmente no ambiente Java.

Ainda, sobre o método CRC e sobre o transceptor Zigbee ficam abertas outras análises/simulações possíveis, além daquelas efetuadas nas Subseções 5.3 e 5.4. Por exemplo, há a possibilidade de se comparar, em termos de eficiência, o método CRC com outros métodos de verificação de erros. Também é possível fazer comparações entre os transceptores de 2,4 GHz e o de 868/915 MHz, e suas respectivas modulações, O-QPSK e BPSK.

Fica, também, para o futuro analisar, no domínio da frequência, o espalhamento espectral gerado pelo transceptor 2,4GHz na codificação em chips.

## **Referências Bibliográficas**

BATISTA, O. **Cálculo de Cyclic Redundancy Check - CRC.** [S.I.]. 2009. Disponível em:  
<http://www.othonbatista.com/arquivos/redes/aulas/redes-othon-crc>.

GISLASON, D. **Zigbee Wireless Networking.** [S.I.]: Newnes.

HEILE, B. In: GISLASON, D. **Zigbee Wireless Networking.** [S.I.]: Newnes, p. 10.

IEEE. Part 15.4:Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks. IEEE. [S.I.]. 2003.

JIN-SHYAN, L.; YU-WEI, S.; CHUNG-CHOU, S. A comparative Study of Wireless Protocols:Bluetooth, UWB, Zigbee, Wi-Fi. **The 33rd Annual Conference of the IEEE Industrial Electronics Society**, Taipei, November 2007. 51.

KHAN, J.; MEHMET, Y.; KARAMI, F. **Performance Evaluation of a Wireless Body Area Sensor Network for Remote Pacient Monitoring.** 30th Annual International IEEE EMBS Conference. Vancouver: [s.n.]. 2008. p. 1266 - 1269.

KOUBAA, A.; ALVES, M.; TOVAR, E. **Modeling and Worst-Case Dimensioning of Cluster Wireless Sensor Networks.** Polytechnic Institute of Porto. [S.I.].

LE BOUDEC, J.-Y.; THIRAN, P. **Network Calculus:** A theory of Deterministic Queuing Systems for the Internet. [S.I.]: Springer, 2012. Disponível em:  
[http://ica1www.epfl.ch/PS\\_files/netCalBookv4.pdf](http://ica1www.epfl.ch/PS_files/netCalBookv4.pdf).

LEITURA Geral. **Leitura Geral.** Disponivel em: <<http://www.leiturageral.com/redes-sem-fio-wireless-wifi-acesso-velocidade-conceito-historico-tipos-de-rede-velocidades-indoor-outdoor-servicos/>>. Acesso em: Outubro 2012.

MUNK-STANDER, J.; SKOVGAARD, M.; NIELSEN, T. **Implementing a Zigbee Protocol Stack and Light Sensor in TinyOS.** University of Copenhagen. [S.I.]. 2005.

NIEUWENHUYSE, A.; ALVES, M.; KOUBÂA, A. **On the use of the Zigbee Protocols for Wireless Sensor Networks**. Polytechnic Institute of Porto, School of Engineering. [S.I.]. 2006.

NOORANI, N. Wireless Multi-Sensor Monitoring Utilizing IEEE 802.15.4 Communication Standards for Water Leakage Detection. California State University of Northridge. [S.I.].

RASIN, Z.; ABDULLAH, M. Water Quality Monitoring System Using Zigbee bases Wireless Sensor Network. **Internation Journal of Engineering & Technology**, v. Vol 9, n. Ed 10.

REYES, R.; MONJE, J. C.; SANTOS, M. E. Implementation of Zigbee-based and ISM-bases Wireless Sensor and Actuator Network with Throughput Power and Const Comparisons. Ateneo de Manila University. Filipinas.

SAKHARE, R.; DESMUKH, B. T. Electric Power Management using Zigbee Wireless Sensor Network. **Int**, v. Vol 4, n. Ed 1, p. 492 - 500, July 2012. ISSN 2231-1963.

THORNHILL, N.; IKRAM, W. **Wireless Communication in Process Automation: A Survey of Opportunities, Requirements, Concerns and Challenges**. Conferênciac Control 2010. Conventry: [s.n.]. 2010.

TJENSVOLD, J. M. **Comparison of the IEEE 802.11, 802.15.1, 802.15.4 and 802.15.6 wireless standards**. [S.I.]. 2007. Disponível em <http://janmagnet.files.wordpress.com/2008/07/comparison-ieee-802-standards.pdf>.

WHAT are Standards? **IEEE Standards**. Disponivel em:  
<https://standards.ieee.org/develop/overview.html>. Acesso em: Novembro 2012.

XIONG, F. Chapter 4 - Phase Shift Keying. In: \_\_\_\_\_ **Digital Modulation Techniques**. 2<sup>a</sup>. ed. [S.I.]: Artech House, 2006. p. 135 - 182.

YEN, L.-H.; LAW, Y. W.; PALANISWAMI, M. Risk-Aware Distributed Beacon Sheding for Tree-Based Zigbee Wireless Networks. [S.I.].

