

UNIVERSIDADE ESTADUAL PAULISTA “JULIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA ELÉTRICA
CAMPUS DE ILHA SOLTEIRA

LUCAS ARRUDA RAMALHO

**MODELAGEM IEEE 1815 DNP3 EM VHDL E ANÁLISE
DE COMUNICAÇÃO SG VIA IEEE 802.15.4 E IEEE
802.11**

Ilha Solteira - SP
2014

LUCAS ARRUDA RAMALHO

**MODELAGEM IEEE 1815 DNP3 EM VHDL E ANÁLISE
DE COMUNICAÇÃO SG VIA IEEE 802.15.4 E IEEE
802.11**

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

Área de Conhecimento: Automação.

Prof. Dr. AÍLTON AKIRA SHINODA

Orientador

Prof. Dr. VALTEMIR EMERÊNCIO DO NASCIMENTO

Co-Orientador

Ilha Solteira - SP
2014

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

R165m Ramalho, Lucas Arruda.
Modelagem IEEE1815 DNP3 em VHDL e análise de comunicação SGs via IEEE 802.15.4 e IEEE 802.11 / Lucas Arruda Ramalho. -- Ilha Solteira: [s.n.], 2014
79 f. : il.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2014

Orientador: Aílton Akira Shinoda
Co-orientador: Valtemir Emerêncio do Nascimento
Inclui bibliografia

1. Smart grid. 2. IEEE 1815 DNP3. 3. IEEE 802.15.4. 4. IEEE 802.11. 5. VHDL.

CERTIFICADO DE APROVAÇÃO

TÍTULO: MODELAGEM IEEE 1815 DNP3 EM VHDL E ANÁLISE DE COMUNICAÇÃO SG VIA IEEE 802.15.4 E IEEE 802.11

AUTOR: LUCAS ARRUDA RAMALHO

ORIENTADOR: Prof. Dr. AILTON AKIRA SHINODA

CO-ORIENTADOR: Prof. Dr. VALTEMIR EMERENCIO DO NASCIMENTO

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


Prof. Dr. AILTON AKIRA SHINODA

Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira


Prof. Dr. ALEXANDRE CESAR RODRIGUES DA SILVA

Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira


Prof. Dr. LEOPOLDO RIDEKI YOSHIOKA

Departamento de Engenharia de Sistemas Eletrônicos / Universidade de São Paulo

Data da realização: 06 de fevereiro de 2014.

Dedico aos meus queridos pais, Luiz Leôncio Ramalho da Silva e Zildineti Alves de Arruda, que me educaram, me deram estrutura para tal feito.

AGRADECIMENTOS

A Deus por ter aberto as oportunidades e dado condições favoráveis de saúde e paz para o cumprimento deste desafio.

Aos meus pais e demais familiares por terem dado educação e valores que me possibilitassem ter sucesso em meus estudos e carreira profissional.

Ao professor Dr. Aílton Akira Shinoda pela paciência e orientação que começou na graduação, se estendeu ao mestrado e continuará no doutorado.

Ao professor Dr. Valtemir Emerêncio do Nascimento, em nome do qual cumprimento os professores do Grupo de Pesquisa em Redes e Segurança (GPRS), que tiveram papel importante em minha formação como pesquisador.

Ao professor Dionízio, em nome de todos os docentes, discentes e técnico-administrativos do programa de pós-graduação em Engenharia Elétrica, pela ajuda, companheirismo e excelente prestação de serviços oferecida. Com certeza, o meu período vivenciado neste ambiente de pesquisa trouxe-me muitas alegrias.

Ao colega de trabalho rotineiro Alcides Ortega, pelo companheirismo e discussões que ajudaram no entendimento e domínio do ambiente *Smart Grid*.

Aos amigos Nelcileno, Juliana, Rothschild e Luiz pela companhia e paciência no convívio na República formada em duas sagas durante o Mestrado.

Aos colegas e amigos Cássia, Douglas, Gilvani, Mário, Reginaldo, Edgar, Marcos e Odilon e demais mato-grossenses pela união para cumprir os créditos através das disciplinas em Ilha Solteira, e para suportar a distância de nossa Cuiabá.

Aos demais amigos por torcerem pelo sucesso em minha carreira profissional.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo auxílio financeiro durante o mestrado.

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

“Para evoluç o de um grupo, cada indiv duo dele tem de fazer o melhor para si mesmo e para o grupo.”

John Nash

"Deus nos concede, a cada dia, uma p gina de vida nova no livro do tempo. Aquilo que colocarmos nela, corre por nossa conta."

Chico Xavier

Resumo

O conceito *Smart Grid* (SG) estabelece que, em adição ao fluxo de energia, a concessionária tenha um fluxo de dados de duas vias em todos os setores da rede até os consumidores. Através da Tecnologia da Informação e Comunicação (TIC), torna-se possível o sensoriamento de toda a grade de energia, a solução de falhas mais ágil e eficiente, e a teleproteção e gerenciamento dos ativos das concessionárias. Para que isso seja garantido, requisitos como segurança, confiabilidade e baixa latência são essenciais. Considerando que existem diversas aplicações SG, adaptar um sistema de comunicação, entre os medidores inteligentes (*Smart Meters*), para cada tipo de ambiente se torna complexo. Além disso, o estudo de comunicação do fluxo SG se torna oneroso na montagem de cenários reais, devido ao alto custo na aquisição de *Smart Meters*. A fim de possibilitar estudos simulados da comunicação SG de baixo custo, neste trabalho foi realizada a modelagem do protocolo IEEE 1815 DNP3, validou seu funcionamento através de modelagem das integrações IEEE 1815/802.15.4 e IEEE 1815/802.11b. A validação e análise foram realizadas pela modelagem e simulação de ambos os protocolos, em *Very High Speed Integrated Circuit* (VHSIC) *Hardware Description Language* (HDL), e efetuando medições de atraso na troca de mensagens DNP3 enviadas através dos protocolos sem fio integrados. Os requisitos de latência de teleproteção, em seu contexto geral, apontam a interface IEEE 802.15.4 como inadequada para esta aplicação por apresentar alta latência na presença de cenários de disputa de acesso, baixa vazão de dados e baixa resiliência de segurança implementada. Situação que não ocorre para a interface IEEE 802.11, que se torna factível pois apresenta latência, vazão de dados e robustez de segurança compatíveis com tal aplicação crítica SG.

Palavras- Chave: IEEE 1815 DNP3. IEEE 802.15.4. IEEE 802.11. VHDL. *Smart Grid*.

Abstract

The Smart Grid (SG) concept provides that, in addition to the power flow, the utilities has a two-way data flow in all sectors of the grid to consumers. Through Information and Communication Technology (ICT), it becomes possible the sensing of entire power grid, the more efficient and faster solution failures, and the teleprotection and management of the assets of utilities. To ensure this, requirements such as security, reliability and low latency are essential. Considering that there are several applications SG, to adapt a communication system between Smart Meters for each type of environment becomes complex. Moreover, the communication study of data flow SG becomes expensive in assembling real scenarios, due to the high cost of acquisition of Smart Meters. To enable SG communication studies simulated in low cost, in this work was constituted the modeling of IEEE 1815 DNP3 protocol, validated its operation through modeling of IEEE 1815 / 802.15.4 and IEEE 1815 / 802.11b integrations. The validation and analysis were performed by modeling and simulation of both protocols in Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (HDL), and performing measurements of delay in the exchange of DNP3 messages sent via integrated wireless protocols. The teleprotection latency requirements in its general context, indicate the IEEE 802.15.4 interface as inadequate for this application due to high latency scenarios in the presence of contention access, low data throughput and low resilience of improved security. Situation that does not occur for the IEEE 802.11 interface, which becomes feasible because it presents latency, data throughput and robustness safety critical application compatible with that SG.

Keywords: IEEE 1815 DNP3. IEEE 802.15.4. IEEE 802.11. VHDL. Smart Grid.

Lista de Figuras

Figura 1 -	Domínios e Níveis de Abstração HDL (Diagrama Y de Gajski e Kuhn).	22
Figura 2 -	Abordagens de desenvolvimento hierárquico de projeto HDL em <i>bottom-up</i> e <i>top-down</i> .	23
Figura 3 -	Abordagem Geral adotada.	24
Figura 4 -	Conversão de máquinas de estado do framework para código VHDL.	25
Figura 5 -	Arquitetura Clássica de uma rede <i>Smart Grid</i> .	26
Figura 6 -	Fluxo de dados <i>Smart Grid</i> .	27
Figura 7 -	Teleproteção de Subestação.	30
Figura 8 -	Pilha de protocolo DNP3.	31
Figura 9 -	Máquinas de estado camadas API dos dispositivos DNP3 <i>Master</i> e <i>Outstation</i> .	33
Figura 10 -	Classes de Comandos API do protocolo DNP3.	35
Figura 11 -	Máquinas de estado camadas TF e DLL.	38
Figura 12 -	Exemplo de Comunicação entre dispositivos DNP3.	40
Figura 13 -	Topologia de entidades do dispositivo DNP3 Master em VHDL.	41
Figura 14 -	Representação de comunicação simulada através de mudanças de estado DNP3.	42
Figura 15 -	Comparativo entre tecnologias <i>wireless</i> .	45
Figura 16 -	Algoritmo <i>slotted</i> CSMA/CA.	48
Figura 17 -	Máquinas de estado da camada MAC IEEE 802.11.	49
Figura 18 -	Estação 802.11b modelada.	51
Figura 19 -	Pilha IEEE 1815/802.11b.	52
Figura 20 -	Algoritmo <i>unslotted</i> CSMA/CA.	54
Figura 21 -	Comunicação protocolo IEEE 802.15.4 com disputa de acesso ao meio.	55
Figura 22 -	Máquinas de estado da camada MAC IEEE 802.15.4.	56
Figura 23 -	Estação IEEE 802.15.4 modelada.	57
Figura 24 -	Encapsulamento DNP3 / IEEE 802.15.4.	58
Figura 25 -	Simulação Cenário 1 – IEEE 1815/802.11b.	60
Figura 26 -	Simulação Cenário 2 – IEEE 1815/802.11b.	60
Figura 27 -	Simulação Cenário 3 – IEEE 1815/802.11b.	61
Figura 28 -	Simulação Cenário 4 – IEEE 1815/802.15.4.	65
Figura 29 -	Simulação Cenário 5 – IEEE 1815/802.15.4.	66
Figura 30 -	Simulação Cenário 6 – IEEE 1815/802.15.4.	67

Lista de Tabelas

Tabela 1 -	Requisitos de Latência para Aplicações SG.	28
Tabela 2 -	Cabeçalho Camada API do protocolo DNP3.	32
Tabela 3 -	Controle de Aplicação.	32
Tabela 4 -	Transições de Estados camada DNP3 API <i>Master</i> .	33
Tabela 5 -	Transições de Estados camada DNP3 API <i>Outstation</i> .	34
Tabela 6 -	Cabeçalho Função Transporte.	36
Tabela 7 -	Cabeçalho de Enlace do protocolo DNP3.	37
Tabela 8 -	Controle de Enlace.	37
Tabela 9 -	Código de Função de Enlace.	37
Tabela 10 -	Transições de Estados camada TF.	39
Tabela 11 -	Transições de Estados camada DLL.	39
Tabela 12 -	Descrição de eventos da comunicação DNP3 entre <i>Master</i> e <i>Outstation</i> .	43
Tabela 13 -	Comparativo entre interfaces de comunicação.	45
Tabela 14 -	Quadro MAC do protocolo IEEE 802.11.	46
Tabela 15 -	Descrição de bits <i>Frame Control</i> da camada MAC 802.11.	47
Tabela 16 -	Descrição dos campos <i>Type</i> e <i>Subtype</i> da camada MAC 802.11 que serão utilizados no trabalho.	47
Tabela 17 -	Transições de Estados camada MAC IEEE 802.11.	50
Tabela 18 -	Quadro PHY do protocolo IEEE 802.11.	51
Tabela 19 -	Quadro MAC do protocolo IEEE 802.15.4.	53
Tabela 20 -	Descrição de bits <i>Frame Control</i> da camada MAC 802.15.4.	53
Tabela 21 -	Transições de Estados camada MAC IEEE 802.15.4.	56
Tabela 22 -	Quadro PHY do protocolo IEEE 802.15.4.	57
Tabela 23 -	Descrição de eventos da simulação do Cenário 1.	62
Tabela 24 -	Descrição de eventos da simulação do Cenário 2.	63
Tabela 25 -	Descrição de eventos da simulação do Cenário 3.	63
Tabela 26 -	Descrição de eventos da simulação do Cenário 4.	68
Tabela 27 -	Descrição de eventos da simulação do Cenário 5.	69
Tabela 28 -	Descrição de eventos da simulação do Cenário 6.	70
Tabela 29 -	Resultados Obtidos nas simulações e comparação com requisitos de latência.	71

Lista de Siglas

ACK	<i>Acknowledgment</i>
AMI	<i>Advanced Metering Infrastructure</i>
AP	<i>Access Point</i>
APH	<i>API Header</i>
API	<i>Application Program Interface</i>
BAN	<i>Build Area Network</i>
BSSID	<i>Basic Service Set Identification</i>
CAP	<i>Contention Active Period</i>
CCA	<i>Clear Channel Assessment</i>
CON	<i>Confirm</i>
CRC	<i>Cyclic Redundant Check</i>
CSMA/CA	<i>Carrier Sense Multiple Access with Collision Avoidance</i>
CTS	<i>Clear to Send</i>
DA	<i>Destination Address</i>
DCF	<i>Distributed Control Function</i>
DFC	<i>Data flow Control</i>
DIFS	<i>Distributed Inter Frame Space</i>
DIR	<i>Direction</i>
DLH	<i>DLL Header</i>
DLL	<i>Data Link Layer</i>
DNP3	<i>Distributed Network Protocol version 3</i>
DS	<i>Distribution System</i>
E/S	<i>Entrada ou Saída</i>
FBC	<i>Frame Count Bit</i>
FCS	<i>Frame Check Sequence</i>
FCV	<i>Frame Count Value</i>
FIFO	<i>First In First Out</i>
FIN	<i>Final</i>
FIR	<i>First</i>
FPGA	<i>Field-Programmable Gate Array</i>
HAN	<i>Home Area Network</i>
HDL	<i>Hardware Description Language</i>

IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IPH	<i>IP Header</i>
IPv4	<i>Internet Protocol version 4</i>
LIFS	<i>Long Inter Frame Space</i>
LLC	<i>Logical Link Control</i>
LLCH	<i>LLC Header</i>
MAC	<i>Media Access Control</i>
MACH	<i>MAC Header</i>
NAN	<i>Neighborhood Area Network</i>
NAV	<i>Network Allocation Vector</i>
PHY	<i>Physical</i>
PHYH	<i>PHY Header</i>
PMU	<i>Phasor Measurement Unit</i>
PRM	<i>Primary Message</i>
QoS	<i>Quality of Service</i>
RA	<i>Receiver Address</i>
RF	<i>Radio Frequency</i>
RFC	<i>Request For Comments</i>
RTL	<i>Register Transfer Level</i>
RTS	<i>Request to Send</i>
SA	<i>Source Address</i>
SAN	<i>Substation Area Network</i>
SCADA	<i>Supervisory Control And Data Acquisition</i>
SEQ	<i>Sequence</i>
SFD	<i>Start Frame Delimiter</i>
SG	<i>Smart Grid</i>
SIFS	<i>Short Inter Frame Space</i>
SM	<i>State Machines</i>
TA	<i>Transmitter Address</i>
TCP	<i>Transport Communication Protocol</i>
TCPH	<i>TCP Header</i>
TF	<i>Transport Function</i>
TFH	<i>TF Header</i>

UNS	<i>Unsolicited</i>
VHDL	<i>VHSIC Hardware Description Language</i>
VHSIC	<i>Very High Speed Integrated Circuit</i>
VPN -	<i>Virtual Private Network</i>
WACS	<i>Wide Area Control Schemes</i>
WAMS	<i>Wide Area Measurement Schemes</i>
WAN	<i>Wide Area Network</i>
WAPS	<i>Wide Area Protection Schemes</i>
WiFi	<i>Wireless Fidelity</i>
WPAN	<i>Wireless Personal Area Networks</i>

SUMÁRIO

1	INTRODUÇÃO	16
	1.1 Trabalhos Relacionados	18
	1.2 Contribuição do trabalho	20
	1.3 Organização do trabalho	21
2	LINGUAGEM VHDL	22
3	SMART GRID	26
	3.1 Aplicação de teleproteção de subestações	29
	3.2 Protocolo de Rede Distribuída DNP3	31
4	INTERFACE DE COMUNICAÇÃO	45
	4.1 Modelagem IEEE 802.11b	46
	4.2 Integração IEEE 1815/802.11b	51
	4.3 Modelagem IEEE 802.15.4	52
	4.4 Integração IEEE 1815/802.15.4	58
5	RESULTADOS E SIMULAÇÕES	59
	5.1 Simulações dos cenários IEEE 1815/802.11b	59
	5.2 Simulações dos cenários IEEE 1815/802.15.4	64
	5.3 Análise de Resultados	71
6	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	73
	REFERÊNCIAS	74
	APÊNDICE A – Artigos publicados	78
	APÊNDICE B – Códigos VHDL	79

1 INTRODUÇÃO

O sistema convencional de energia depende de estruturas de transmissão e distribuição para fornecer a potência gerada ao consumidor final (HASSAN; RADMAN, 2010). Nesse sistema, as concessionárias têm controle e detecção de falhas limitados às subestações de transmissão e, portanto, não alcançando o sistema de distribuição como um todo (FANG et al., 2011).

A fim de melhorar o sistema elétrico, para efetuar um controle mais amplo e uso da energia otimizado, vários países têm investido bilhões de dólares implementando projetos piloto *Smart Grids* (SG) e aplicando seus conceitos em cidades inteligentes (NEICHIN; CHENG, 2010).

O conceito SG estabelece que, em adição ao fluxo de energia, a concessionária tenha um fluxo de dados de duas vias em todos os setores da rede até os consumidores (FANG et al., 2012). Através da Tecnologia da Informação e Comunicação (TIC), torna-se possível o sensoriamento de toda a grade de energia, a solução de falhas mais ágil e eficiente, além da teleproteção e gerenciamento dos ativos das concessionárias.