ZIGBEE ALLIANCE. **Zigbee Specification**. [S.I.]. 2008. Disponível em:  
<http://www.zigbee.org/Standards/Downloads.aspx>.

## Bibliografias Consultadas

BRALEY, R.; GIFFORD, I.; HEILE, R. Wireless Personal Area Networks: An Overview of the IEEE P802.15 Working Group. **Mobile Computing and Communications Review**, v. vol 4, n. ed 1.

C. SELADA, R. S. **Redes Wireless de Banda Larga**. Universidade de Trás-os-Montes e Alto Douro. [S.I.]. 2008.

CARDEIRA, C.; COLOMBO, A.; SCHOOOP, R. **Wireless solutions for automation requirements**. IDMEC, Instituto Superior Técnico e Schneider Electric. [S.I.].

CHADRAMOULI, V. **A Detailed Study on Wireless LAN Technologies**. University of Texas. Arlington.

DAINTREE NETWORKS. **Understanding 802.15.4 and Zigbee Networking**. [S.I.]. 2005. Disponível em:

[http://www.daintree.net/downloads/whitepapers/understanding\\_sensor\\_networking\\_whitepaper.pdf](http://www.daintree.net/downloads/whitepapers/understanding_sensor_networking_whitepaper.pdf).

DAINTREE NETWORKS. **Getting Started with Zigbee and IEEE 802.15.4**. [S.I.]. 2008.

Disponível em: [http://www.daintree.net/downloads/whitepapers/zigbee\\_primer.pdf](http://www.daintree.net/downloads/whitepapers/zigbee_primer.pdf).

ERGEN, S. C. **Zigbee/IEEE 802.15.4 Summary**. Berkeley. [S.I.]. 2004. Disponível em: <http://staff.ustc.edu.cn/~ustcsse/papers/SR10.ZigBee.pdf>.

IEEE. **Overview and Guide to the IEEE802.15.4 LMSC**. [S.I.]. 2012. Disponível em <http://www.ieee802.org/orientation.shtml>.

IEEE 802.15 Working Group for Wireless Personal Area Networks. **ieee802**. Disponível em: <<http://www.ieee802.org/15/about.html>>.

IEEE. **IEEE-SA Study Group Guidelines**. [S.I.]. Disponível em <http://standards.ieee.org/develop/corpchan/studygrp.pdf>.

Introducing Zigbee RF4CE. DaintreeNetwork. [S.I.]. Disponível em <http://www.daintree.net/downloads/whitepapers/zigbee-rf4ce-intro.pdf>.

JENNIC. **Zigbee Stack User Guide**. [S.I.]. 2008. disponível na versão 1.6 em [www.jennic.org/support/user\\_guides/](http://www.jennic.org/support/user_guides/).

KINNEY, P. Zigbee Technology: Wireless Control that Simply Works. [S.I.]. 2003.

KO, L.-C.; FANG, H.-W. **Design and Implementation of IEEE 802.15.4 Beacon-enabled Network Devices**. Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops. [S.I.]: [s.n.]. 2006.

LASKOSKI, G. T.; MARCONDES, M.; SZEREMETA, O. **Tópicos e Comunicações: Modulação Digital**. UTFPR. Curitiba. 2006. Disponível em:  
<http://pessoal.utfpr.edu.br/gustavothl/modulacao/doc1.pdf>.

MAUDONET, R. L. **Estudo comparativo entre redes sem fio e redes cabeadas**. Faculdade de Jaguariúna. Jaguariúna. 2007.

MENNENGA, M. Designing Wireless Sensor Technology. Ventocom. Dresden.

P. D. **Protocolos de Comunicação - AULA2**. [S.I.]. Disponível em <http://xa.yimg.com/kq/groups/22932771/1111132330/name/REDES-aula02.pdf>.

RABBIT. **An introduction to Zigbee**. [S.I.]. 2008. Disponível em:  
[ftp://ftp1.digi.com/support/documentation/0190162\\_c.pdf](ftp://ftp1.digi.com/support/documentation/0190162_c.pdf).

SCHWEITZER, C. et al. Tecnologias de rede sem fio:WPANs, WLANs e WMNs. Desafios de segurança, vulnerabilidade e soluções. [S.I.].

TJENSVOLD, J. M. **Comparison of the IEEE 802.11, 802.15.1, 802.15.4 and 802.15.6 wireless standards**. [S.I.]. 2007. Disponível em <http://janmagnet.files.wordpress.com/2008/07/comparison-ieee-802-standards.pdf>.

UMAR, A. **7 Wireless Personal Area Networks - Bluetooth, UWB and Sensor Networks**. [S.I.]. Disponível em <http://harikrishnanck.com/wp-content/uploads/2011/02/c07-WPANs.pdf>.

VISHWAKARMA, D. IEEE 802.15.4 and Zigbee: A Conceptual Study. International Journal of Advanced Research in Computer and Communication Engineering IJARCCE, Vadodara, v. 1, n. 7.

**WILAN. Spread Spectrum Wireless Technology white paper.** [S.I.]. Disponível em [http://old.optivera.ru/products/Wilan/whitepaper\\_mcdsss.pdf](http://old.optivera.ru/products/Wilan/whitepaper_mcdsss.pdf).