Para que isso seja garantido, requisitos como segurança, confiabilidade e baixa latência são essenciais, como demonstra a norma do *Institute of Electrical and Electronics Engineers* (IEEE, 2011a; LU et al., 2013; SADEGHI et al., 2012). Considerando que existem diversas aplicações SG, adaptar um sistema de comunicação, entre os medidores inteligentes (*Smart Meters*), para cada tipo de ambiente se torna complexo.

Entre os vários protocolos desenvolvidos para interligação de *Smart Meters*, destaca-se o *Distributed Network Protocol version 3* (DNP3) (IEEE, 2012). Este protocolo foi desenvolvido entre 1992 e 1994, com a pretensão de ser o primeiro protocolo verdadeiramente aberto no ambiente de transmissão de dados industriais. A ideia era construir uma compilação das melhores soluções proprietárias existentes na época, adicionando conceitos como disponibilidade, escalabilidade, confiabilidade, segurança e interoperabilidade de equipamentos.

A disponibilidade de acesso e escalabilidade de uma rede de dispositivos são garantidas quando o mesmo é capaz de manter seu desempenho e funcionamento de forma semelhante com o passar de um grande período de tempo ou aumento do número de dispositivos.

Essa mesma rede deve fazer uso de ferramentas que garantam que os dados transmitidos não sejam alterados mesmo em ambientes hostis à comunicação de dados. Exemplos de ambiente hostis estão em interferência de sinais eletromagnéticos em indústrias ou subestações, ou ainda em ataques digitais para acesso não autorizado às informações que trafegam pela internet. A confiabilidade e segurança da comunicação podem ser aumentadas com ferramentas como os sistemas de verificação de erro e criptografia, respectivamente. Além dos conceitos descritos anteriormente, a interoperabilidade de equipamentos é importante para que a comunicação possa ocorrer entre dispositivos de fabricantes diferentes.

O protocolo DNP3 já é implementado no monitoramento de subestações e permite a transmissão e recebimento de diferentes tipos de dados. Os dados de pontos flutuantes e de natureza analógica, como tensão ou corrente, podem ser convertidos para forma digital de 8 ou 16 bits, dependendo da resolução que se deseja. De mesma forma, os dados digitais, como comandos de abertura ou fechamento de chaves, podem ser recebidos pelo dispositivo através de portas de entrada digitais.

Apesar do protocolo DNP3 ser inicialmente desenvolvido para operar em interface serial, existe uma tendência, na literatura (ARAVINTHAN et al., 2011; FADLULLAH et al., 2011; PARIKH et al., 2010; SHAH et al., 2008), no ambiente comercial e industrial (TELESCADA, 2010; KALKI TECH, 2012; COX et al., 2012), para a utilização deste em outras interfaces, como IEEE 802.3, IEEE 802.11, IEEE 802.15.4, Zigbee (IEEE, 2003; IEEE, 2007; IEEE, 2011b; ZIGBEE ALLIANCE, 2007), entre outras.

Essa tendência é necessária para viabilizar as comunicações dos dados dentro do conceito SG. Isso porque, mesmo que o DNP3 possibilite a transmissão de diferentes tipos de dados, as prioridades de latência de cada um também são diferentes e, portanto, algumas não poderiam ser transmitidas em um mesmo enlace serial de baixa/média capacidade. Ou seja, em uma subestação, por exemplo, dados de teleproteção de eventos críticos ou que informem o mal funcionamento de um dispositivo devem ser transmitidos com latência inferior aos dados de monitoramento rotineiros e não críticos como temperatura. Então, para satisfazer requisitos diferentes de aplicações SG diversas, tem-se assim a possível formação de uma rede de comunicação heterogênea, ou seja, cada aplicação ou setor SG fará uso de diferentes tecnologias de transmissão para efetuar o fluxo de dados.

Considerando que esse fluxo seja realizado por diversas tecnologias e interfaces de comunicação, a latência do sistema de dados será o resultado da soma das latências de cada enlace. Sendo assim, cada enlace deve ter a menor latência possível, a fim de diminuir o

atraso das tomadas de decisões da concessionária de energia com as possíveis falhas do sistema.

Os estudos e análises de sistemas de comunicação *Smart Grid* têm a escolha de implementar cenários onerosos de comunicação entre *Smart Meters*, simular tais cenários via *software*, ou emular o funcionamento desses dispositivos via *hardware*. A fim de possibilitar estudos simulados da comunicação SG, neste trabalho foi realizada a modelagem do protocolo IEEE 1815 DNP3, validou seu funcionamento através de modelagem das integrações IEEE 1815/ 802.15.4 e IEEE 1815/ 802.11b. A validação e análise foram realizadas pela modelagem e simulação de ambos os protocolos, em *Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (HDL)* (IEEE, 2002), e efetuando medições de atraso na troca de mensagens DNP3 enviadas através dos protocolos sem fio integrados.

Foram realizadas simulações de seis cenários para medição de latência, considerando as disputas de acesso ao meio previstas pelas camadas MAC dos protocolos IEEE 802.11 e IEEE 802.15.4.

A modelagem VHDL diminui o tempo de resposta de protocolos devido ao processamento paralelo das mensagens (WANG et al., 2011), a montagem e segmentação rápida dos quadros de comunicação foi possibilitada devido a essa característica. As transmissões modeladas consideraram a vazão das interfaces sem fio em questão.

Para validar a modelagem e simulações realizadas, foi utilizada a norma IEEE 2030, que define o atraso mínimo para cada área de rede de comunicação. Os resultados das simulações mostram que as integrações IEEE 1815/ 802.15.4 e IEEE 1815/ 802.11b propostas são promissoras e estão dentro dos requisitos de latência para aplicações SG.

1.1 Trabalhos relacionados

A utilização de diversas interfaces de comunicação em *Smart Grid* é analisada por vários trabalhos presentes na literatura, e descritos nesta seção.

Os requisitos mínimos de escalabilidade, confiabilidade, segurança e disponibilidade, em uma comunicação sem fio para aplicações SG, são abordados em (ARAVINTHAN et al., 2011). Segundo o trabalho, a escolha da tecnologia de rede sem fio que deverá operar em cada etapa do fluxo de dados, dependerá do suporte que a mesma possui para atender os requisitos de cada aplicação SG, como baixo custo e alta confiabilidade dos dados.

Comparações da utilização de várias tecnologias de rede sem fio foram realizadas em (FADLULLAH et al., 2011; PARIKH et al., 2010). Os autores (FADLULLAH et al., 2011) afirmam que a confiabilidade das tecnologias empregadas na conexão necessita ser alta mesmo para ambientes *Home Area Network* (HAN). Por isso, o protocolo Zigbee pode possibilitar essas conexões, pois além de confiabilidade, este possui desempenho e autonomia de energia superiores a outros protocolos comumente utilizados em *Wireless Personal Area Networks* (WPAN). Analisando as características das redes Zigbee, como alcance, autonomia energética e baixo consumo, o trabalho (PARIKH et al., 2010) sugere que a tecnologia deva ser utilizada no controle direto da carga de energia nas residências dos consumidores. A única ressalva feita está em relação ao futuro dessas redes, que pode mostrar desafios relacionados à coexistência de outros protocolos que utilizam a faixa de frequência livre 2,4 GHz, causando interferência de sinais.

O contexto apresentado pelos três trabalhos anteriores foi essencialmente conclusivo através de conceitos e características teóricas já presentes na literatura. Essas abordagens definem que as integrações analisadas por este trabalho, IEEE 1815/802.15.4 e IEEE 1815/802.11, tenham aplicações SG definidas como HAN, *Build Area Network* (BAN) ou *Neighborhood Area Network* (NAN) para a interface IEEE 802.15.4 (WPAN) e teleproteção de subestações para interface IEEE 802.11 (WLAN).

Os cenários apresentados por (ARAVINTHAN et al., 2011; FADLULLAH et al., 2011; PARIKH et al., 2010; SHAH et al., 2008; ULLO et al., 2010; PFITSCHER et al., 2011), além de soluções comerciais existentes (TELESCADA, 2010; KALKI TECH, 2012; COX et al., 2012) fazem uso de inúmeras interfaces sem fio nas aplicações de rede inteligente de energia.

A modelagem em VHDL de dispositivos de comunicação já foi realizada na literatura e auxiliaram na modelagem das camadas MAC e PHY do protocolo IEEE 802.15.4 e IEEE 802.11.

Em (PANG et al., 2007) foi realizada uma modelagem do protocolo IEEE 802.11b em VHDL. O modelo empregou 3 símbolos ou entidades para representar o comportamento da camada *Physical* (PHY) e *Media Access Control* (MAC), com ênfase na modelagem do protocolo da função *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) com modo de acesso *Distributed Control Function* (DCF). Para simular o comportamento, os autores não efetuaram transmissões de dados através de montagem e segmentação de quadros, tal como é descrito via especificação de protocolo. Ou seja, o modelo de comunicação era através de sinais lógicos binários que representavam comportamento do protocolo. As

simulações mostraram o funcionamento adequado das sinalizações *Clear Channel Assessment* (CCA), *Acknowledgment* (ACK), *Clear to Send* (CTS), *Request to Send* (RTS) e *Backoff*.

No trabalho de (WANG, 2007) foi realizada uma análise de viabilidade de sensoriamento e controle sem fio para aplicações de estruturas civis. Para efetuar uma sincronização entre dispositivos, o autor efetuou a modelagem dos mesmos em máquinas de estado. O trabalho afirma que a abordagem do conceito de máquinas de estado em sistemas de comunicação, facilita a sincronização entre dispositivos de protocolos diferentes. Isso ocorre na modelagem do fluxo de ações, que facilita o comportamento entre estados de um dispositivo e outro.

Uma modelagem mais completa foi realizada em (BHAVIKATTI et al., 2001). Os autores descrevem cinco entidades, modelando o comportamento do gerenciador de filas *First In First Out* (FIFO), verificação de erro *Cyclic Redundant Check* 32 bits (CRC 32 bits) e montagem e segmentação de quadros de acordo com protocolo IEEE 802.11 *Wireless Fidelity* (WiFi). Através das simulações e análise do comportamento modelado, os autores afirmam que, após a síntese do *hardware*, o mesmo teria capacidade de comunicação com outros dispositivos comerciais WiFi.

O trabalho realizado por (BIANCHI, 2000) modela todas as características da camada MAC DCF e PHY 802.11b em 11 entidades. Além da simulação do comportamento houve a síntese e implementação no dispositivo *Field-Programmable Gate Array* (FPGA). Como resultado, o dispositivo FPGA teve sucesso em sua comunicação com uma rede WLAN.

A integração do DNP3 com as interfaces de comunicação, através de modelagem VHDL e posterior síntese em FPGA, podem contribuir como legado de baixo custo, via simulação ou emulação, quando comparado à aquisição de medidores inteligentes comerciais, permitindo avaliações de desempenho do ambiente de comunicação SG.

1.2 Contribuição do trabalho

Este trabalho busca contribuir nos seguintes tópicos:

- **Modelagem em VHDL de dispositivos DNP3:** O estudo de comunicação do fluxo SG se torna oneroso na montagem de cenários reais, devido ao alto custo na aquisição de *Smart Meter*. Sendo assim, a modelagem DNP3 realizada pode simular o comportamento de dispositivos DNP3, tornando viáveis o estudo da comunicação em baixo custo;

- **Modelagem de integrações IEEE 1815/ 802.11b e IEEE 1815/ 802.15.4:** O fluxo de dados SG será realizado por uma rede distribuída que utilize diversos tipos de interfaces de comunicação. Através das integrações modeladas em VHDL, tem-se um início de legado de integrações que possibilitarão emulações de uma comunicação escalável em número de nós e interoperável entre diferentes interfaces de comunicação;
- **Avaliação de atraso de mensagens das integrações quando em disputa de acesso:** A comunicação entre *Smart Meters* DNP3 deve conter latência dentro dos requisitos relacionados a aplicação SG onde se encontra. Ao utilizar interfaces sem fio, a latência é influenciada pelo número de dispositivos em disputa de acesso ao meio compartilhado. Este trabalho realizou análises nesse sentido que mostraram que a latência da integração IEEE 1815/ 802.15.4 é mais afetada pelo aumento do tempo de disputa que a integração IEEE 1815/ 802.11b;
- **Análise de possíveis aplicações SG as quais as integrações podem ser aplicadas:** Ao analisar as características teóricas das interfaces de comunicação utilizadas, bem como das simulações realizadas, pode-se afirmar sobre as possíveis aplicações onde as integrações podem ser aplicadas. Os requisitos de latência de teleproteção, em seu contexto geral, apontam a interface IEEE 802.15.4 como inadequada para esta aplicação por apresentar alta latência na presença de cenários de disputa de acesso, baixa vazão de dados e baixa resiliência de segurança implementada. Caso que não ocorre, para a interface IEEE 802.11, que se torna factível pois apresenta latência, vazão de dados e robustez de segurança compatíveis com tal aplicação crítica SG.

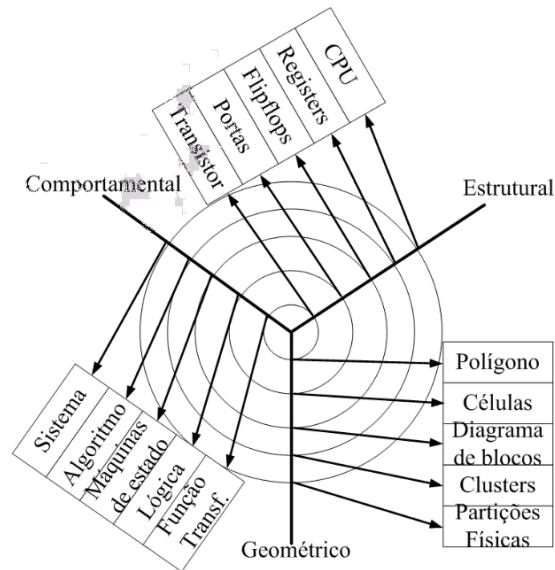
1.3 Organização do trabalho

Este trabalho está organizado em Capítulos. Uma visão geral dos principais conceitos da linguagem de descrição VHDL, é apresentada no Capítulo 2. O Capítulo 3 aborda o conceito de *Smart Grid* e ainda apresenta o protocolo DNP3 juntamente com a sua modelagem em VHDL. O Capítulo 4 traça um comparativo entre algumas interfaces de comunicação, descreve o protocolo IEEE 802.15.4 e IEEE 802.11 com suas respectivas modelagem em VHDL e integrações com o IEEE 1815. Os detalhes dos cenários de simulação e os resultados estão no Capítulo 5. Por fim, o Capítulo 6 descreve as conclusões e trabalhos futuros.

2 LINGUAGEM VHDL

Em 1986, o VHDL, se tornou um padrão IEEE (IEEE, 2002) para programação de circuitos integrados de altas velocidades. Os domínios de modelagem estrutural, comportamental, geométrica (PERRY, 2002) são apresentados na Figura 1.

Figura 1 – Domínios e Níveis de Abstração HDL (Diagrama Y de Gajski e Kuhn).



Fonte: (ROLFES, 2009; GONÇALVES, 2005).

As metodologias de projeto de descrição de *hardware* podem utilizar as abordagens *bottom-up* ou *top-down*. A abordagem *top-down* define que o projeto deve ser iniciado a partir de um *hardware* ou *software* complexo e prosseguir na direção de simplificar suas funcionalidades, por exemplo (BRITO, 2008). Após ciclos de simulações o projeto deve ser adequado na abordagem *bottom-up* que efetua a integração de funcionalidades simples em estruturas complexas (GONÇALVES, 2005).

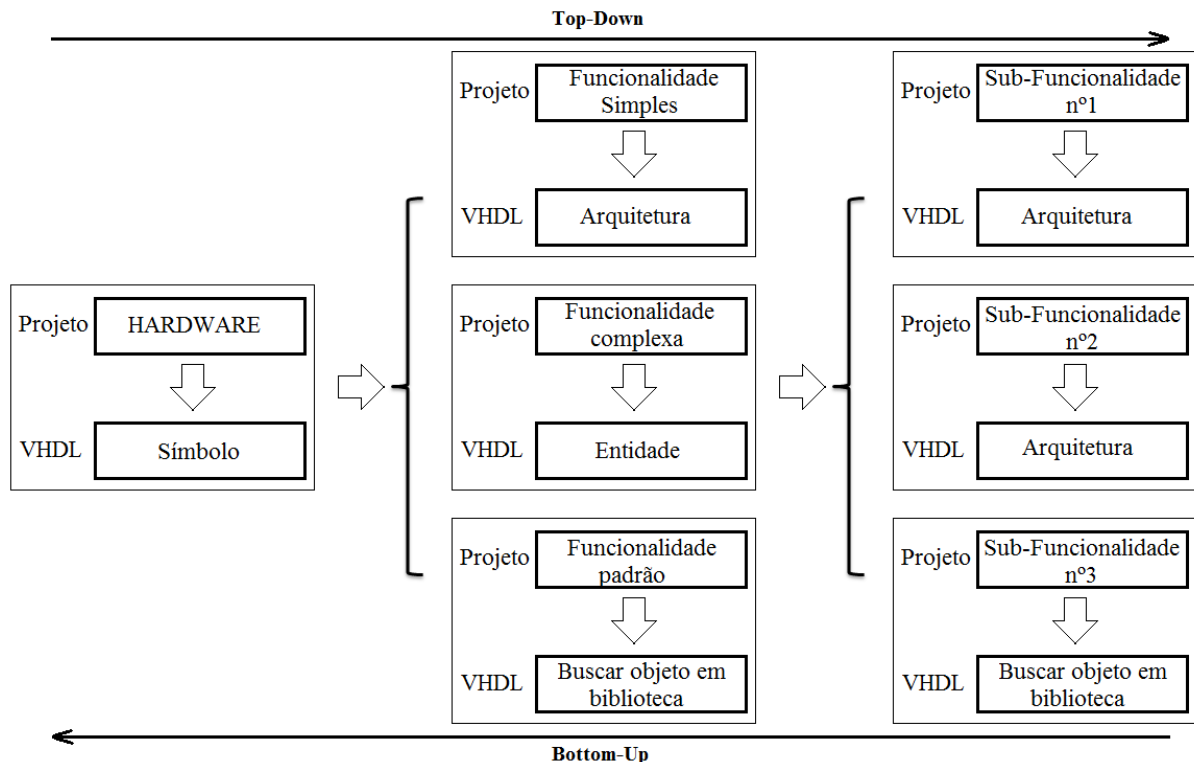
O termo central dessa hierarquia de descrição do *hardware* proposto é símbolo no esquemático. O símbolo contém todo código VHDL modelado, estruturas especificadas através de arquiteturas, objetos padrão já instanciados em biblioteca ou outro esquemático composto por uma ou mais entidades. Ou seja, ao realizar um projeto de *hardware*, o símbolo pode conter estruturas já existentes e instanciadas no mesmo.