**WILLIG, A.; MATHEUS, K.; WOLISZ, A. Wireless Technology in Industrial Network.** [S.I.].

**ZIGBEE ALLIANCE. Zigbee Specification.** [S.I.]. 2008. Disponível em: <http://www.zigbee.org/Standards/Downloads.aspx>.

**ZIGBEE ALLIANCE. Zigbee Home Automation Application Profile.** [S.I.]. 2010. Disponível em: <http://www.zigbee.org/Standards/Downloads.aspx>.

**ZIGBEE ALLIANCE. Zigbee Cluster Library Specification.** [S.I.]. 2012. Disponível em: <http://www.zigbee.org/Standards/Downloads.aspx>.

**ZIGBEE ALLIANCE. Zigbee Wireless Sensor Applications for Health, Wellness and Fitness.** [S.I.]. Disponível em: [www.zigbee.org/imwp/download.asp?contentid=15585](http://www.zigbee.org/imwp/download.asp?contentid=15585).

## Apêndice

### [A]Classificação de redes sem fio

As redes podem ser classificadas de acordo com o seu tamanho físico (extensão). O tamanho e alcance da rede influência em outras características. Tecnologias para redes de pequena dimensão apresentam uma máxima taxa de transmissão, potência, custo, proteção dos dados, tamanho do pacote de dados, geralmente menores que as redes de longo alcance.

Quanto à extensão espacial, as redes sem fio são caracterizadas como redes pessoais (WPAN - *Wireless Personal Area Network*), redes de locais (WLAN - *Wireless Local Area Network*), rede metropolitanas (WMAN - *Wireless Metropolitan Area Network*), redes amplas (WWAN – *Wireless Wide Area Network*), sendo que o Wi-Fi é classificado com uma WLAN e o Bluetooth e Zigbee como uma WPAN.

A Figura 41 sumariza os alcances e taxas de transmissões típicas para diversos padrões sem fio.

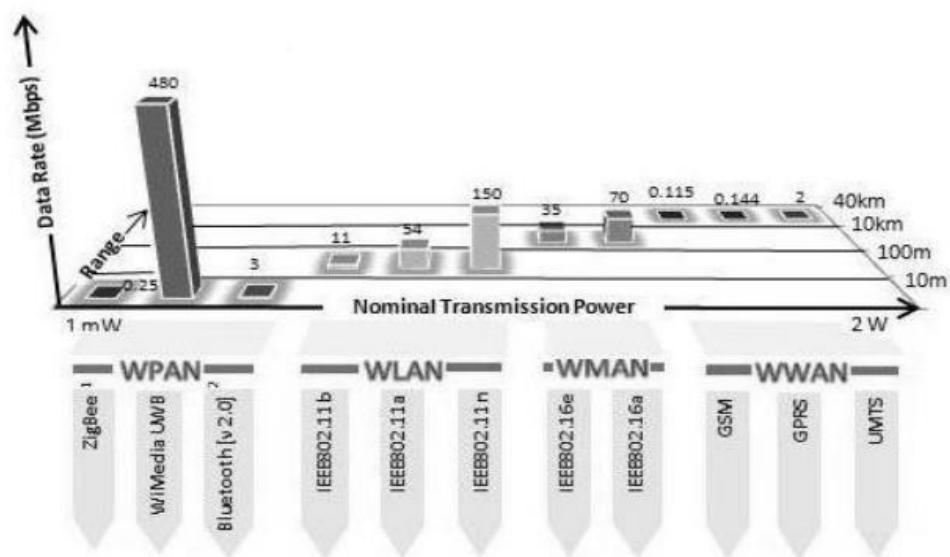


Figura 41 - Alcance típico das redes WPAN, WLAN, WMAN, WWAN – retirada da fonte (THORNHILL e IKRAM, 2010)

## [B] PRIMITIVAS IEEE 802.15.4

### [B1] Primitivas PHY

---

**Tabela 8 - Primitivas da camada Física**

Primitiva de Dados	.request	.confirm	.indication
PD-DATA	✓	✓	✓
<b>Primitiva de gerenciamento</b>			
PLME-CCA	✓	✓	
PLME-ED	✓	✓	
PLME-GET	✓	✓	
PLME-SET-TRX-STATE	✓	✓	
PLME-SET	✓	✓	