Caso exista alguma funcionalidade específica da descrição do *hardware* que não está instanciada em uma biblioteca, pode ser realizada uma especificação em uma arquitetura que represente seu comportamento. Entretanto, se a funcionalidade pode ser segmentada e simplificada, pode-se criar uma ou mais entidades em outro nível hierárquico para representar

a funcionalidade complexa.

Essa hierarquia de níveis de símbolos deve ser realizada de forma que se tenha a estrutura de descrição lógica completa do *hardware* proposto, partindo de ambiente comportamental de funcionalidades, do dispositivo que se pretende descrever, em direção a arquiteturas que definam a lógica por trás dessas funcionalidades. A estrutura hierárquica descrita anteriormente é apresentada na Figura 2.

Figura 2 – Abordagens de desenvolvimento hierárquico de projeto HDL em *bottom-up* e *top-down*.



Fonte: Elaboração do Autor.

A comunicação entre entidades diferentes é dada pela utilização de portas, sinais lógicos, ou interfaces de entrada ou saída. Sendo assim, essa característica torna compatível a modelagem de protocolos de comunicação e suas camadas, que na sua essência se comunicam em sinais lógicos ou bits.

As arquiteturas definem o comportamento da entidade através de funções, processos ou procedimentos. Uma variável pode ser declarada para assumir valores inteiros, reais, booleanos, *strings*, entre outros. Além disso, quando declarada em uma arquitetura, a mesma pode ser utilizada por todos os processos presentes na mesma arquitetura, ou seja, se tornam uma variável global da entidade. As variáveis declaradas nos processos são locais e, portanto, somente utilizadas dentro do próprio processo.

Dentro de uma arquitetura, os processos são executados paralelamente. Isso porque em

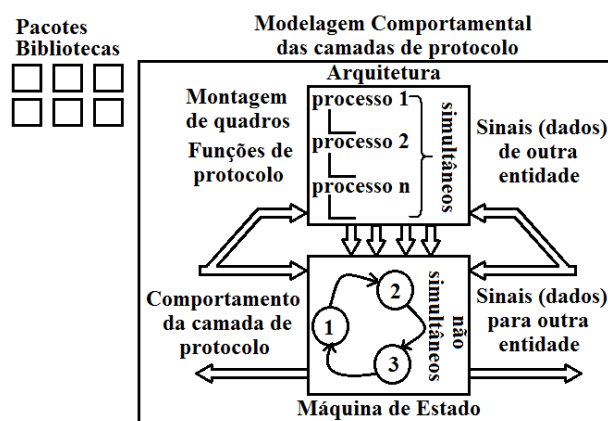
um *hardware* real, seus componentes podem funcionar de forma concorrente entre si, o que determina que as mudanças de sinais de entrada possuam atraso que se propaga para os componentes e sinais de saída. Entretanto, as instruções pertencentes a cada processo podem ser simuladas ou sintetizadas opcionalmente de forma paralela ou procedural, utilizando critérios de espera (*wait*) entre as ações.

Neste trabalho utiliza-se a abordagem comportamental para descrição dos protocolos através de arquiteturas do tipo máquina de estado finito (*Finite State Machine - FSM*). Contudo a utilização de abordagem de máquinas de estado, de forma exclusiva, não é compatível para realização montagem ou segmentação de quadros de comunicação. Para realizar tais tarefas específicas com maior agilidade foram modeladas arquiteturas padrão em adição as máquinas de estado.

Assim, para descrição dos protocolos de comunicação propostos neste trabalho, foi utilizada a abordagem comportamental e sistêmica. Conforme ilustrado na Figura 3, a modelagem comportamental de cada camada, dos protocolos IEEE 1815, IEEE 802.15.4 e IEEE 802.11b, dividiu-se em duas entidades.

Uma das arquiteturas é responsável pela montagem de quadros e/ou realização de funções como cálculos, contagem de ciclos do sistema, entre outros. Os quadros formados são repassados às entidades de máquinas de estado, responsáveis pelo comportamento da camada do protocolo em questão. Através das transições de estado modeladas, a transmissão de sinais entre as camadas dos protocolos ocorre de acordo com a especificação da norma.

Figura 3 – Abordagem Geral adotada.



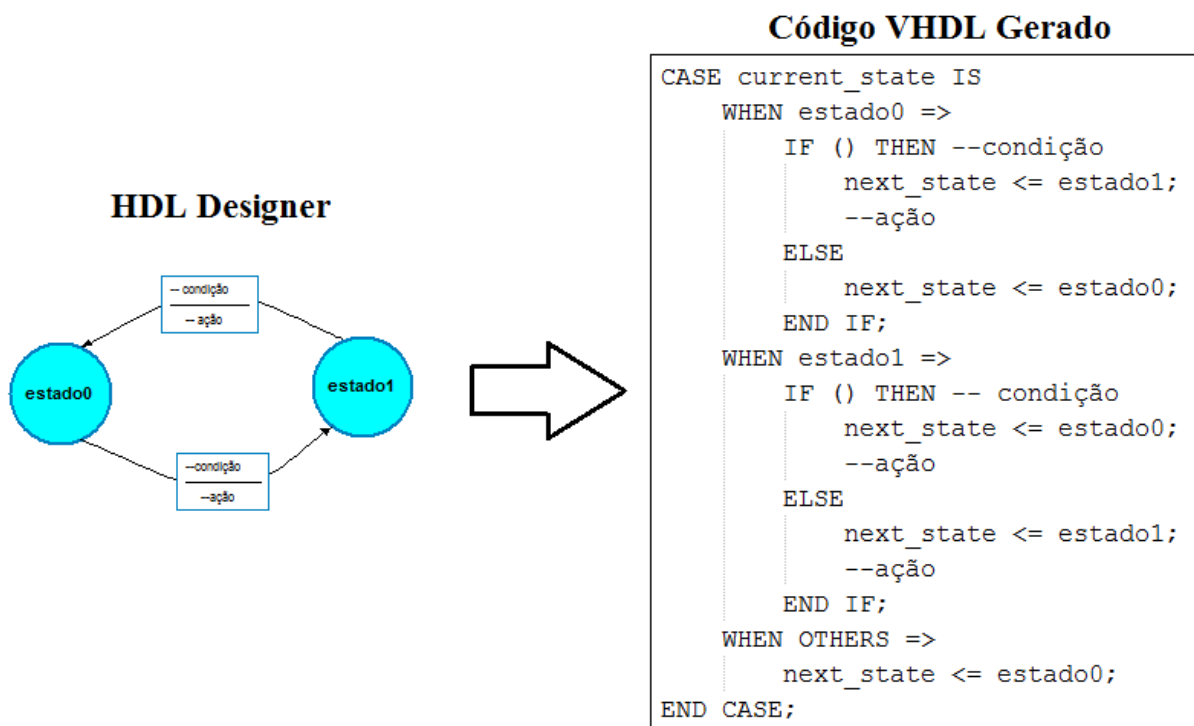
Fonte: Elaboração do próprio autor.

A modelagem deste trabalho foi realizada através do *Framework HDL (Hardware Description Language) Designer* (MENTOR GRAPHICS, 2012), acelerando o desenvolvimento do projeto e automatização da produção do código, através de uma interface gráfica intuitiva ou GUI (*Graphic User Interface*). A linguagem VHDL (PERRY, 2002), bem

como o ambiente *HDL Designer*, são ferramentas comumente empregadas na indústria para projetos de componentes e dispositivos.

A principal vantagem da utilização deste *Framework* está na diminuição do tempo dos estágios iniciais e na documentação dos comentários relacionados a estrutura geral do código, em VHDL, dos projetos de *hardware* desenvolvidos para posterior síntese e aplicação em dispositivos de alto desempenho FPGA (BRITO, 2008). Na Figura 4 ilustra-se como o *HDL Designer* efetua a tradução de máquinas de estados, configuradas via *framework*, para código VHDL.

Figura 4 – Conversão de máquinas de estado do *framework* para código VHDL.



Fonte: Elaboração do próprio autor.

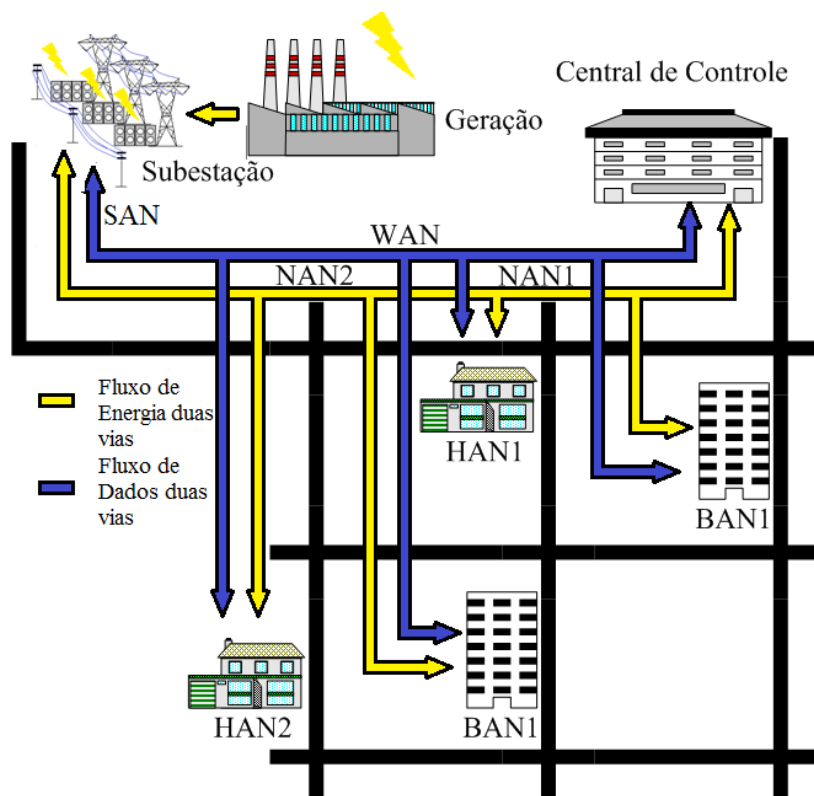
Para possibilitar a análise do comportamento das camadas, baseada no mapeamento do conjunto de sinais de entradas/saídas, foram realizadas simulações através do *software* *Modelsim* (MODELSIM, 2012), descritas com detalhes no Capítulo 5. O *Modelsim* é um simulador de códigos VHDL que utiliza o conceito de eventos de E/S através de gráfico de onda lógica.

3 SMART GRID

A rede de energia convencional consiste em dois sistemas primários. Primeiro, um sistema de transmissão da energia das fontes geradoras até as subestações. Segundo, um sistema de distribuição de energia entre as subestações e os consumidores (OFFICE OF ELECTRICITY DELIVERY AND ENERGY RELIABILITY, 2009). Nesse contexto, as concessionárias têm controle e detecção de falhas limitados às subestações de transmissão e, portanto, não alcançando o sistema de distribuição como um todo (FANG et al., 2012). Assim, a localização e assistência aos usuários domésticos depende que os mesmos denunciem as falhas do sistema para que ações sejam executadas por equipes de campo, o que gera um maior atraso para as soluções do problema.

O conceito SG estabelece que, em adição ao fluxo de energia, a concessionária tenha um fluxo de dados de duas vias em todos os setores da rede até os consumidores (FANG et al., 2012), formando assim sua arquitetura clássica apresentada na Figura 5. Além disso, as redes inteligentes possibilitam a geração distribuída através de fontes renováveis de energia em ambiente doméstico (HASSAN; RADMAN, 2010). O fluxo de dados possui enlaces de comunicação entre as aplicações SG até a Central de Controle através de um *Backbone*.

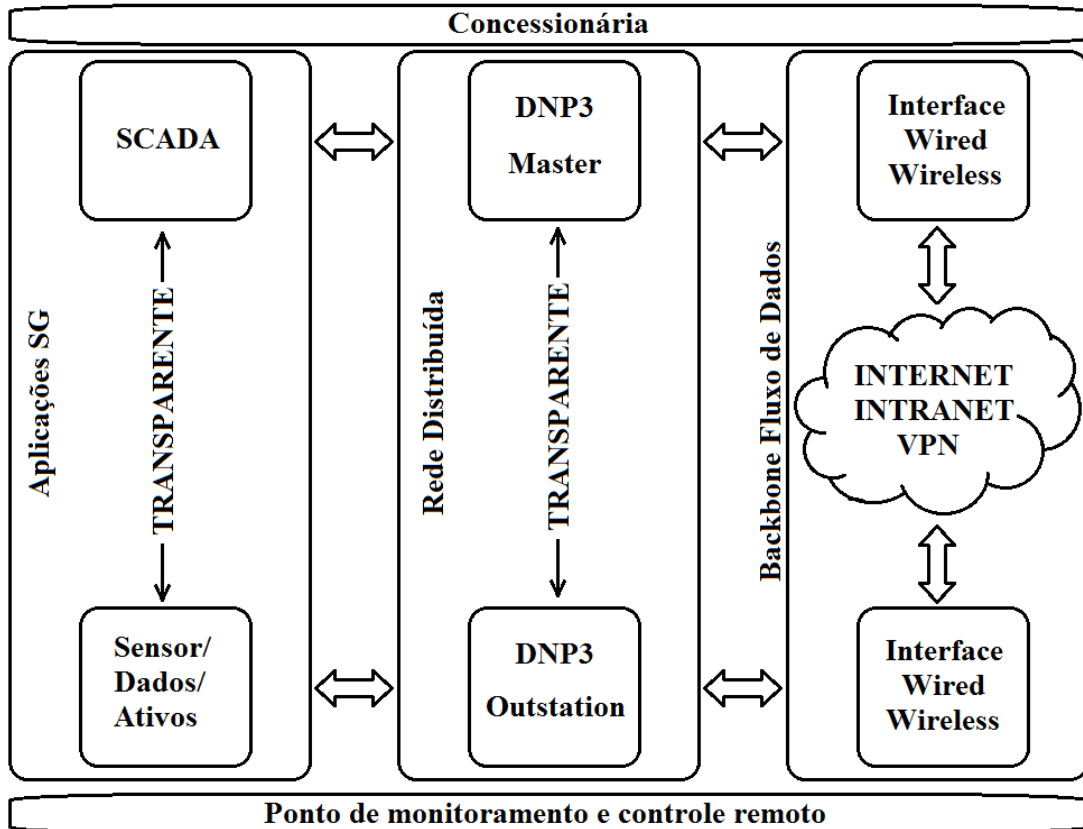
Figura 5 - Arquitetura Clássica de uma rede *Smart Grid*.



Fonte: Elaborado pelo próprio autor.

A descrição apresentada na Figura 6, detalha cada camada do fluxo de dados que formam a conexão entre a concessionária e o ponto de monitoramento remoto.

Figura 6 – Fluxo de dados Smart Grid.



Fonte: Elaborado pelo próprio autor.

As aplicações SG são consideradas o objetivo de monitoramento da concessionária. Como exemplo de tipo de aplicação SG pode-se citar o monitoramento e teleproteção de ativos de subestações. Todo o monitoramento é supervisionado por um sistema de controle SCADA (*Supervisory Control and Data Acquisition*) presente no Centro de Operações.

Através de sensores específicos a cada informação que se pretende monitorar, a coleta de dados é realizada e enviada para dispositivos distribuídos. Estes dispositivos, denominados medidores inteligentes, gateways ou *Smart Meters*, estão organizados em rede e têm, entre outras funções, o papel de armazenar os dados para retransmiti-los quando necessário ou solicitado. Ou seja, o envio dos dados pode ser espontâneo, sem necessidade de requisição, ou como resposta de requisições da Central de Controle (ZHOU et al., 2012).

O enlace de Backbone é heterogêneo e formado por uma estrutura hierárquica composta por HAN, BAN, NAN, WAN (*Home / Build / Neighborhood / Wide Area Networks*). As tecnologias de comunicação dessa estrutura hierárquica devem ser compatíveis aos requisitos específicos a cada rede de área.

A comunicação do fluxo de dados entre as aplicações SG remotas e o sistema SCADA na Central de controle, assim como a comunicação na rede distribuída deve ser transparente aos tipos de interfaces utilizados no *Backbone* de dados (FADLLULAH et al., 2011).

O *Backbone* de dados pode ser realizado por Intranet, ambiente de rede criado apenas para conexão entre aplicação SG e SCADA, ou Internet, onde os dados SG trafegam pela rede mundial de computadores e utilizam a estrutura de telecomunicações já fornecida pela operadora. Como uma nova infraestrutura de comunicação, a implementação em Intranet é uma solução que traz garantias de desempenho e segurança em contraste com seu alto custo. Analisando a solução de Internet, entretanto, o custo é reduzido assim como suas garantias. A assinatura de *link* dedicados e VPN (*Virtual Private Network*) podem mitigar essas métricas de escolha.

A implementação do sistema de comunicação deve satisfazer a requisitos que possibilitem a tomada de decisões sem atrasos e que seja confiável. Considerando que existem várias aplicações SG, como PMU (*Phasor Measurement Unit*), WACS/ WAMS/ WAPS (*Wide Area Control/ Measurement/ Protection Schemes*), teleproteção de Subestações, AMI (*Advanced Metering Infrastructure*) *Reading, Demand Response*, entre outros, adequar um sistema de comunicação a cada uma se torna mais complexo.

A fim de validar a modelagem e simulação realizada neste trabalho, foram utilizados os requisitos da norma IEEE 2030 (IEEE, 2011a), que define os requisitos mínimos de atraso de acordo com a área de atuação da rede de comunicação. Os autores (LU et al., 2013; SADEGHI et al., 2012) fizeram uma análise dos requisitos de latência em relação as aplicações SG. Na Tabela 1 apresenta-se os requisitos de latência para as aplicações SG.

Tabela 1 - Requisitos de Latência para Aplicações SG.

		Classe Nível 1		Classe Nível 2		Classe Nível 3	
Nível de Confiabilidade		Alta		Média		Baixa	
Prioridade		1		2		3	
Requisitos de Latência		Baixa, Baixa-Baixa	< 3ms, < 16 ms	Média, Alta	< 160 ms, < 300 ms	Alta, Alta-Alta	< 300 ms, > 300 ms
		Média, Alta	< 160 ms, < 300 ms	-----	-----	-----	-----
Aplicações SG	LU et al, 2013	Teleproteção (dentro e entre subestações)		Comunicação WAN, Dados SCADA		Resposta à Demanda, Leitura AMI, Aplicações de Distribuição	
	SADEGHI et al, 2012	Teleproteção (dentro e entre subestações), PMU		Comunicação WAN, WACS		Dados SCADA	

Fonte: (IEEE, 2011a; LU et al., 2013; SADEGHI et al., 2012).

Analisando os requisitos de forma geral e integrada das aplicações, pode-se afirmar que o maior nível de exigência em requisitos de latência é encontrado em aplicações de teleproteção de subestações e medição de fases na geração. Seguindo este raciocínio, o menor nível de exigência está na comunicação da aplicação SG de última milha, ou seja, de distribuição para o consumidor final.

Os requisitos apresentados na Tabela 1 representam uma métrica para dimensionamento e escolha de tecnologias que podem compor a comunicação em cada aplicação SG em um contexto mais geral. Contudo, ao se analisar uma aplicação crítica como a teleproteção de subestações, por exemplo, pode-se encontrar dados de características críticas e não críticas e, por consequência, com maior ou menor rigidez de requisitos.

A seção 3.1 efetua uma análise do fluxo de dados presente na aplicação SG de teleproteção de subestações. O objetivo é levantar as características de quantidade, frequência de envio e níveis de criticidade e prioridade de envio dos dados trafegados.

A seção 3.2 efetua a descrição das características do protocolo de rede distribuída, DNP3, bem como sua modelagem em VHDL.

3.1 Aplicação de teleproteção de subestações

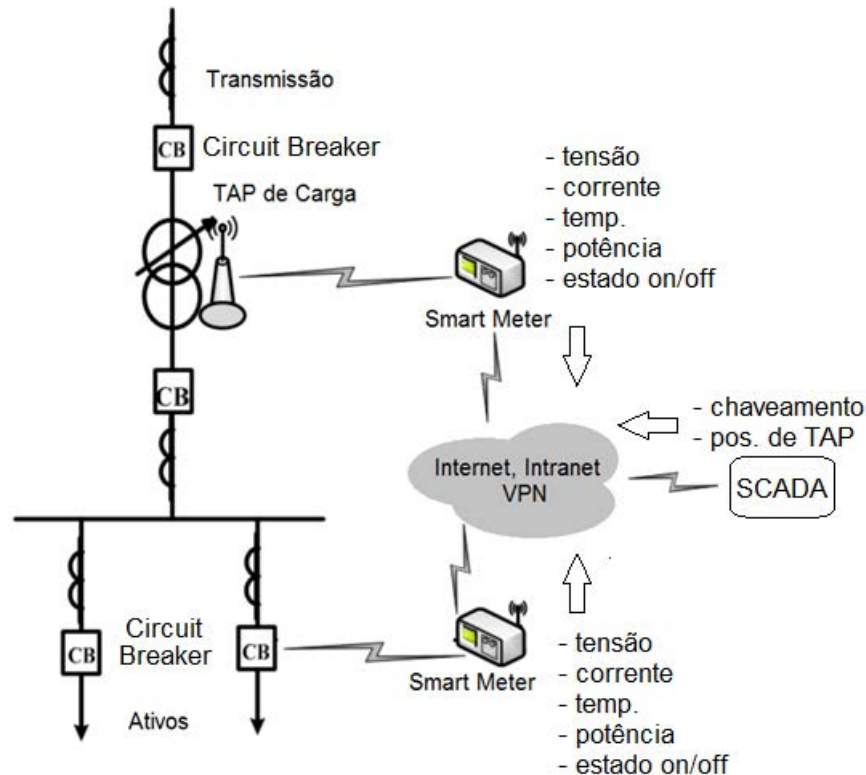
Atualmente, as subestações de transmissão e distribuição possuem monitoramento e controle local de ativos para funcionamento adequado na garantia de disponibilidade e qualidade de energia (LU et al., 2013). O conceito SG permite que essas ações sejam monitoradas e tomadas remotamente através do fluxo de dados. A fim de tornar viável tal comunicação, o fluxo de dados entre subestação e Central de Controle deverá ser rápido. A latência não deve ultrapassar limite de 16 *ms*, segundo norma (IEEE 2030, 2011).

Através do cruzamento de dados coletados de forma detalhada de todo o ambiente de distribuição de energia, o controle de qualidade de energia se torna mais preciso e ágil. Uma vez que se monitore o perfil de carga o chaveamento de ações controladoras de tensão, por exemplo, pode ser realizada em tempo real.

Além de permitir maior controle na qualidade de fornecimento de energia, o fluxo de dados permite ainda que os ativos (equipamentos) da concessionária tenham seu funcionamento monitorado e controlado remotamente. Essa diretriz pode prevenir falha ou dano desses ativos. Um exemplo de teleproteção de subestações é ilustrado na Figura 7.

Os dados de natureza analógica como corrente, tensão, potência e temperatura representam maior volume de espaço nas mensagens e maior periodicidade de envio. Pois estes, devem ser convertidos de sinais analógicos para vetores digitais. Tal conversão delimita uma relação linear entre o valor digital e a representação além da resolução analógica desejadas. Dependendo do equipamento que efetua a medição o tamanho desses vetores pode variar entre 8, 16 ou 32 bits.

Figura 7 – Teleproteção de Subestação.



Fonte: (PARIKH et al., 2010).

Os dados de natureza digital, estado do equipamento (*on/off*), abertura e fechamento de chaves, posicionamento de TAP de transformador, representam geralmente menor volume de espaço (1 a 2 bits) nas mensagens, mas são enviados apenas na mudança de algum estado. Se configurados como eventos críticos, podem ter maior prioridade de envio quando comparado a dados periódicos.

Efetuada uma análise de escolha de tecnologia que atendam aos requisitos específicos de cada tipo de dado pode se afirmar que:

- Os dados de natureza analógica têm sua importância na continuidade de recebimento, cruzamento de informações e velocidade moderada. Entretanto, a vazão da quantidade de dados torna-se fator determinante para que não se criem

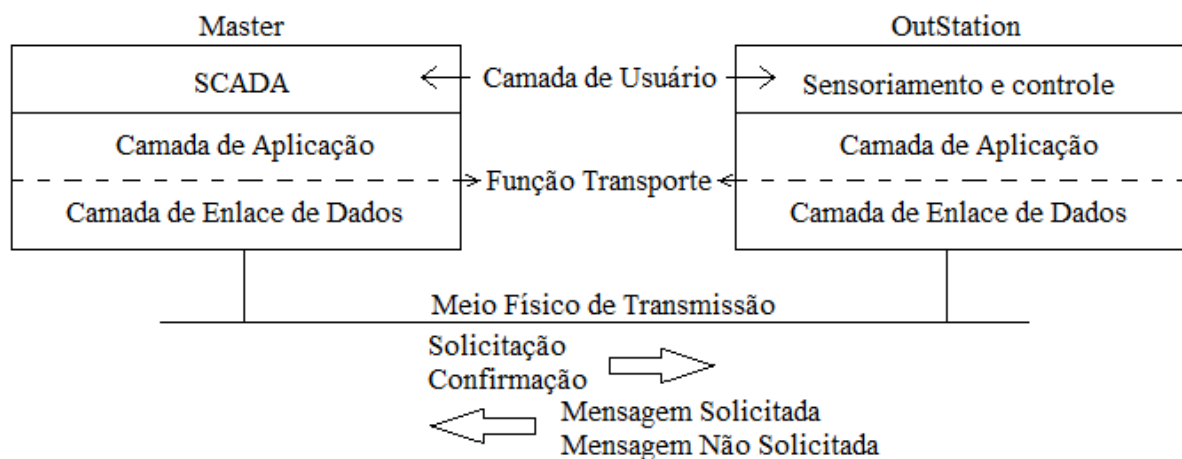
grandes atrasos na transmissão interna a subestação. Sendo assim, as tecnologias envolvidas na transmissão deste tipo de informação devem dar suporte a transmissão ou vazão de grandes quantidades de dados, mesmo que apresentem latência pouco acima dos requisitos apresentados na Tabela 1;

- Os dados de natureza digital são eventuais e em baixas quantidades de dados. Contudo, devido ao fator crítico que representam na teleproteção de ativos, a baixa latência entre a ocorrência do evento na subestação, sua passagem pelo *Smart Meter*, e sua chegada à Central de Operações torna-se importante para evitar faltas ou prejuízos com ativos. Assim, pode-se afirmar que, para transmissão de dados dessa natureza crítica, a tecnologia deve apresentar altas velocidades e confiabilidade de dados, ainda que não dê suporte a alta vazão de dados.

3.2 Protocolo de Rede Distribuída DNP3

A comunicação SG é realizada em rede de *Smart Meters*, que pode operar o protocolo DNP3. A pilha do protocolo DNP3 (IEEE, 2012) define as camadas de usuário, aplicação, função transporte e enlace de dados conforme apresentada na Figura 8.

Figura 8 - Pilha de protocolo DNP3.



Fonte: (IEEE, 2012).

Uma rede DNP3 (IEEE, 2012) possui dois tipos de dispositivos em sua topologia. O dispositivo *Master* geralmente responsável pela solicitação de dados aos outros dispositivos e ainda pelo escoamento das informações para o sistema SCADA. O dispositivo *Outstation* responsável pela coleta, armazenagem e envio de dados para o *Master*.

A camada de usuário é composta de dois módulos: o sistema SCADA (conectado ao *Master*) que é responsável pelo processamento das requisições e dados coletados; e os sensores e atuadores (conectados ao *Outstation*) que efetuam a coleta de dados ou execução de ações através de comandos recebidos da central de controle. Portanto, a camada de aplicação recebe as possíveis requisições SCADA, e as traduz em comandos *Application Program Interface* (API). Esses comandos são compreendidos como fragmentos de no máximo 4 kbytes de tamanho, e são utilizados para transmitir dados para as camadas inferiores.

Os fragmentos formados na camada de API contêm dois octetos principais como cabeçalho, o Controle de Aplicação e o Código de Função, conforme apresentado na Tabela 2. Os bits do octeto de Controle de Aplicação são descritos na Tabela 3.

Tabela 2 – Cabeçalho Camada API do protocolo DNP3.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	<=BITS
Controle de Aplicação								Código de Função								

Fonte: (IEEE, 2012).

Tabela 3 – Controle de Aplicação.

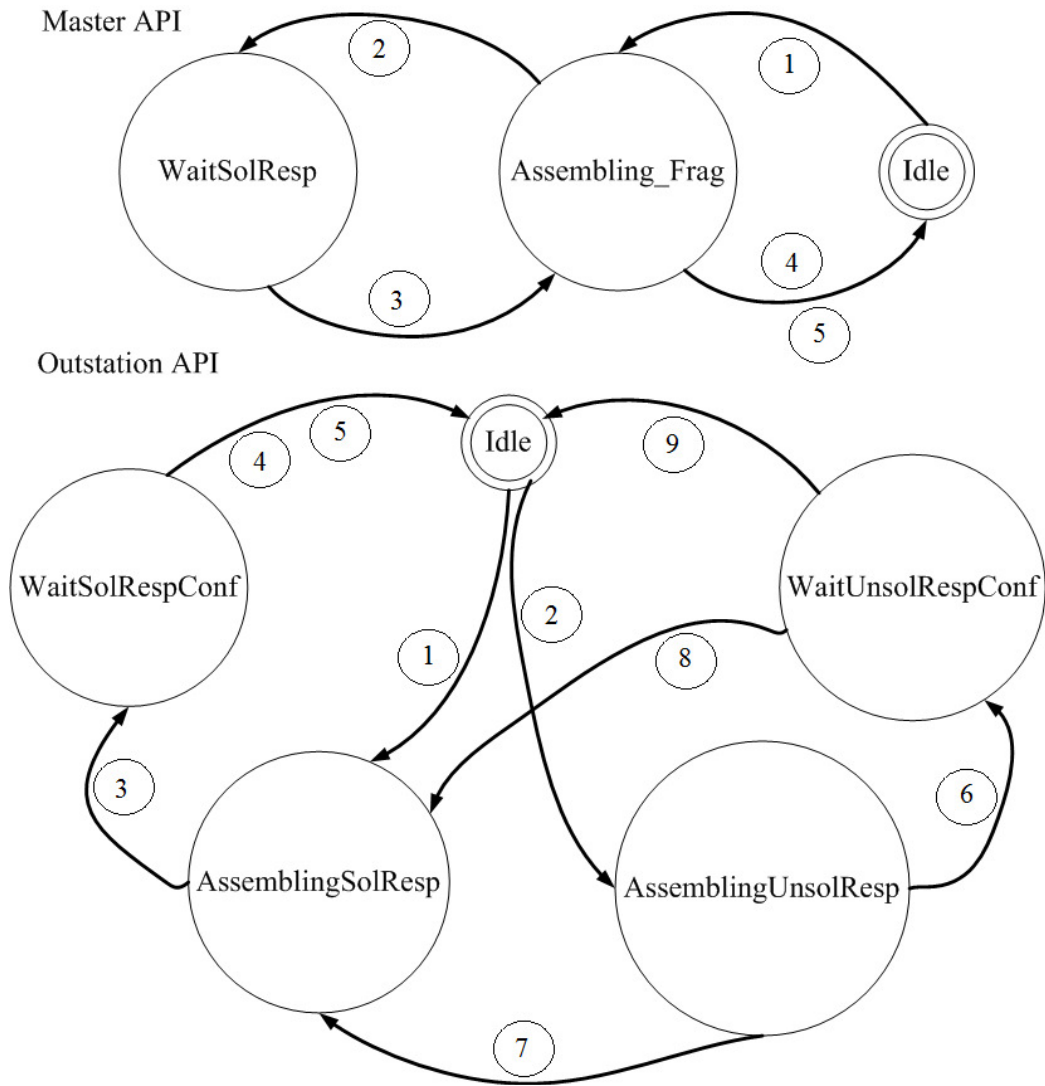
Bits	Nome	Descrição
7	FIR	<i>First</i> – ‘1’ para o primeiro fragmento de uma sequência
6	FIN	<i>Final</i> – ‘1’ para o último fragmento de uma sequência
5	CON	<i>Confirm</i> – ‘1’ para o fragmento que necessita de confirmação
4	UNS	<i>Unsolicited</i> – ‘1’ para o fragmento que corresponde a uma mensagem não solicitada
3-0	SEQ	<i>Sequence</i> – número de 0 a 15 da ordem daquele fragmento dentro da sequência pretendida

Fonte: (IEEE, 2012).

Entre os octetos do cabeçalho API, o Código de Função define qual a finalidade ou comando API que cada fragmento representa, por exemplo, comandos de requisição de leitura, de resposta, entre outros. Através do octeto pode-se diferenciar até 256 códigos de função.

Além da entidade responsável pela montagem de quadros e execução de comandos (*MAR_API* ou *OAR_API*), este trabalho modelou as máquinas de estado (*State Machines* - SM) referentes ao comportamento da camada API (*MSM_API* ou *OSM_API*) de ambos dispositivos DNP3. Na Figura 9 ilustra-se como foram modeladas as SMs e enumera-se as transições de estado que são descritas nas Tabelas 4 e 5. As transições de estado em código VHDL estão no Anexo B.

Figura 9 – Máquinas de estado camadas API dos dispositivos DNP3 *Master* e *Outstation*.



Fonte: Elaborado pelo próprio autor.

Tabela 4 – Transições de Estados camada DNP3 API *Master*.

Transição n°	Transições		Descrição	Ações
	Atual	Próximo		
1	<i>Idle</i>	<i>Assembling_Frag</i>	Gatilho para envio de comando	Aguardar montagem de fragmento
2	<i>Assembling_Frag</i>	<i>WaitSolResp</i>	Fragmento de comando devidamente montado	Repassar fragmento a camada inferior e aguardar Resposta API
3	<i>WaitSolResp</i>	<i>Assembling_Frag</i>	Recebe Resposta API	Aguarda montagem de fragmento de confirmação API caso haja necessidade
4	<i>Assembling_Frag</i>	<i>Idle</i>	Fragmento de confirmação devidamente montado	Repassar fragmento a camada inferior
5	<i>Assembling_Frag</i>	<i>Idle</i>	Não há necessidade de confirmação	-----

Fonte: Elaborado pelo próprio autor.