### [B2] Primitivas MAC

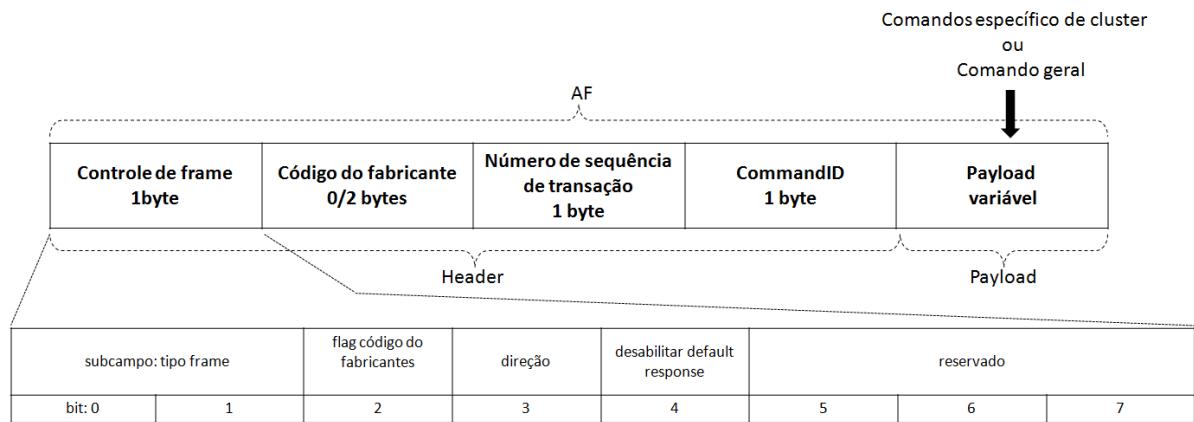
---

**Tabela 9 - Primitivas da camada MAC**

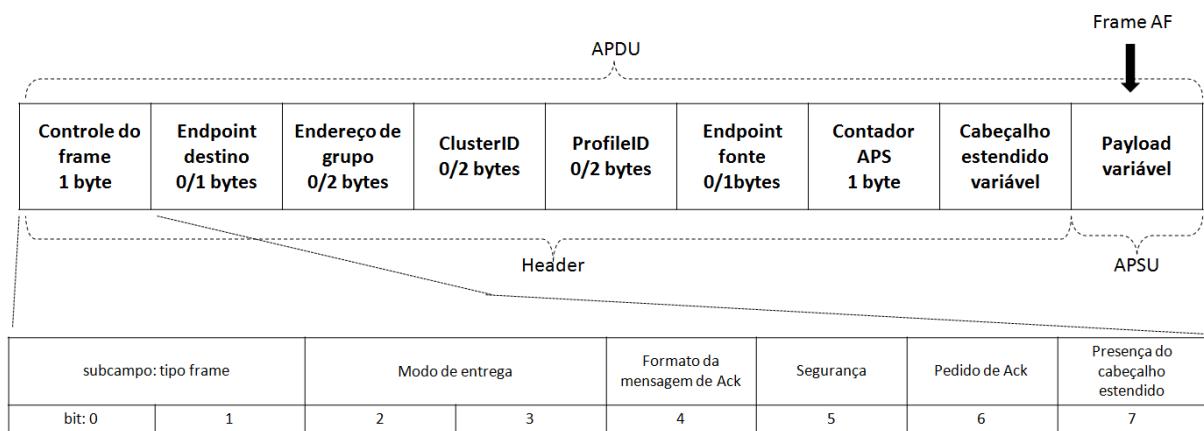
Primitiva de Dados	.request	.confirm	.indication	.response
MCPS-DATA	✓	✓	✓	
MCPS-PURGE	✓	✓		
<b>Primitiva de gerenciamento</b>				
MLME-ASSOCIATE	✓	✓	✓	✓
MLME-DISASSOCIATE	✓	✓	✓	
MLME-BEACON-NOTIFY			✓	
MLME-GET	✓	✓		
MLME-GTS	✓	✓	✓	
MLME-ORPHAN			✓	✓
MLME-RESET	✓	✓		
MLME-RX-ENABLE	✓	✓		
MLME-SCAN	✓	✓		
MLME-COMM-STATUS			✓	
MLME-SET	✓	✓		
MLME-START	✓	✓		
MLME-SYNC	✓			
MLME-SYNC-LOSS			✓	
MLME-POLL	✓	✓		