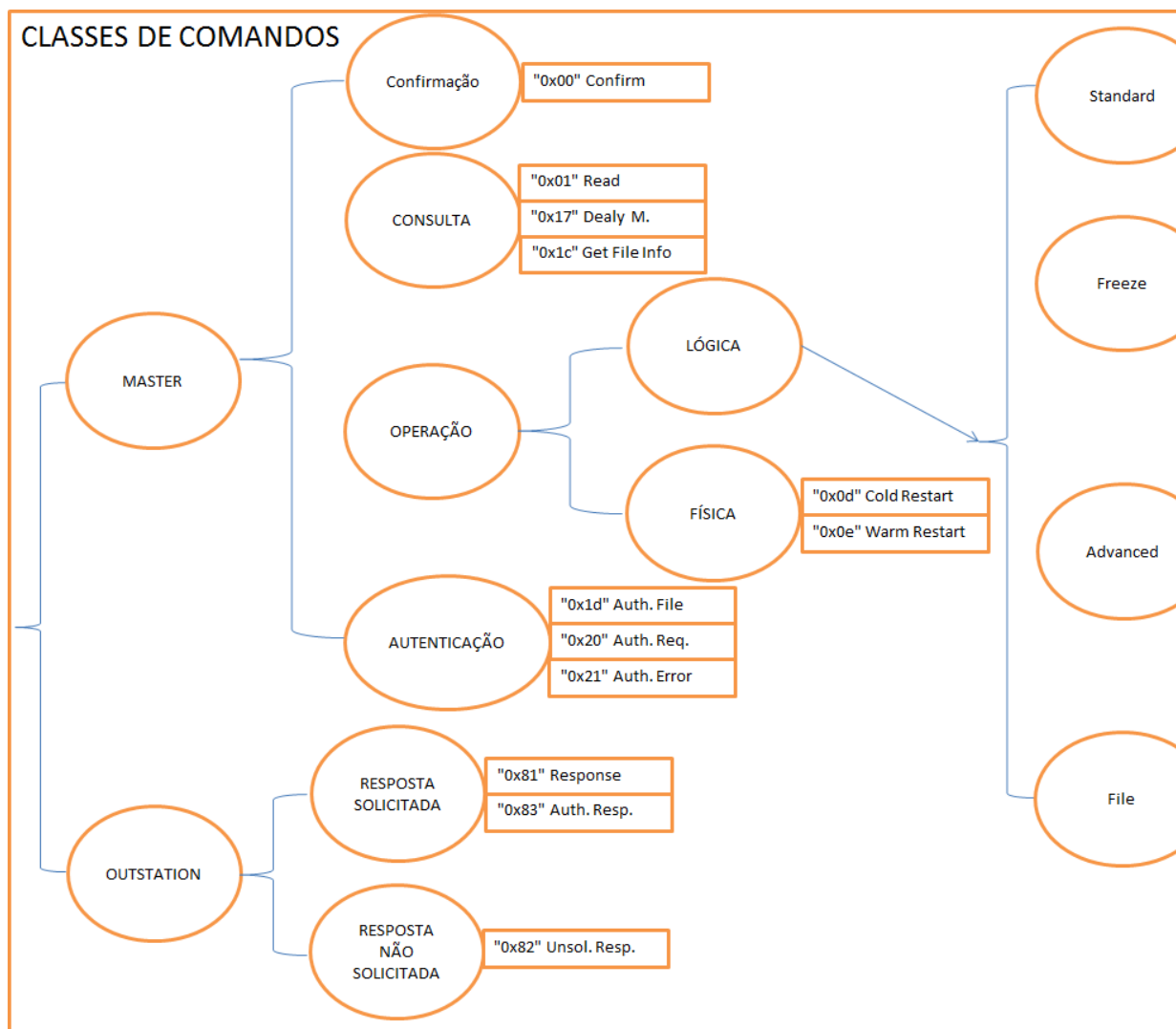
Tabela 5 – Transições de Estados camada DNP3 API *Outstation*.

Transição n°	Transições		Descrição	Ações
	Atual	Próximo		
1	<i>Idle</i>	<i>AssemblingSolResp</i>	Recebe comando API	Aguardar montagem de fragmento com resposta solicitada
2	<i>Idle</i>	<i>AssemblingUnsolResp</i>	Gatilho para evento crítico	Aguarda montagem de fragmento com resposta não solicitada
3	<i>AssemblingSolResp</i>	<i>WaitSolRespConf</i>	Fragmento de resposta devidamente montado	Repassar fragmento a camada inferior e aguardar recebimento de confirmação API se necessário
4	<i>WaitSolRespConf</i>	<i>Idle</i>	Recebe confirmação API	-----
5	<i>WaitSolRespConf</i>	<i>Idle</i>	Não há necessidade de confirmação	-----
6	<i>AssemblingUnsolResp</i>	<i>WaitUnsolRespConf</i>	Fragmento de resposta não solicitada devidamente montado	Repassar fragmento a camada inferior e aguardar recebimento de confirmação API
7	<i>AssemblingUnsolResp</i>	<i>AssemblingSolResp</i>	Recebe um comando API	Cancelar montagem de resposta não solicitada e montar resposta solicitada
8	<i>WaitUnsolRespConf</i>	<i>AssemblingSolResp</i>	Recebe um comando API	Cancelar espera por confirmação e montar resposta solicitada
9	<i>WaitUnsolRespConf</i>	<i>Idle</i>	Recebe confirmação API	-----

Fonte: Elaborado pelo próprio autor.

Cada um dos comandos API DNP3 têm uma característica diferente e sua utilização depende da capacidade e recursos do dispositivo *Smart Meter* ao qual está associado. Com o propósito de facilitar sua utilização em diferentes ambientes e *hardwares*, este trabalho seguiu a recomendação do (IEEE, 2012), separando os comandos da camada API em classes, conforme apresentado na Figura 10.

Figura 10 - Classes de Comandos API do protocolo DNP3.



Fonte: (IEEE, 2012).

As funcionalidades foram separadas por característica, finalidade e nível de complexidade.

As classes são:

- Classe “Confirmação”: possui apenas um comando que confirma o recebimento de um evento de resposta;
- Classe “Consulta”: possui comandos de consulta ao *OutStation*;
- Classe “Operação”: possui comandos de configuração, divididos em subclasses, “Lógica” (*software*) ou “Física” (*hardware*);
- Subclasse “Lógica”: possui comandos de configuração divididos em “Standard” (básicos), “Freeze” (bufferização), “Advanced” (avançados) e “File” (relacionados a arquivos);
- Subclasse “Física”: possui comandos de reinicialização de *hardware*;
- Classe “Autenticação”: possui comandos de funcionalidade de segurança;
- Classe “Resposta Solicitada”: possui comandos de resposta específicos na requisição recebida pelo *OutStation*;
- Classe “Resposta não Solicitada”: comando de envio espontâneo de mensagens que reportam eventos críticos.

A implementação dos comandos dependerá das características da rede DNP3 que se pretende criar. Além da capacidade individual, o dispositivo *Master* não deverá efetuar o envio de comandos a um *Smart Meter OutStation* que não esteja apto a respondê-lo com seus recursos.

Dentre os vários comandos DNP3 modelados, os cenários simulados no Capítulo 5 utilizam apenas o *Delay Measure*. O comando “*Delay Measure*” é enviado pelo *Master* ao *Outstation* com a finalidade de medir o atraso entre eles.

Assim que os comandos API são formatados em fragmentos, estes são repassados para a função de transporte (TF) que, através da segmentação dos fragmentos, efetua a transmissão em porções de dados menores denominadas de segmentos. Os segmentos possuem tamanho fixo de 250 octetos. Além disso, essa função também remonta os segmentos em fragmentos durante a recepção de dados. A função transporte insere apenas um octeto como cabeçalho, conforme descrito na Tabela 6.

Tabela 6 – Cabeçalho Função Transporte.

Bits	Nome	Descrição
7	FIR	<i>First</i> – ‘1’ para o primeiro segmento de uma sequência
6	FIN	<i>Final</i> – ‘1’ para o último segmento de uma sequência
5-0	SEQ	<i>Sequence</i> – número de 0 a 63 da ordem daquele segmento dentro da sequência pretendida

Fonte: (IEEE, 2012).

A camada de enlace de dados (DLL) monta os segmentos em datagramas seguido pela transmissão dos mesmos pelo meio físico. Cada datagrama pode possuir 17 blocos de bits com sistema de verificação de erro *Cyclic Redundancy Check* (CRC) 16 bits por bloco, ou seja, até 34 octetos de um datagrama são destinados à verificação de erro. Além da verificação de erros a camada de enlace retorna confirmações de entrega de datagramas ou ACK (*Acknowledgement*), garantindo assim a confiabilidade de dados.

A camada de enlace possui um cabeçalho de 10 octetos, conforme apresentado na Tabela 7, sendo que 2 campos estão relacionados com sistema de verificação de erro dedicado do próprio cabeçalho. Os bits do octeto de Controle de Enlace são descritos na Tabela 8.

Tabela 7 – Cabeçalho de Enlace do protocolo DNP3.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	<=BITS
Delimitador de início de quadro x"0564"																
Tamanho do <i>payload</i>								Controle de Enlace								
Endereço de Origem 16 bits																
Endereço de Destino 16 bits																
CRC 16 bits																

Fonte: (IEEE, 2012).

Tabela 8 – Controle de Enlace.

Bits	Nome	Descrição
7	DIR	<i>Direction</i> – ‘1’ quando o datagrama é enviado do Master ao Outstation
6	PRM	<i>Primary Message</i> – ‘1’ indica mensagem primária
5	FBC	<i>Frame Count Bit</i> – Bit de Alternância
4	FCV	<i>Frame Count Valid</i> - ‘1’ para FCB válido
4	DFC	<i>Data Flow Control</i> – ‘1’ para buffer de recepção sem espaço de memória
3-0	Código de Função	Código de Função

Fonte: (PADILLA et al., 2012; IEEE, 2012).

O bit *Primary Message*, descrito na Tabela 8, define se o quadro é primário ou secundário na comunicação, assim indicando se o bit 4 ganha a funcionalidade de *Frame Count Valid* (FCV) ou *Data Flow Control* (DFC), respectivamente. O Código de Função do Datagrama define a funcionalidade do quadro, conforme apresentado na Tabela 9.

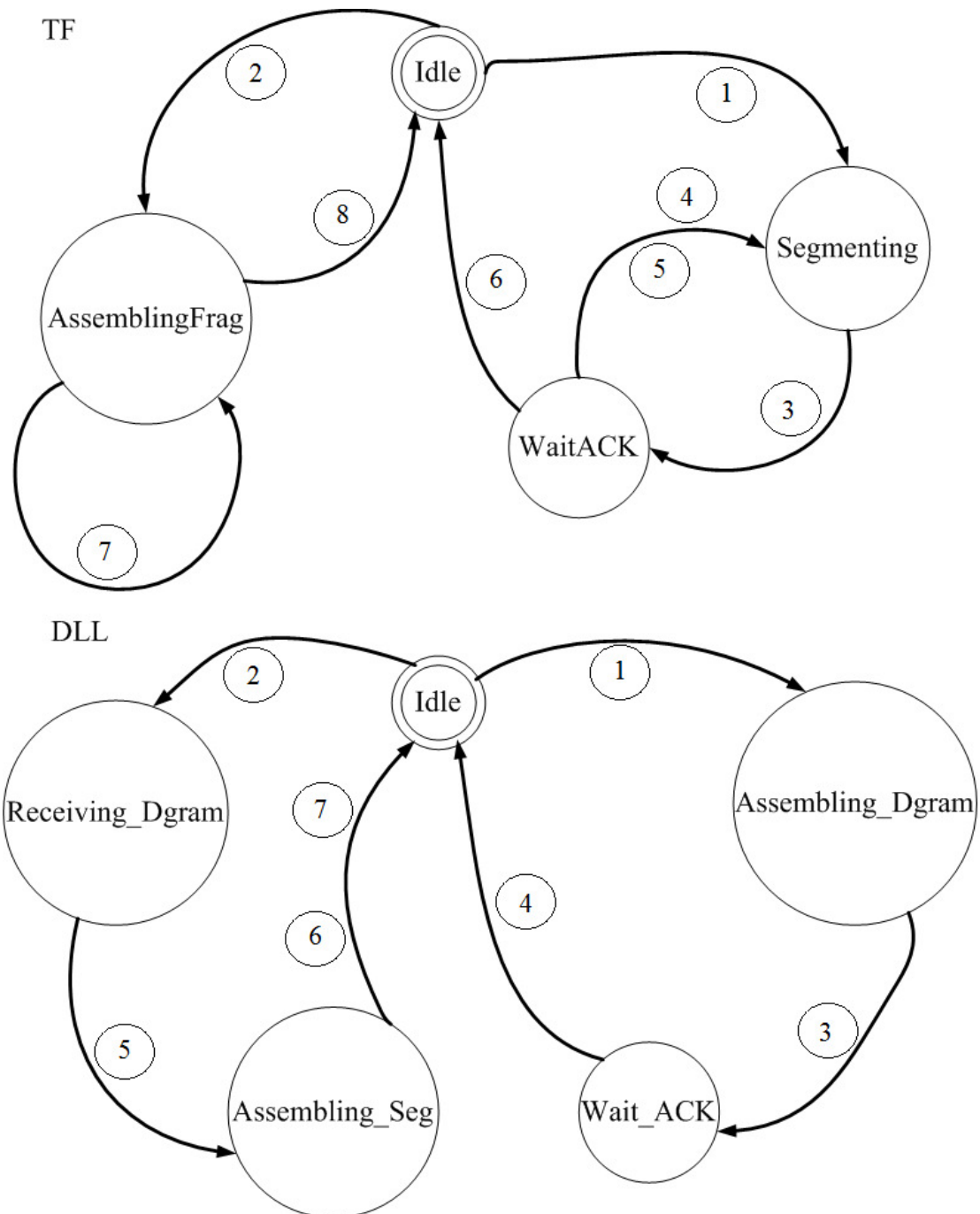
Tabela 9 – Código de Função de Enlace.

Bits 3-0	Código de Função para Quadro Primário	Código de Função para Quadro Secundário
0000	Reiniciar enlace do nó remoto	Quadro ACK
0001	Não se aplica	Quadro NACK
0010	Teste de enlace com nó remoto	Não se aplica
0011	Quadro de dados de usuário que exige ACK	Não se aplica
0100	Quadro de dados de usuário que não exige ACK	Não se aplica
1001	Requisição de Estado de Link	Não se aplica
1011	Não se aplica	Estado de Link
1111	Não se aplica	Serviço de Enlace não suportado pelo nó

Fonte: (PADILLA et al., 2012; IEEE, 2012).

Além das entidades responsáveis pela montagem de quadros e execução de funcionalidades das camadas TF (*MAR_TF* ou *OAR_TF*) e DLL (*MAR_DLL* ou *OAR_DLL*), este trabalho modelou as SMs referentes ao comportamento das mesmas (*MSM_TF* ou *OSM_TF*, *MSM_DLL* ou *OSM_DLL*) de ambos dispositivos DNP3. Na Figura 11 ilustra-se como foram modeladas as SMs e nas Tabelas 10 e 11 descrevem-se as transições de estado das camadas TF e DLL, respectivamente.

Figura 11 – Máquinas de estado camadas TF e DLL.



Fonte: Elaborado pelo próprio autor.

Tabela 10 – Transições de Estados camada TF.

Transição n°	Transições		Descrição	Ações
	Atual	Próximo		
1	<i>Idle</i>	<i>Segmenting</i>	Camada API repassa fragmento	Aguardar segmentação e montagem do primeiro segmento
2	<i>Idle</i>	<i>AssemblingFrag</i>	Camada DLL repassa segmento	Aguardar montagem de fragmento recebido ou próximo segmento
3	<i>Segmenting</i>	<i>WaitACK</i>	Segmento devidamente montado	Repassar para camada inferior e aguardar notificação de recebimento de ACK ou NACK
4	<i>WaitACK</i>	<i>Segmenting</i>	Recebe notificação de ACK e existem mais segmentos a transmitir	Aguardar montagem do próximo segmento
5	<i>WaitACK</i>	<i>Segmenting</i>	Recebe notificação de NACK	Repassar o segmento anterior novamente
6	<i>WaitACK</i>	<i>Idle</i>	Recebe notificação de ACK e não existem mais segmentos a transmitir	-----
7	<i>AssemblingFrag</i>	<i>AssemblingFrag</i>	Recebe próximo segmento para montagem de fragmento	Aguardar montagem de fragmento recebido ou próximo segmento
8	<i>AssemblingFrag</i>	<i>Idle</i>	Fragmento devidamente montado	Repassar fragmento a camada superior

Fonte: Elaborado pelo próprio autor.

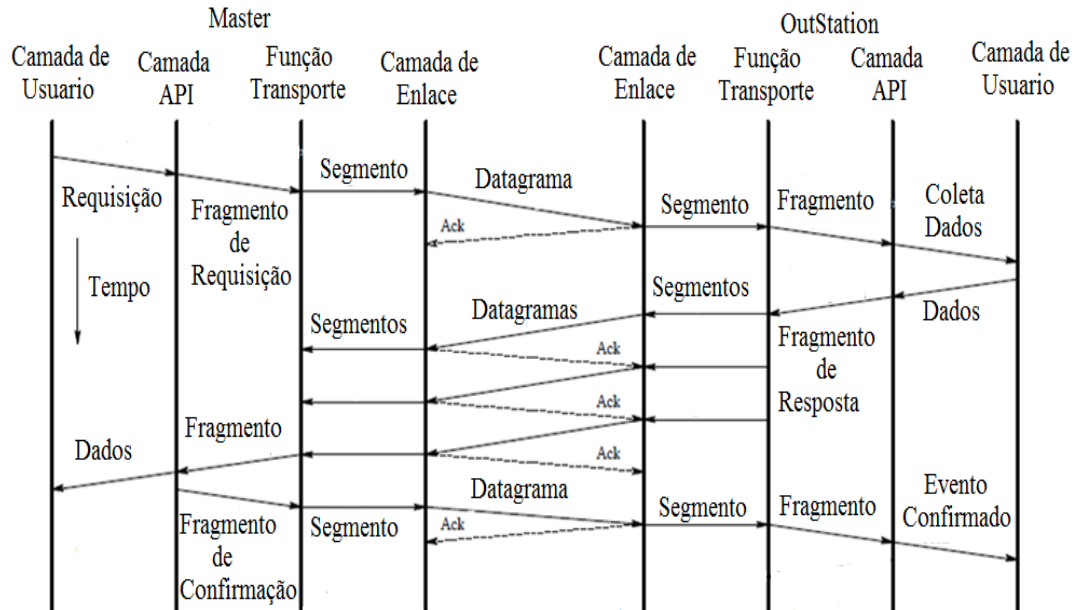
Tabela 11 – Transições de Estados camada DLL.

Transição n°	Transições		Descrição	Ações
	Atual	Próximo		
1	<i>Idle</i>	<i>Assembling_Dgram</i>	Camada TF repassa segmento para envio	Aguardar montagem de datagrama
2	<i>Idle</i>	<i>Receiving_Dgram</i>	Recebimento de datagrama vindo da rede	Aguardar término de transmissão
3	<i>Assembling_Dgram</i>	<i>Wait_ACK</i>	Datagrama devidamente montado	Aguarda retorno de ACK ou NACK
4	<i>Wait_ACK</i>	<i>Idle</i>	Recebido ACK ou NACK do destinatário	Notificar camada superior do recebimento
5	<i>Receiving_Dgram</i>	<i>Assembling_Seg</i>	Término da recepção do datagrama	Aguardar montagem de segmento
6	<i>Assembling_Seg</i>	<i>Idle</i>	Segmento devidamente montado e datagrama ACK pronto para envio	Repassar segmento a camada superior e enviar ACK
7	<i>Assembling_Seg</i>	<i>Idle</i>	Datagrama contém erros de transmissão e datagrama NACK pronto para envio	Enviar NACK

Fonte: Elaborado pelo próprio autor.

Um exemplo de transmissão de dados entre os dispositivos, *Master* e *Outstation*, e suas camadas é ilustrado na Figura 12.