## [C] Camada MAC, NWK e APL: frame e comandos

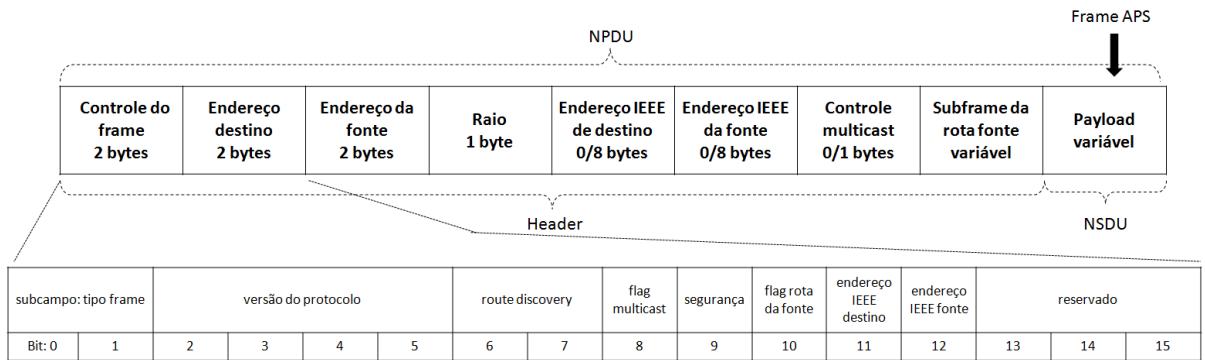
### [C1] Frames



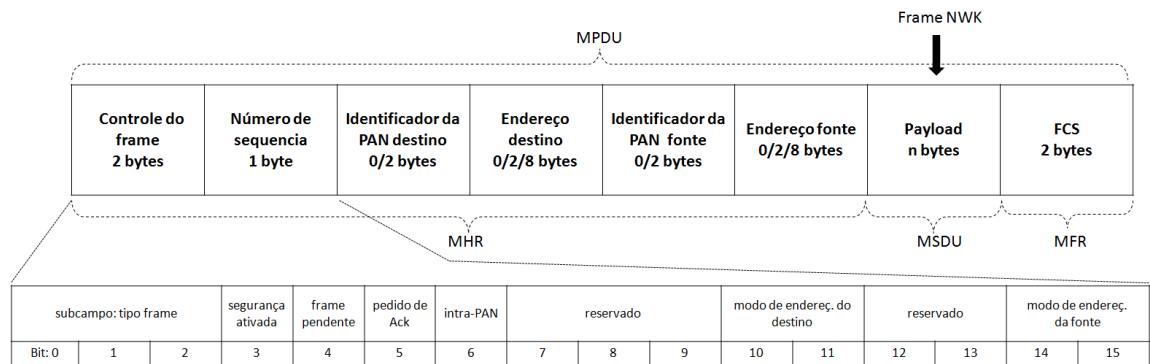
**Figura 42 - Frame AF da subcamada de framework de aplicacção e campo controle do frame**



**Figura 43- Frame da subcamada de suporte a aplicação**



**Figura 44–Frame da camada NWK e campo controle do frame**



**Figura 45–Frame da camada MAC e campo controle de frame**

- Tipo de frames MAC

**Frame Beacon:** é o sinalizador de início do período de transmissão em uma rede em modo *beacon*. O frame é usado para sincronizar a rede e para a associação de novos dispositivos à rede. Ele é apenas emitido pelos coordenadores da PAN e contém informações sobre a estrutura do superquadro, lista de endereços, endereços pendentes, dentre outras. O frame é originado na própria MAC e depois repassado para a PHY.

**Frame de Dados:** destinado a enviar os dados formados nas camadas de cima. A MAC apenas adicionará o cabeçalho e o rodapé ao campo de dados (e.g., MSDU) e o repassará para a PHY.

**Frame de Reconhecimento:** é empregado para reduzir os erros de transmissão, já que os *frames* recebidos corretamente serão sempre seguidos com uma confirmação da recepção. Se não confirmados, os dados serão enviados novamente. Tem-se, então, um esquema de envio-resposta, na qual é certificado se um pacote foi recebido ou se houve algum problema. O *frame* de reconhecimento também se origina na MAC.

**Frame de Comando MAC:** lida com a transferência de dados entre as camadas MAC. Pedido de associação, resposta da associação, notificação da dissociação, pedido de dados são alguns exemplos de comandos MAC que podem ser transferidos para outras entidades MAC.

- Tipo de frames NWK

**Frame de Dados:** é o *frame* NWK, cujo *payload* comporta o *frame* APDU da subcamada de suporte a aplicação.

**Frame de Comando NWK:** é o *frame* NWK, cujo *payload* engloba o número identificador do comando NWK e o próprio comando NWK. Tem-se uma lista de dez comandos NWK diferentes.

## [C2] Comandos

---

**Tabela 10 - Comandos MAC**

Valor do comando	Comando NWK	Reduced Function Device	
		Trans	Recep
0x01	pedido de associação	✓	
0x02	resposta da associação		✓
0x03	notificação da dissociação		✓
0x04	pedido de dados	✓	
0x05	notificação de conflito de PAN ID	✓	
0x06	notificação órfão	✓	
0x07	pedido de beacon		
0x08	realinhamento do coordenador		✓
0x09	pedido de GTS		
0xa – 0xff	reservado		

**Tabela 11 - Comandos NWK**

Valor do Comando	Comando NWK
0x01	pedido da rota
0x02	resposta da rota
0x03	status da rede
0x04	sair da rede
0x05	gravar rota
0x06	pedido de reingresso
0x07	resposta de reingresso
0x08	status do link
0x09	relatório da rede
0x0a	atualização da rede
0x0b - 0xff	reservado

## [D] Fluxograma de primitivas entre as camadas

Nas subseções abaixo será mostrado as trocas de primitivas entre as camadas , quando ocorre a formação de uma e a entrada de novos dispositivos na rede.