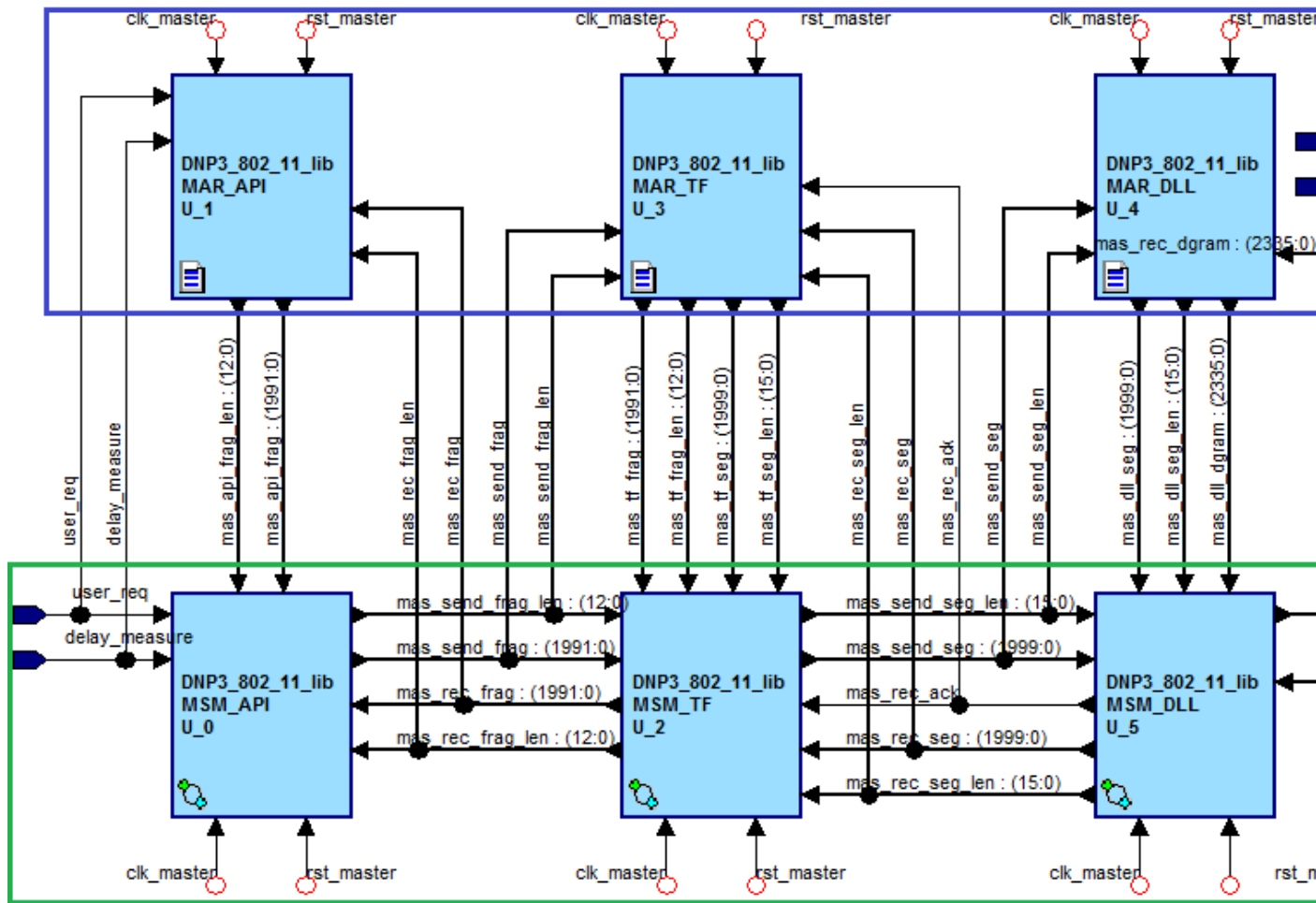
Figura 12 - Exemplo de Comunicação entre dispositivos DNP3.



Fonte: (IEEE, 2012).

Na Figura 13 ilustra-se a topologia de modelagem do dispositivo DNP3 *Master* com suas respectivas entidades em VHDL. A topologia referente ao dispositivo *Outstation* é semelhante à ilustrada. A modelagem realizada foi simulada na Figura 13, que ilustra uma representação simulada de uma troca de mensagens entre os dispositivos *Master* e *Outstation*, cujo numeração é descrita na Tabela 12.

Figura 13 - Topologia de entidades do dispositivo DNP3 Master em VHDL.



Fonte: Elaborado pelo próprio autor.

Figura 14 – Representação de comunicação simulada através de mudanças de estado DNP3

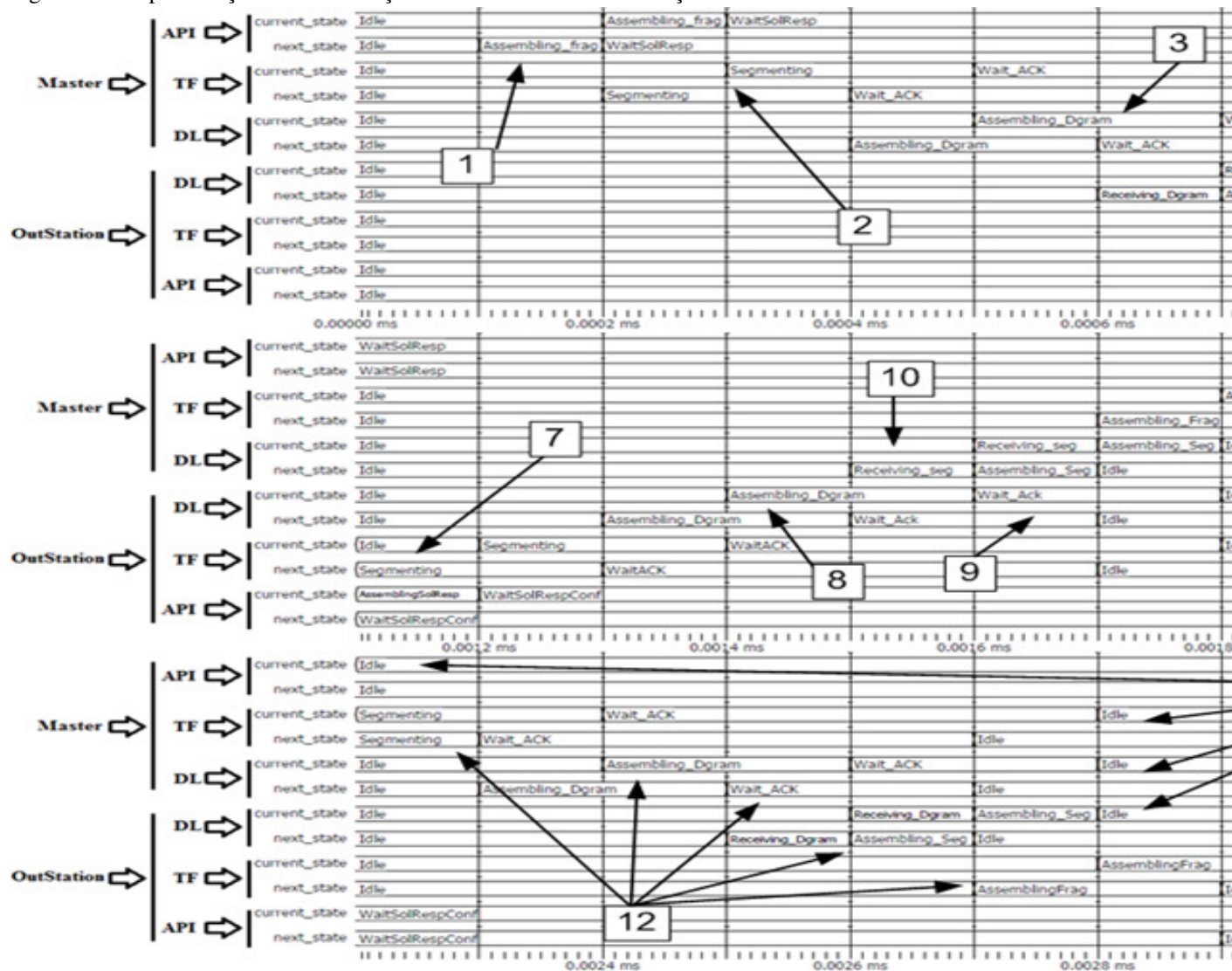


Tabela 12 – Descrição de eventos da comunicação DNP3 entre *Master* e *Outstation*

Eventos	Transição de estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>Assembling_Frag</i>	<i>Master</i> API envia fragmento com comando de Código de Função API x"17" e de Aplicação x"C1" que indica que o fragmento é único e não necessita de confirmação API
2	<i>Idle</i>	<i>Segmenting</i>	<i>Master</i> TF envia segmento cujo Cabeçalho de Transporte x"C1" indica que o segmento é único
3	<i>Idle</i>	<i>Assembling_Dgram</i>	<i>Master</i> DL envia datagrama cujo Código de Função de Enlace x"F3" indica que é um datagrama de dados primário de direção <i>Master</i> para <i>OutStation</i>
4	<i>Wait_ACK</i>	<i>Idle</i>	<i>Master</i> DL recebe ACK vindo do <i>Outstation</i>
5	<i>Receiving_Dgram</i>	<i>Assembling_Seg</i>	<i>Outstation</i> remove os cabeçalhos DL e TF e repassa o fragmento à camada API
	<i>Idle</i>	<i>Assembling_Frag</i>	
6	<i>Idle</i>	<i>Assembling_SolResp</i>	<i>Outstation</i> API recebe o comando e monta fragmento de resposta com comando de Código de Função API x"81" e Controle de Aplicação x"E1" que indica que o fragmento é único e necessita de confirmação API
7	<i>Idle</i>	<i>Segmenting</i>	<i>Outstation</i> TF envia segmento cujo Cabeçalho de Transporte x"C1" indica que o segmento é único
8	<i>Idle</i>	<i>Assembling_Dgram</i>	<i>Outstation</i> DL envia datagrama cujo Código de Função de Enlace x"03" indica que é um datagrama de dados secundário de direção <i>Outstation</i> para <i>Master</i>
9	<i>Wait_ACK</i>	<i>Idle</i>	<i>Outstation</i> DL recebe ACK vindo do <i>Master</i>
10	<i>Receiving_Dgram</i>	<i>Assembling_Seg</i>	<i>Master</i> remove os cabeçalhos DL e TF e repassa o fragmento à camada API
	<i>Idle</i>	<i>Assembling_Frag</i>	
11	<i>Wait_SolResp</i>	<i>Assembling_Frag</i>	<i>Master</i> API recebe resposta ao comando e monta fragmento de confirmação
12	----	----	Procedimento semelhante aos eventos 2 a 5
13	----	<i>Idle</i>	Troca de mensagens se finaliza quando todas as camadas retornam ao estado <i>Idle</i>

A simulação realizada anteriormente valida a modelagem realizada do protocolo DNP3. Entretanto para representação de um cenário SG, esta modelagem deve ser compatível para integração com outros protocolos. No Capítulo 4 efetua-se uma descrição geral de algumas tecnologias de transmissão que podem ser utilizadas. Além disso, descreve-se e modela-se em VHDL os protocolos IEEE 802.11b e IEEE 802.15.4, de forma a apresentar suas características de acesso ao meio de transmissão e, quando integradas ao protocolo de rede distribuída, como elas afetam a latência do enlace entre dois *Smart Meters* DNP3.

4 INTERFACES DE COMUNICAÇÃO

A comunicação do fluxo de dados SG pode ser realizada por diversos tipos de interfaces. Na Tabela 13 descreve-se características relacionadas a algumas tecnologias de interface sem fio que podem ser utilizadas em SG. A Figura 15 traça um comparativo dessas tecnologias em relação ao consumo de energia, taxa de dados e alcance.

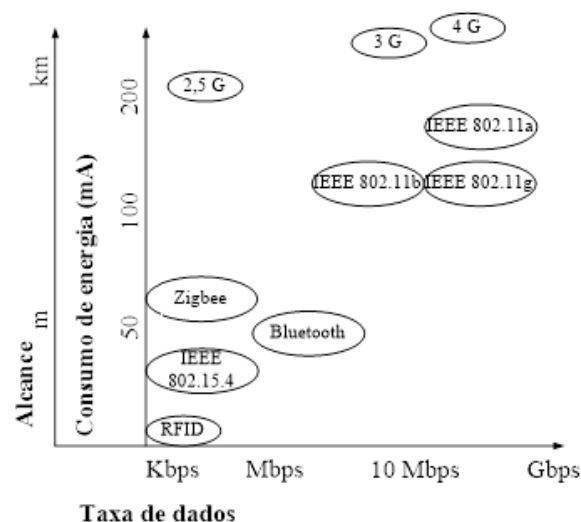
Tabela 13 - Comparativo entre interfaces de comunicação.

Características	Interfaces de Comunicação						
	802.11	Bluetooth	802.15.4	Zigbee	2,5 G GPRS	3G HSPA	4G WiMax
Alcance (m)	centenas	dezenas	mil	mil	milhares	milhares	Milhares
Taxa de Transmissão	11, 54 ou 600 Mbps	1, 2 ou 3 Mbps	250 Kbps	250 Kbps	170 Kbps	14 Mbps	1 Gbps
Segurança	WPA2 RSA	Shared Secret	AES 128 bits	AES 128 bits	GEA4	AES TKIP WPA	AES 3DES RSA
Número de nós	dezenas	dezenas	milhares	milhares	milhares	milhares	milhares
Tamanho de Datagrama	Até 64 kbytes	Até 64 kbytes	Até 133 bytes	Até 133 bytes	Até 64 kbytes	Até 64 kbytes	Até 64 kbytes
Aplicação SG	Leitura AMI, Teleproteção	Leitura AMI, V2G	Leitura AMI, BAN, NAN	Leitura AMI, BAN, NAN	V2G, NAN, WAN	V2G, NAN, WAN	V2G, NAN, WAN

*As informações de parâmetros específicos variam de acordo com modelo e fabricante do produto em questão.

Fonte: (ARAVITHAN et al., 2011; FADLULLAH et al., 2011; PARIKH et al., 2010; SHAH et al., 2008).

Figura 15 – Comparativo entre tecnologias *wireless*.



Fonte: (SANTOS, 2003).

A seção 4.1 detalha o protocolo IEEE 802.11b modelado neste trabalho. A seção 4.2 descreve a integração IEEE 1815/802.11. As seções 4.3 e 4.4 apresentam a modelagem e integração do IEEE 802.15.4.

4.1 Modelagem IEEE 802.11b

A pilha de protocolo IEEE 802.11 define as camadas MAC (*Media Access Control*) e PHY (*Physical*). As redes 802.11 podem ser constituídas em modo infra estruturado, com a presença de ponto de acesso, ou em modo *Ad Hoc* (ANTUNES, 2012). A fim de representar uma comunicação P2P, neste trabalho modelou-se uma comunicação Ad Hoc entre duas estações.

Ao receber dados para envio, a camada MAC insere 32 octetos de cabeçalho e 4 octetos de verificação de erro, conforme descritos na Tabela 14. A entidade *SAR_MAC* modelada em VHDL na montagem de quadros MAC 802.11 considerou somente a construção dos quadros RTS (*Request to Send*), CTS (*Clear to Send*), ACK (*Acknowledgment*) e Dados. Portanto, a simulação ocorre entre estações IEEE 802.11 já associadas e sincronizadas em rede.

Os bits do octeto *Frame Control* estão descritos nas Tabelas 15 e 16.

Tabela 14 – Quadro MAC do protocolo IEEE 802.11.

Campo	Tamanho	Descrição
<i>Frame Control</i>	2 octetos	Campo responsável por trazer informações relevantes à interpretação do mesmo pela camada MAC.
<i>Duration/ID</i>	2 octetos	Quando usado como duração, traz o tempo que toda a comunicação ocupará. Assim as estações não envolvidas na troca de mensagens possam entrar em período NAV (<i>Network Allocation Vector</i>). Pode atuar como identificador de conexão.
<i>Address 1</i>	6 octetos	O endereçamento deste campo pode variar entre BSSID (<i>Basic Service Set Identification</i>), SA (<i>Source Address</i>), DA (<i>Destination Address</i>), TA (<i>Transmitter Address</i>) ou RA (<i>Receiver Address</i>), dependendo das informações dentro do Campo <i>Frame Control</i> .
<i>Address 2</i>	6 octetos	O endereçamento deste campo pode variar entre BSSID, SA, DA, TA ou RA, dependendo das informações dentro do Campo <i>Frame Control</i> .
<i>Address 3</i>	6 octetos	O endereçamento deste campo pode variar entre BSSID, SA, DA, TA ou RA, dependendo das informações dentro do Campo <i>Frame Control</i> .
<i>Sequence Control</i>	2 octetos	Informa o número de sequência dos quadros transmitidos entre dois nós da rede, além do número do fragmento caso necessite de remontagem de dados.
<i>Address 4</i>	6 octetos	O endereçamento deste campo pode variar entre BSSID, SA, DA, TA ou RA, dependendo das informações dentro do Campo <i>Frame Control</i> .
<i>QoS Control</i>	2 octetos	Define configurações de QoS (<i>Quality of Service</i>). Campo opcional de acordo com as informações dentro do <i>Frame Control</i> .
<i>Payload</i>	0-23124 octetos	Espaço para alocação de dados de camadas superiores.
CRC 32 bits	4 octetos	Sequência de verificação de erros

Fonte: (IEEE, 2007; ARAÚJO, 2013).

Tabela 15 – Descrição de bits *Frame Control* da camada MAC 802.11.

Bits	Nome	Descrição
15-14	<i>Protocol Version</i>	Indica versão do protocolo 802.11
13-12	<i>Type</i>	Determina se o quadro é do tipo gerenciamento, controle ou dados. Maiores detalhes descritos na Tabela 16.
11-8	<i>Subtype</i>	Determina subtipo do quadro em função do seu do tipo. Maiores detalhes descritos na Tabela 16.
7	<i>To DS</i>	Esses campos são utilizados apenas por quadros do tipo dados. Indicam se os dados contidos são destinados ao sistema distribuído ou são originados no sistema distribuído.
6	<i>From DS</i>	
5	<i>More Frag</i>	Indica se existem mais quadros de dados ou gerenciamento a serem recebidos da rede.
4	<i>Retry</i>	Indica se o quadro recebido foi retransmitido.
3	<i>Power Mgt</i>	Utilizado apenas pelo AP (<i>Access Point</i>) para desativar modo de espera do receptor.
2	<i>More Data</i>	Indica se existem mais quadros de dados a serem transmitidos.
1	<i>Protected Frame</i>	Indica se este é um quadro protegido por criptografia.
0	<i>Order</i>	Indica se a montagem de manter a ordem de recebimento dos dados.

Fonte: (IEEE, 2007; ARAÚJO, 2013).

Tabela 16 – Descrição dos campos *Type* e *Subtype* da camada MAC 802.11 que serão utilizados no trabalho.

Type (13-12)	Descrição	Subtype (11-8)	Descrição
01	Gerenciamento	1011	RTS
01	Gerenciamento	1100	CTS
01	Gerenciamento	1101	ACK
10	Dados	0000	Dados

Fonte: (IEEE, 2007; ARAÚJO, 2013).

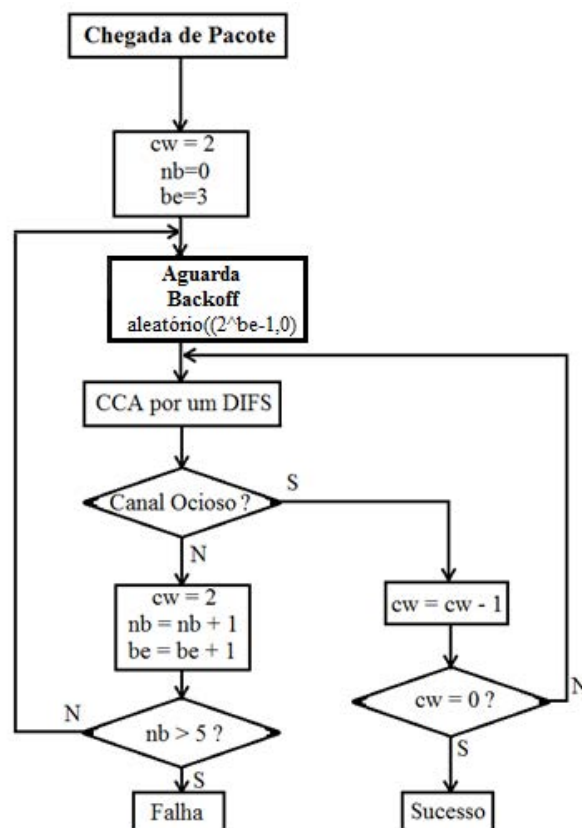
Além de uma entidade responsável pela montagem dos quadros, *SAR_MAC*, foi modelada em VHDL a entidade arquitetura *SAR_Backoff* encarregada da disputa de acesso ao meio prevista pela DCF (*Distributed Coordination Function*) MAC IEEE 802.11, que possibilita o controle distribuído do acesso ao meio compartilhado (ANTUNES, 2012). Esta camada é regida pela utilização de uma série de funções que em conjunto organizam o acesso ao enlace sem fio.

Neste modo quando um nó da rede deseja efetuar uma transmissão, é feita uma verificação de assinatura no canal RF (*Radio Frequency*) através da função CCA (*Clear Channel Assignment*). Se o canal de frequência estiver ocioso, então o nó pode efetuar a transmissão dos dados sem ativar o algoritmo *slotted CSMA/CA* (*Carrier Sense Multiple Access with Collision Avoidance*), ilustrado na Figura 16. Caso o canal esteja ocupado, o nó que deseja enviar dados deverá ativar o algoritmo *slotted* para esperar um período aleatório de *timeslots* chamados *backoffs* para efetuar a varredura da assinatura de canal novamente.

Com o término do período *backoff*, a função CCA é ativada novamente para verificação da assinatura do canal. Se o canal estiver ocioso, então a transmissão ocorre. Caso o canal

esteja ocupado, há um incremento nos parâmetros nb e be para que seja executado novamente o *backoff*. Quando o parâmetro nb for maior que 5, a transmissão é abortada. Através de *Backoffs*, as estações aguardam determinado espaço de tempo aleatório para realizar a varredura do canal. Para que uma transmissão seja autorizada, a função CCA deverá apontar canal livre durante dois períodos DIFS (*Distributed Inter Frame Space*) de tempo. Além do DIFS, a camada MAC ainda tem outro período de tempo mais curto, SIFS (*Short Inter Frame Space*), que define prioridades de envio de tipos de pacotes, como CTS (*Clear to Send*), RTS (*Request to Send*) e ACK (*Acknowledgment*) (AAD; CASTELLUCCIA, 2001).

Figura 16 – Algoritmo *slotted* CSMA/CA.



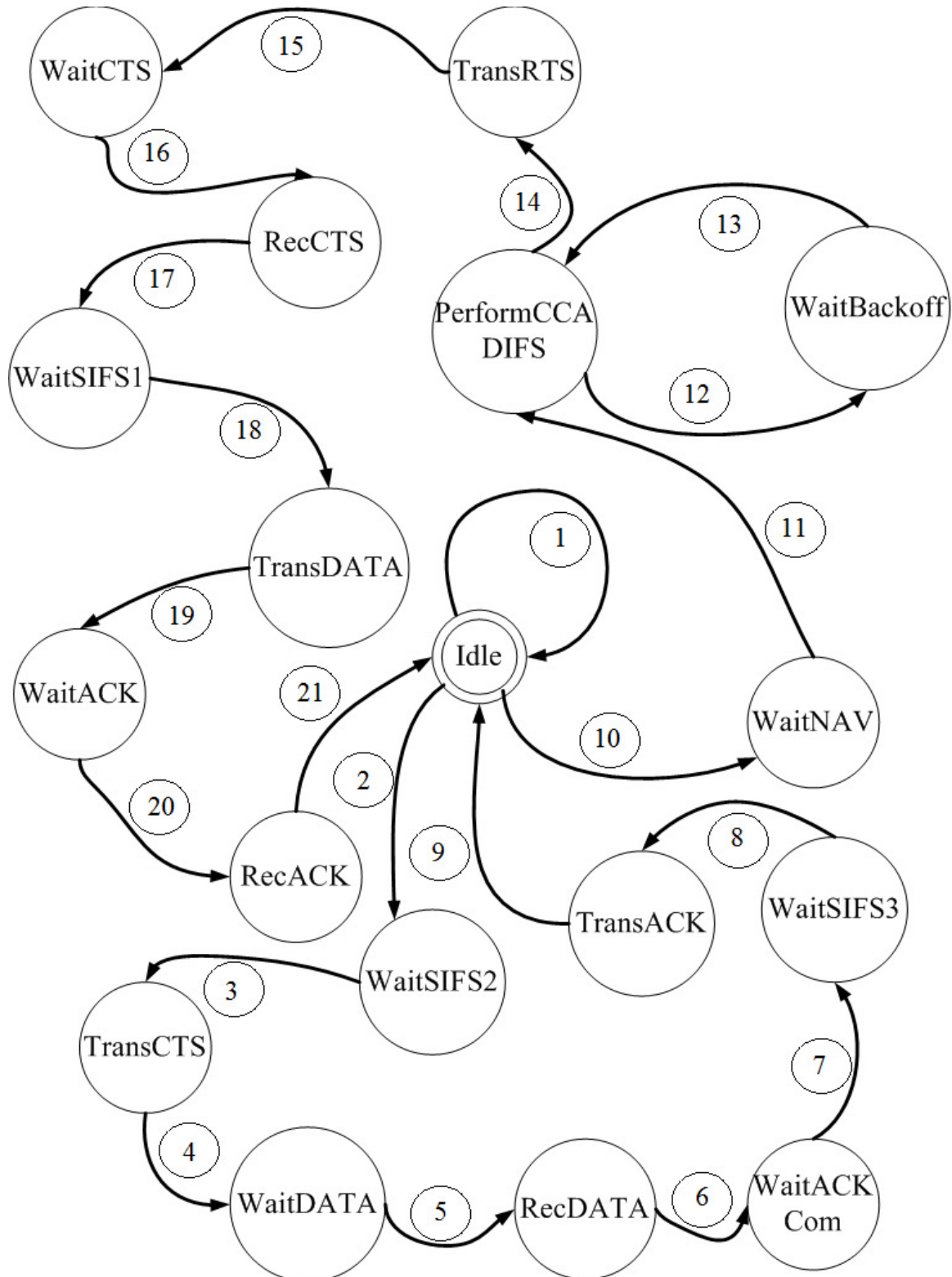
Fonte: (JING; AINDA, 2011).

Além das entidades IEEE 802.11 apresentadas anteriormente, sendo uma responsável pela montagem de quadros e outra pelo controle de acesso ao meio, a máquina de estado modelada para controlar o comportamento da camada MAC IEEE 802.11 e suas funcionalidades é ilustrada pela Figura 17. As transições de estado estão descritas na Tabela 17.

A camada PHY 802.11b modelada no trabalho é responsável por fazer o controle de velocidade de transmissão de dados (11 Mbps) (IEEE, 2007), montagem de cabeçalho e

preamble, descrito na Tabela 18, e auxílio na utilização da função CCA. Na Figura 18 ilustra-se a topologia de entidades de uma das estações IEEE 802.11b modeladas.

Figura 17 – Máquinas de estado da camada MAC IEEE 802.11.



Fonte: (MURTHY; MANOJ, 2004).

Tabela 17 – Transições de Estados camada MAC IEEE 802.11.

Transição n°	Transições		Descrição	Ações
	Atual	Próximo		
1	<i>Idle</i>	<i>Idle</i>	Recebe quadro MAC não endereçado à estação	Repassar valor do campo <i>Duration/ID</i> para entidade de controle ao acesso
2	<i>Idle</i>	<i>WaitSIFS2</i>	Recebe quadro MAC RTS endereçado à estação	Aguardar período SIFS (28 us) de tempo e montagem do quadro CTS
3	<i>WaitSIFS2</i>	<i>TransCTS</i>	Fim do período SIFS e quadro CTS devidamente montado	Repassar quadro para camada inferior e aguardar fim de sua transmissão
4	<i>TransCTS</i>	<i>WaitDATA</i>	Fim da transmissão do quadro CTS	Aguardar início do recebimento do quadro de dados
5	<i>WaitDATA</i>	<i>RecDATA</i>	Início do recebimento do quadro de Dados	Aguardar término do recebimento do quadro de dados
6	<i>RecDATA</i>	<i>WaitACKCom</i>	Término do recebimento do quadro de dados	Repassar <i>payload</i> para camadas superiores para verificação de erro e aguardar ordem de envio de ACK
7	<i>WaitACKCom</i>	<i>WaitSIFS3</i>	Recebe ordem de envio de ACK	Aguardar período SIFS de tempo e montagem do quadro ACK
8	<i>WaitSIFS3</i>	<i>TransACK</i>	Fim do período SIFS e quadro ACK devidamente montado	Repassar quadro para camada inferior e aguardar fim de sua transmissão
9	<i>TransACK</i>	<i>Idle</i>	Fim da transmissão do quadro ACK	-----
10	<i>Idle</i>	<i>WaitNAV</i>	Recebe dados das camadas superiores para serem enviados	Aguardar período sem contenção (NAV) caso haja
11	<i>WaitNAV</i>	<i>PerformCCADIFS</i>	Término ou inexistência de período NAV	Acionar função CCA durante dois períodos DIFS (50 us cada)
12	<i>PerformCCADIFS</i>	<i>WaitBackoff</i>	Detectada atividade no canal	Aguardar período <i>Backoff</i>
13	<i>WaitBackoff</i>	<i>PerformCCADIFS</i>	Fim do período <i>Backoff</i>	Acionar função CCA durante dois períodos DIFS (50 us cada)
14	<i>PerformCCADIFS</i>	<i>TransRTS</i>	Não detectada atividade no canal	Repassar quadro RTS para camada inferior e aguardar fim de sua transmissão
15	<i>TransRTS</i>	<i>WaitCTS</i>	Fim da transmissão do quadro RTS	Aguardar início do recebimento do quadro CTS
16	<i>WaitCTS</i>	<i>RecCTS</i>	Início do recebimento do quadro CTS	Aguardar término do recebimento do quadro CTS
17	<i>RecCTS</i>	<i>WaitSIFS1</i>	Término do recebimento do quadro CTS	Aguardar período SIFS de tempo e montagem do quadro de dados
18	<i>WaitSIFS1</i>	<i>TransDATA</i>	Fim do período SIFS e quadro de dados devidamente montado	Repassar quadro para camada inferior e aguardar fim de sua transmissão
19	<i>TransDATA</i>	<i>WaitACK</i>	Fim da transmissão do quadro de dados	Aguardar início do recebimento do quadro ACK
20	<i>WaitACK</i>	<i>RecACK</i>	Início do recebimento do quadro ACK	Aguardar término do recebimento do quadro ACK
21	<i>RecACK</i>	<i>Idle</i>	Término do recebimento do quadro ACK	-----

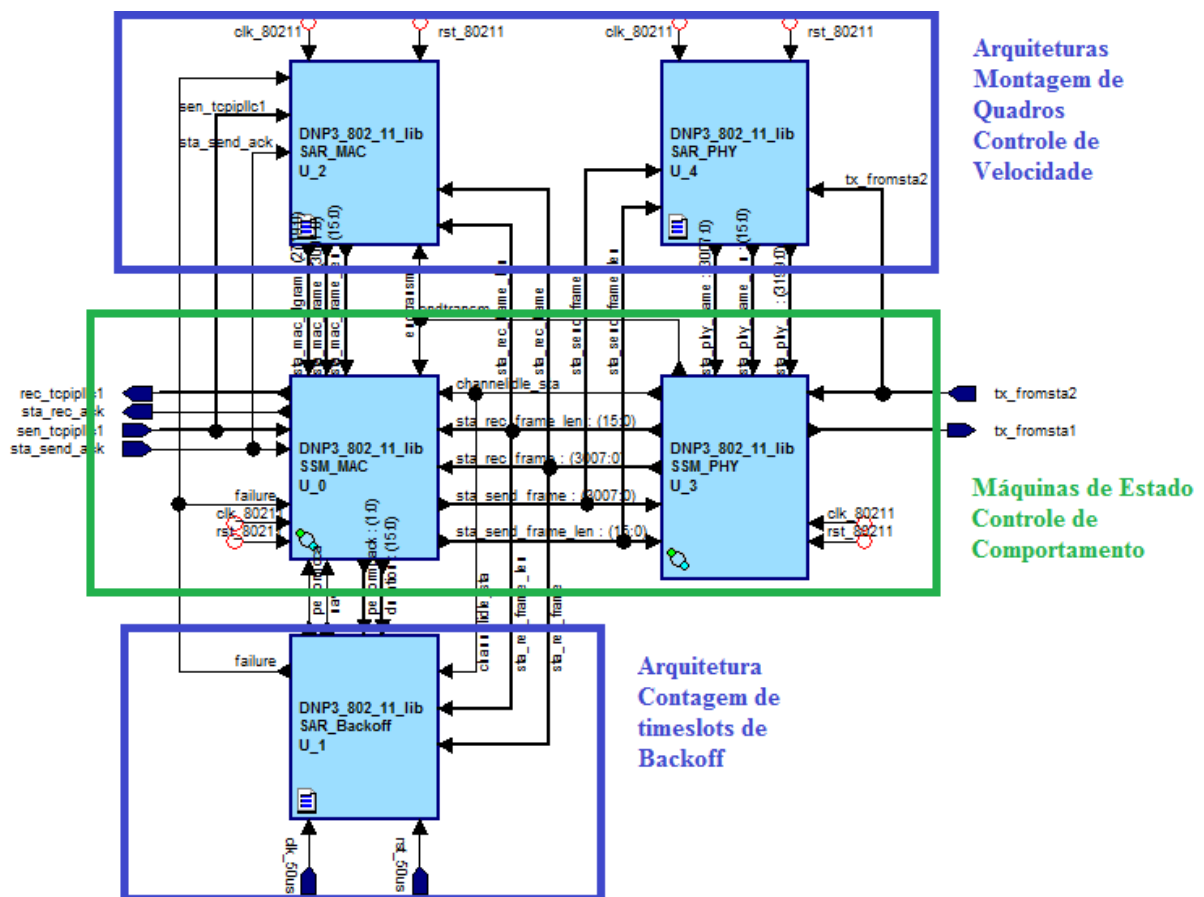
Fonte: Elaborado pelo próprio autor.

Tabela 18 – Quadro PHY do protocolo IEEE 802.11.

Campo	Tamanho	Descrição
<i>Synchronization</i>	16 octetos	Sequência de zeros que antecedem o início do quadro.
<i>SFD</i>	2 octetos	<i>Start Frame Delimiter</i> representa os octetos que indicam início do quadro.
<i>Signal</i>	1 octeto	Indica qual velocidade a transmissão deve ocorrer
<i>Service</i>	1 octeto	Indica formato do quadro PHY, neste trabalho sempre configurado como x"01" que indica que o campo <i>Lenght</i> é tempo em microssegundos da transmissão desse único quadro.
<i>Lenght</i>	2 octetos	Indica o tempo em microssegundos da transmissão desse único quadro.
<i>Payload</i>	-----	Dados provenientes das camadas superiores.

Fonte: (IEEE, 2007; ARAÚJO, 2013).

Figura 18 – Estação 802.11b modelada.



Fonte: Elaborado pelo próprio autor.