### [D1] Fluxograma para formação de uma nova rede Zigbee.

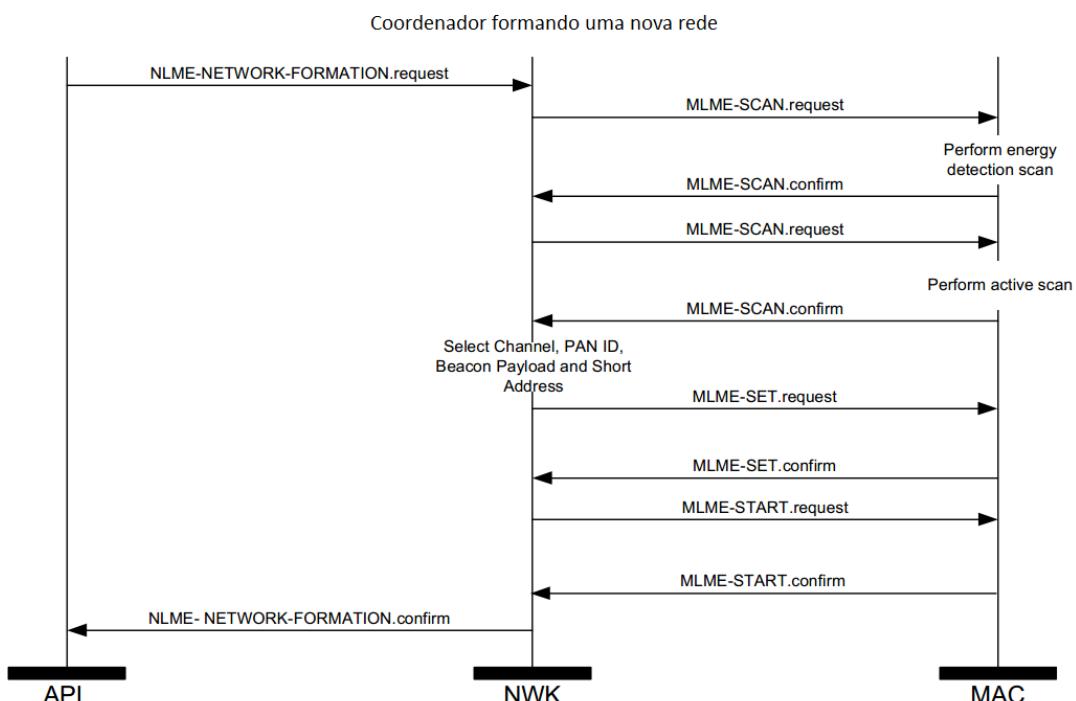
Quando a aplicação rodando no coordenador quer começar uma rede, ela expõe a primitiva **NLME-NETWORK-FORMATION.request** para a NWK. É responsabilidade da camada de rede formar a rede apropriadamente.

A NWK emite a primitiva **MLME-SCAN.request** para medir a energia do canal (o parâmetro *ScanType* da MLME-SCAN é definido em 0x00 - ED).

Após a detecção de energia, outra primitiva **MLME-SCAN.request** será emitida, mas selecionada para uma varredura ativa (o parâmetro *ScanType* da MLME-SCAN é definido em 0x01 – active scan). A PAN ID será escolhida aleatoriamente entre um valor menor ou igual a 0x3fff .O valor identificador PAN ID é armazenado no atributo *macPANID*, através da primitiva **MLME-SET.request**.

O coordenador também precisa de um endereço NWK, e como padrão será 0x0000para o endereço de rede do coordenador.

O último passo para iniciar a rede é a emissão da primitiva ***MLME-START.request*** para PHY com informações de configuração da rede, PAN ID e canal escotilhos. Por fim, a camada NWK também avisa a ZDO (dentro da APL) se foi possível formar a rede com uma primitiva ***NLME-NETWORK-FORMATION.confirm***.



**Figura 46 - Primitivas de formação para rede – retirada da fonte (ZIGBEE ALLIANCE, 2008)**

#### [D2] Primitivas para associação de um novo dispositivo na rede Zigbee

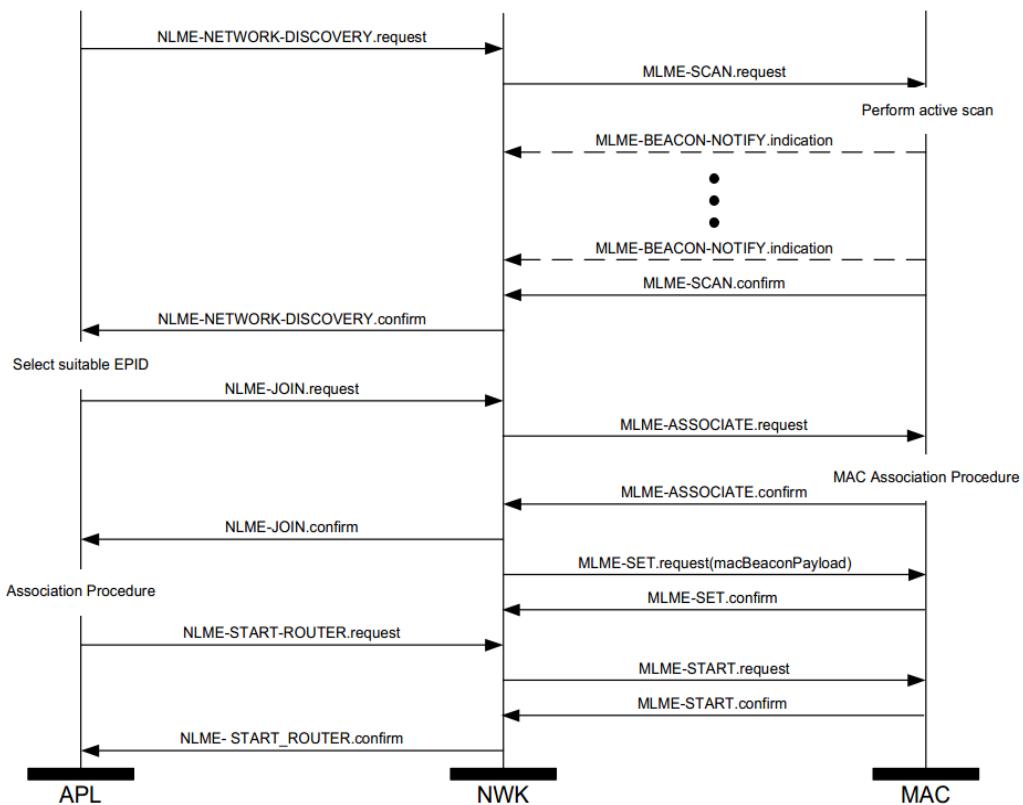
O processo para um nó se juntar a uma rede comum pede de descoberta de rede ***NLME-NETWORK-DISCOVERY.request*** pela APL para a NWK.

A camada NWK então fará vários pedidos de varredura ativa ou passiva com a primitiva ***MLME-SCAN.request***. Todos os *beacons* recebidos, de um coordenador ou de roteadores, são notificados pela primitiva ***MLME-BEACON-NOTIFY.indication***. O *beacon* contém descritores PAN, e outras informações que descrevem a rede PAN.

A camada NWK armazena informações sobre os roteadores vizinhos, que enviaram um *beacon*, na tabela de vizinhança e também fornece para a ZDO, a lista de redes ativas ao redor.

Assim que a ZDO escolhe uma PAN, ela envia a primitiva **NLME-JOIN.request**, na qual a camada NWK procurará, na sua tabela de vizinhança, por um dispositivo parente apropriado. A tabela de vizinhança contém um campo que setado em 1, indica que o dispositivo é um potencial pai. Com a escolha de um pai, a **MLME-ASSOCIATE.request** requerida para a MAC, o envio de um pedido de associação (comando MAC com identificador 0x01) para o pretendente a pai.

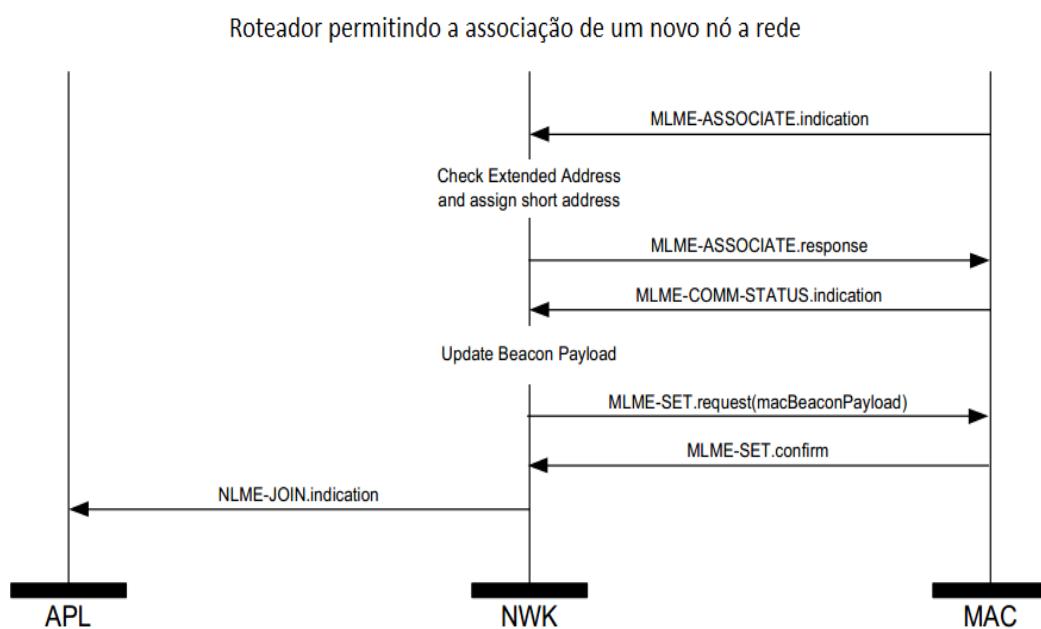
As outras primitivas são **MLME-SET.request**, **MLME-START.request** e **NLME-START-ROUTER.request** e finalizam o processo de associação a uma nova rede.



**Figura 47 - Primitivas para entrada de um dispositivo na rede – retirada da fonte (ZIGBEE ALLIANCE, 2008)**

No lado do roteador-pai, a primitiva **MLME-ASSOCIATE.response** contém a decisão da camada NWK ao pedido de associação. A primitiva contém o endereço NWK atribuído ao nó associado, o status da tentativa de associação, e outros dados. Uma resposta a associação (comando MAC identificador 0x02) é enviado pelo roteador ao nó, com o endereço NWK dele, caso a associação seja bem-sucedida.

A primitiva **NLME-JOIN.indication** avisa a aplicação na ZDO que um novo dispositivo filho foi aceito.



**Figura 48 - Primitivas no roteador/coordenador sobre a entrada de novos dispositivos na rede – retirada da fonte (ZIGBEE ALLIANCE, 2008)**

## [E] Constantes, Atributos e Tabelas Zigbee.

Algumas Constantes	Valor	Descrição	Alguns Atributos	Descrição
aMaxPHYPacketSize	127	máximo tamanho em bytes do PSDU da PHY	phyCurrentChannel	canal usado para transmissão/recepção
aTurnaroundTime	12	máximo período em símbolos para a transição RX-TX ou TX-RX	phyChannelSupported	canais disponíveis para uso
aBaseSlotDuration	60	duração em símbolos de um slot superquadro	phyTransmitPower	potência nominal de transmissão em dBm
aBaseSuperquadroDuration	960	mínima duração em símbolos de um superquadro		
aExtendedAddress	Variável	endereço IEEE de 64 bits		
aMaxBE	5	máximo valor do expoente do cálculo de backoff		
aMaxBeaconOverhead	75	máximo tamanho do payload do frame beacon		
aMaxBeaconPayloadLength	52	máximo tamanho do payload do frame beacon		
aMaxFrameOverhead	25	máximo tamanho de um cabeçalho MAC se segurança		
aMaxMACFrameSize	102	máximo tamanho do payload MAC		
aMinCAPLength	440	mínima duração em símbolos do CAP de um superquadro		
aNumSuperquadroSlots	16	número de slots contidos em um superquadro		
aUnitBackoffPeriod	20	período em símbolos de uma unidade backoff		
nwkcCoordinatorCapable	-	se setado em 0x00 indica se o dispositivo é capaz de se tornar um coordenador da rede	nwkMaxChildren	máximo número de filhos associados permitido.
nwkcProtocolVersion	2	versão do protocolo Zigbee instalado no dispositivo	nwkMaxDepth	a profundidade que um dispositivo pode ter na rede
nwkcMinHeaderOverhead	8	mínimo tamanho do cabeçalho NWK	nwkMaxRouters	quantidade de roteadores que um dispositivo pode ter como filho
nwkcMACFrameOverhead	11	tamanho do cabeçalho MAC usado pela camada NWK	nwkNeighborTable	vizinhos do dispositivo na tabela neighborable
apscMinHeaderOverhead	12	mínimo tamanho de bytes do cabeçalho APS	nwkUseTreeRouting	indica se a camada de NWK deve ser capaz de fazer roteamento em árvore
apscMaxFrameRetries	3	máximo numero de novas tentativas após falha na transmissão	nwkNetworkAddress	endereço NWK do dispositivo
			nwkStackProfile	tipo de pilha Zigbee usada: Zigbee ou Zigbee PRO
			nwkGroupIDTable	tabela com grupos de dispositivos
			nwkExtendedPANID	endereço estendido da rede
			nwkAddrAlloc	indica o método de alocação de endereço : estocástico ou Cskip
			nwkRouteTable	tabela de rotas em malha
			apsBindingTable	entradas na tabela Binding

apsGroupTable	tabela com grupos de endpoints
---------------	--------------------------------

parâmetros da nwkNeighborTable	Descrição
Endereço IEEE	endereço IEEE de 64 bits do vizinho
Endereço NWK	endereço de rede de 16 bits do vizinho
Tipo de dispositivo	indica se o vizinho é um coordenador, roteador ou end-device
RxOnWhenidle	indica se o vizinho está em modo de recepção durante períodos ociosos.
Relação	o que o vizinho é do dispositivo no nível hierárquico: pai, filho, ou irmão.
Falha na transmissão	indica se a última transmissão foi bem sucedida ou não.
LQI	estimativa da qualidade de link entre o dispositivo e seu vizinho.
A tabela também pode ter outros parâmetros optativos	

parâmetros da nwkRouteTable	Descrição
Endereço NWK	parâmetro que contém o endereço NWK do dispositivo no fim da rota
Status da Rota	indica se a rota está ativa ou inativa, e outros status possíveis
Flag route cache	informação se o destinatário não armazena "sourceroutes"
Muitos-para-um	indica se o destinatário é um concentrador que é capaz de gerenciar a recepção de mensagens de vários dispositivos, ao mesmo tempo
Exigência de registro de rota	indica se um comando para gravar rota deve ser enviado antes de um pacote de dados
Flag GroupID	parâmetro que aponta que o destino da mensagem é para um grupo de dispositivo. O endereço NWK terá um valor de GroupID
Endereço do próximo vizinho	Endereço NWK do próximo dispositivo dentro da rota a receber a mensagem.

## [F] Equivalência entre símbolos e segundos

Todos os intervalos de tempo na especificação IEEE 802.15.4 são descritos na unidade símbolo. A equivalência em segundos da unidade símbolo varia de acordo com a faixa de frequência empregada.

- Em 2,4 GHz com a modulação O-QPSK, 1 símbolo dura 16 µs.
- Em 915 MHz com a modulação BPSK, 1 símbolo dura 25 µs.
- Em 868 MHz com a modulação BPSK, 1 símbolo dura 50 µs.

Assim, os intervalos de tempo definidos na especificação são relativos, variando conforme a banda do transceptor.