4.2 Integração IEEE 1815/802.11b

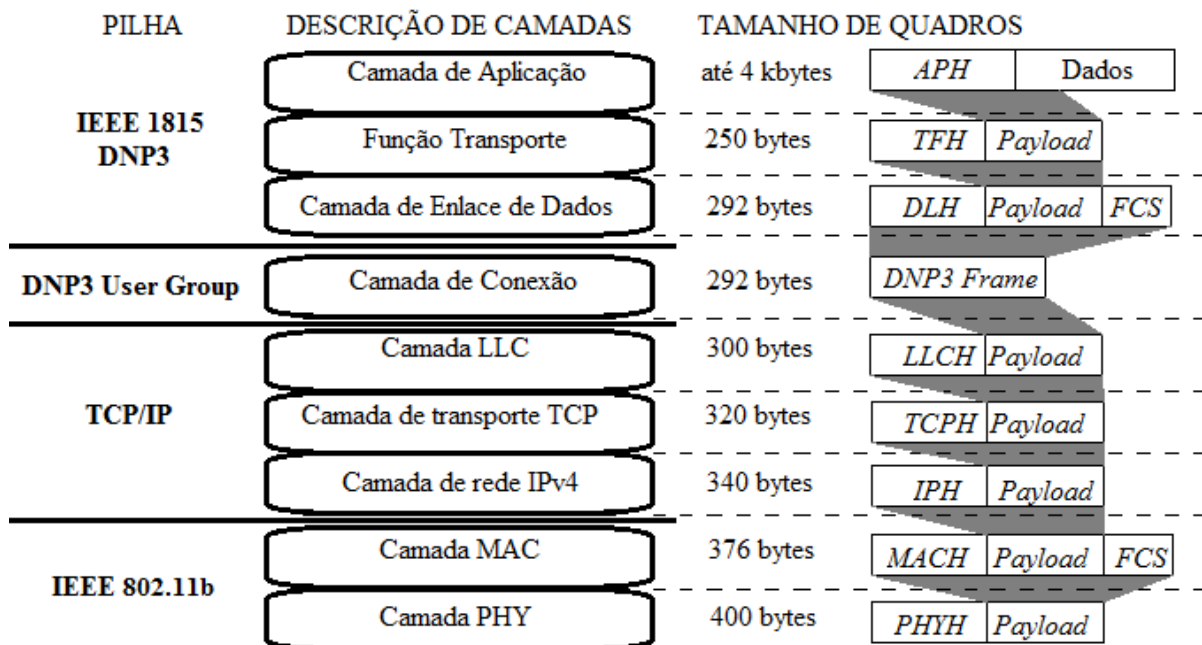
Nesta seção analisa-se a modelagem realizada passando pelas camadas de cada protocolo utilizado no trabalho. Para possibilitar a comunicação dos dados SG pela internet, os quadros DNP3 devem ser encapsulados também nos protocolos da pilha TCP/IP. Trata-se dos protocolos LLC (*Logical Link Control*) (IEEE, 1985), TCP (*Transport Communication*

Protocol) (REQUEST FOR COMMENTS - RFC, 1981) e IP (*Internet Protocol*) (RFC, 1981). Somente depois de passar por essas camadas os quadros DNP3 entram em processo de encapsulamento no protocolo sem fio IEEE 802.11b, conforme ilustrado na Figura 19.

Apesar de considerar, a quantidade de bytes dos cabeçalhos dos protocolos, LLC, TCP, IP, necessários à comunicação em rede TCP/IP, neste trabalho não se modelam os comportamentos dos mesmos na íntegra. Na modelagem descrita a seguir focou-se em representar as camadas dos protocolos IEEE 1815 DNP3 e IEEE 802.11 principalmente.

Segundo a recomendação do DNP3 (IEEE, 2012), a integração deve desabilitar o sistema de confirmação da camada de enlace de dados do DNP3, ou seja, o ACK. Essa funcionalidade está sob responsabilidade do protocolo IEEE 802.11. Além disso, a pilha de protocolo DNP3 não deve ser alterada.

Figura 19 – Pilha IEEE 1815/802.11b.



Fonte: (IEEE, 2012).

4.3 Modelagem IEEE 802.15.4

A pilha do protocolo IEEE 802.15.4 (IEEE, 2011b), define as camadas de acesso ao meio (MAC) e física (PHY). Apesar destas camadas serem utilizadas no protocolo Zigbee, este implementa um sistema de roteamento e segurança de forma nativa. A modelagem e documentação deste trabalho tem seu foco apenas na pilha IEEE 802.15.4 devido à topologia

desejada não necessitar de roteamento. Em trabalho futuros pretende-se utilizar o protocolo Zigbee em outros experimentos.

A camada MAC IEEE 802.15.4, que em uma comunicação ponto a ponto entre dispositivo *Coordenador* e *End Device*, insere 13 octetos de cabeçalho e 2 octetos de verificação de erro, conforme descreve-se na Tabela 19. O Campo *Frame Control* está descrito na Tabela 20. A montagem de quadros MAC foi modeladas na entidades *MACCRD* e *MACED*.

Tabela 19 – Quadro MAC do protocolo IEEE 802.15.4.

Campo	Tamanho	Descrição
<i>Frame Control</i>	2 octetos	Campo responsável por trazer informações relevantes à interpretação do mesmo pela camada MAC.
<i>Sequence</i>	1 octetos	Define a numeração do quadro recebido
<i>Address 16 bits</i>	2 octetos	Endereçamento dinâmico de estação de origem ou destino de dados associada à rede.
<i>Address 64 bits</i>	8 octetos	Endereçamento fixo de estação de origem ou destino de dados.
<i>Payload</i>	Até 103 octetos	Dados provenientes de camadas superiores.
<i>CRC 32 bits</i>	2 octetos	Verificação cíclica de erro.

Fonte: (IEEE, 2011b).

Tabela 20 – Descrição de bits *Frame Control* da camada MAC 802.15.4.

Bits	Nome	Descrição
15-13	<i>Frame Type</i>	Indica se o quadro é do tipo comando MAC '011', ACK '010' ou dados '001'.
12	<i>Security Enable</i>	Indica se o quadro está sob proteção de segurança através de criptografia.
11	<i>Frame Pending</i>	Indica se a origem do quadro possui mais dados a enviar.
10	<i>AR</i>	Indica se o quadro necessita de confirmação ACK do receptor.
9	<i>PAN ID Compression</i>	Indica a presença de campo de identificação da PAN (<i>Personal Area Network</i>), neste trabalho sempre configurado com valor '0'.
8-6	<i>Reserved</i>	-----
5-4	<i>Dest. Address Mode</i>	Indica quais endereços do destino o quadro possui. As possibilidades são utilizar ou não endereçamento 16 bits e 64 bits.
3-2	<i>Frame Version</i>	Indica versão do protocolo 802.15.4 utilizada.
1-0	<i>Source Address Mode</i>	Indica quais endereços da origem o quadro possui. As possibilidades são utilizar ou não endereçamento 16 bits e 64 bits.

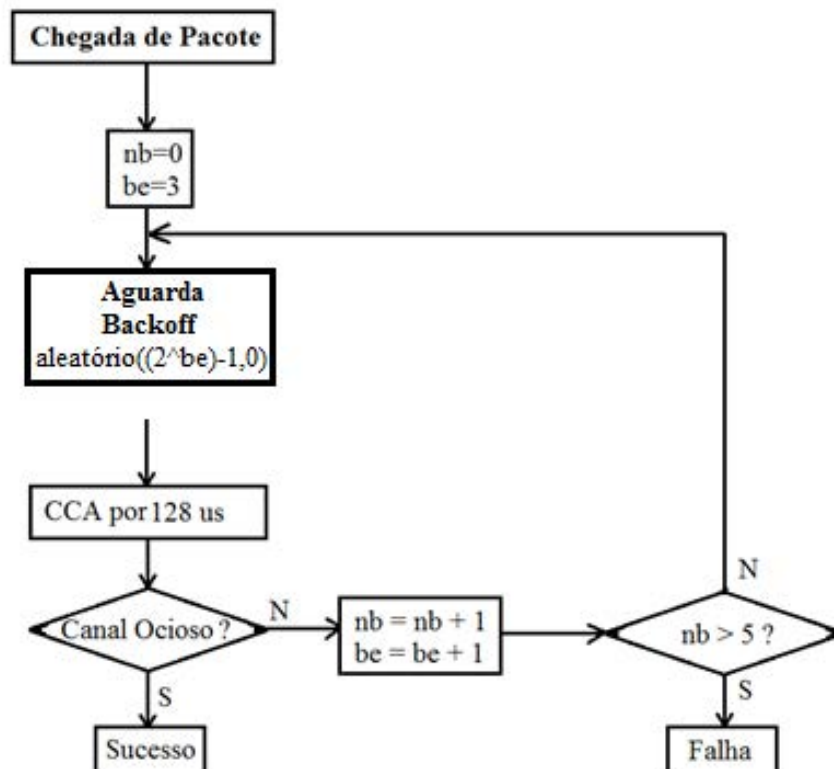
Fonte: (IEEE, 2011b).

Considerando que a topologia de rede IEEE 802.15.4 utilizada neste trabalho foi de comunicação ponto-a-ponto entre um nó coordenador e um *end device*, o sistema de endereçamento contém apenas o endereço do *end device*. Ou seja, quando o coordenador desejar enviar dados, o quadro contém os endereços 16 bits e 64 bits de destino do dispositivo *end device*. Quando o sentido da comunicação for contrária o quadro informará o endereço 16 bits e 64 bits de origem do *end device*.

A camada de MAC (*Media Access Control*) utiliza uma série de funções para evitar que ocorram colisões nas transmissões realizadas através do meio compartilhado de Rádio

Frequência. Essas funções de acesso ao meio estão modelados nas entidades *BACKOFFCRD* e *BACKOFFED*. O protocolo define dois modos de operação para controle de acesso, sendo o primeiro de forma centralizada através de *beacons* enviados pelo coordenador da rede, e o segundo de forma distribuída através da função *unslotted CSMA/CA* (*Carrier Sense Multiple Access with Collision Avoidance*), cujo algoritmo está ilustrado na Figura 20. Este trabalho efetuou a modelagem da camada MAC com modo de controle de acesso distribuído.

Figura 20 - Algoritmo *unslotted CSMA/CA*.



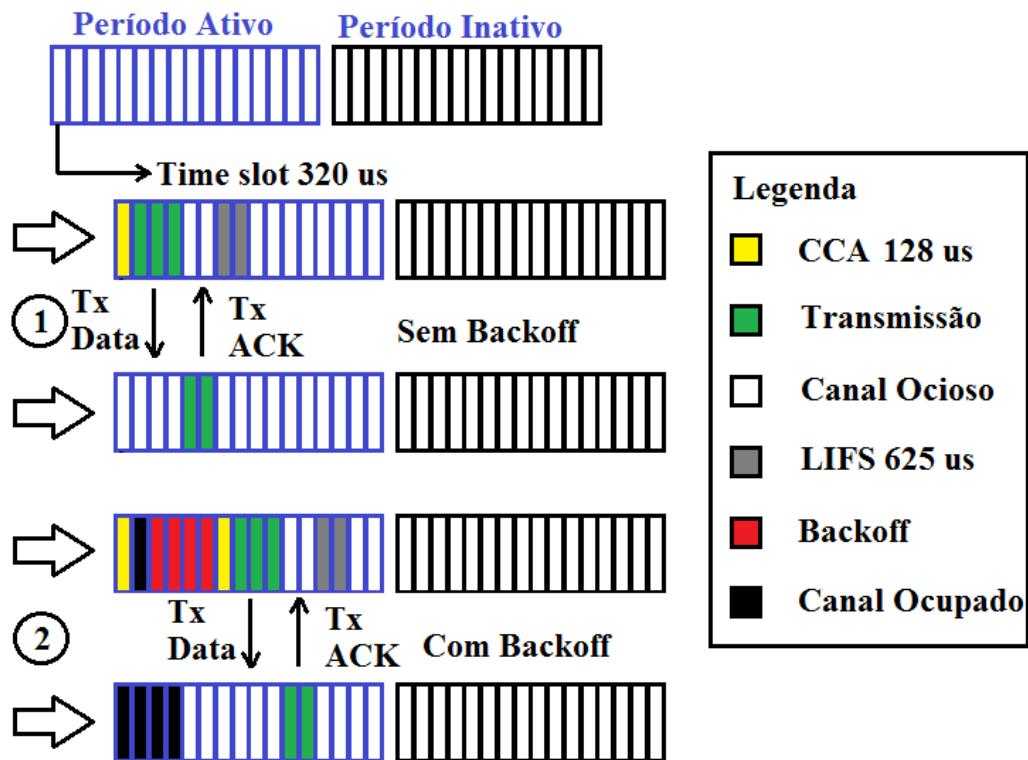
Fonte: (BURATTI; VERDONE, 2009).

Nesse modo quando um nó da rede deseja efetuar uma transmissão, este faz uma verificação de assinatura do canal RF durante 128 *us* através da função CCA (*Clear Channel Assignment*). Se o canal de frequência está ocioso, então o nó pode efetuar a transmissão dos dados sem ativar o algoritmo CSMA/CA. Caso o canal esteja sendo utilizado, o nó que deseja enviar dados deverá ativar o algoritmo para esperar um período aleatório de *timeslots* (320 *us*) chamados *backoff* para varreruda de canal novamente.

Com o término do período *backoff*, a função CCA é ativada novamente e verifica a assinatura do canal. Se o canal estiver ocioso, então a transmissão ocorre. Caso o canal esteja ocupado, há um incremento nos parâmetros *nb* e *be* para que seja executado novamente o *backoff*. Quando o parâmetro *nb* for maior que 5, a transmissão é abortada.

Além das funções CSMA/CA e CCA, a camada MAC estabelece períodos de disputa pelo acesso ao meio. Na Figura 21 apresenta-se o funcionamento das transmissões no protocolo IEEE 802.15.4. No evento da transmissão 1, dois nós da rede IEEE 802.15.4 participam de uma troca de mensagens sem necessidade de utilização de *backoff*. Primeiramente é realizado uma varredura de assinatura do canal, através da função CCA, durante um período de 128 *us* (KOHVAKKA et al., 2006). Como o canal está inativo, a transmissão é realizada. O evento da transmissão 2 representa uma situação de tentativa de envio com ocupação do canal, ocasionando assim, a utilização dos *timeslots* aleatórios. Cada nó que deseje transmitir deve efetuar o processo de disputa apenas no período de atividade CAP (*Contention Active Period*). Entretanto, uma vez que consiga acesso ao meio, a transmissão pode utilizar os *timeslots* do período de inatividade. O nó que efetuar uma transmissão deve esperar por um período LIFS de 625 *us* (*Long Inter Frame Spacing*) durante período ativo de disputa.

Figura 21 - Comunicação protocolo IEEE 802.15.4 com disputa de acesso ao meio.

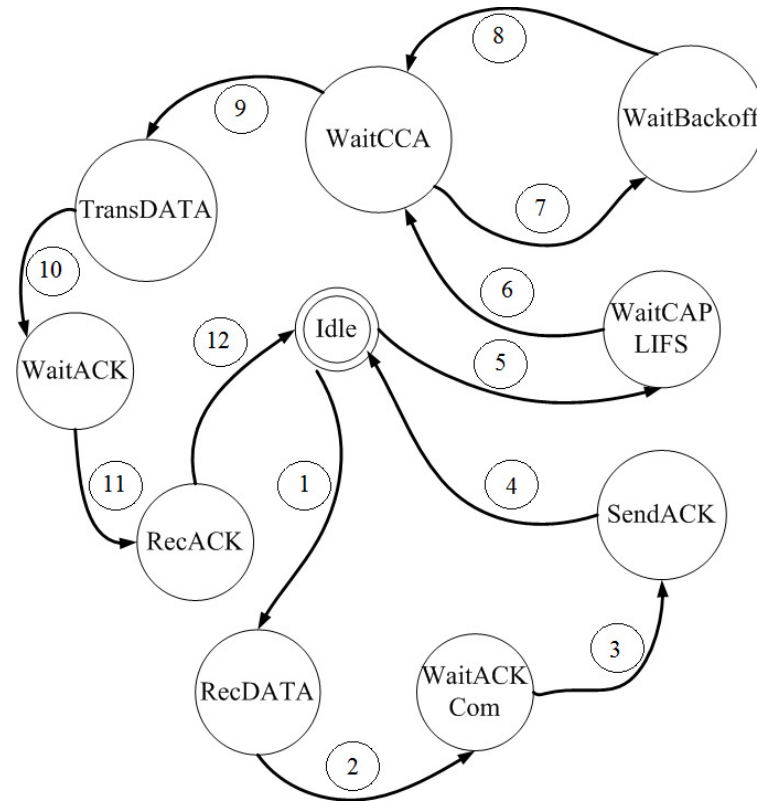


Legenda: 1- Transmissão sem backoff, 2- Transmissão com backoff.

Fonte: (KOHVAKKA et al., 2006).

Por fim, a camada MAC IEEE 802.15.4 foi modelada em uma entidade de arquitetura do tipo máquina de estado (*MAC802154_Coord* ou *MAC802154_Ed*), que tem a função de controlar o comportamento do protocolo, conforme ilustra a Figura 22 e descrita na Tabela 21.

Figura 22 – Máquinas de estado da camada MAC IEEE 802.15.4.



Fonte: Elaborado pelo próprio autor.

Tabela 21 – Transições de Estados camada MAC IEEE 802.15.4.

Transição n°	Transições		Descrição	Ações
	Atual	Próximo		
1	<i>Idle</i>	<i>RecDATA</i>	Início do recebimento de dados endereçados à estação	Aguardar término de recebimento
2	<i>RecDATA</i>	<i>WaitACKCom</i>	Término de recebimento de quadro de dados	Repassar <i>Payload</i> para as camadas superiores e aguardar ordem de envio de ACK
3	<i>WaitACKCom</i>	<i>SendACK</i>	Recebe ordem para envio de ACK	Repassar quadro ACK para a camada inferior e aguardar fim de sua transmissão
4	<i>SendACK</i>	<i>Idle</i>	Fim da Transmissão do quadro ACK	-----
5	<i>Idle</i>	<i>WaitCAPLIFS</i>	Recebe dados das camadas superiores para serem enviados	Aguardar CAP (<i>Contention Active Period</i>) e período LIFS caso necessário
6	<i>WaitCAPLIFS</i>	<i>WaitCCA</i>	Período CAP ativo e término ou inexistência de período LIFS	Acionar função CCA durante 128 <i>us</i>
7	<i>WaitCCA</i>	<i>WaitBackoff</i>	Detectada atividade no canal	Aguardar período <i>Backoff</i>
8	<i>WaitBackoff</i>	<i>WaitCCA</i>	Fim do período <i>Backoff</i>	Acionar função CCA durante 128 <i>us</i>
9	<i>WaitCCA</i>	<i>TransDATA</i>	Não detectada atividade no canal	Repassar quadro de dados para a camada inferior e aguardar fim de sua transmissão
10	<i>TransDATA</i>	<i>WaitACK</i>	Fim da Transmissão do quadro de dados	Aguarda início do recebimento do quadro ACK
11	<i>WaitACK</i>	<i>RecACK</i>	Início do recebimento do quadro ACK	Aguardar término de recebimento
12	<i>RecACK</i>	<i>Idle</i>	Término de recebimento de quadro ACK	-----

Fonte: Elaborado pelo próprio autor.

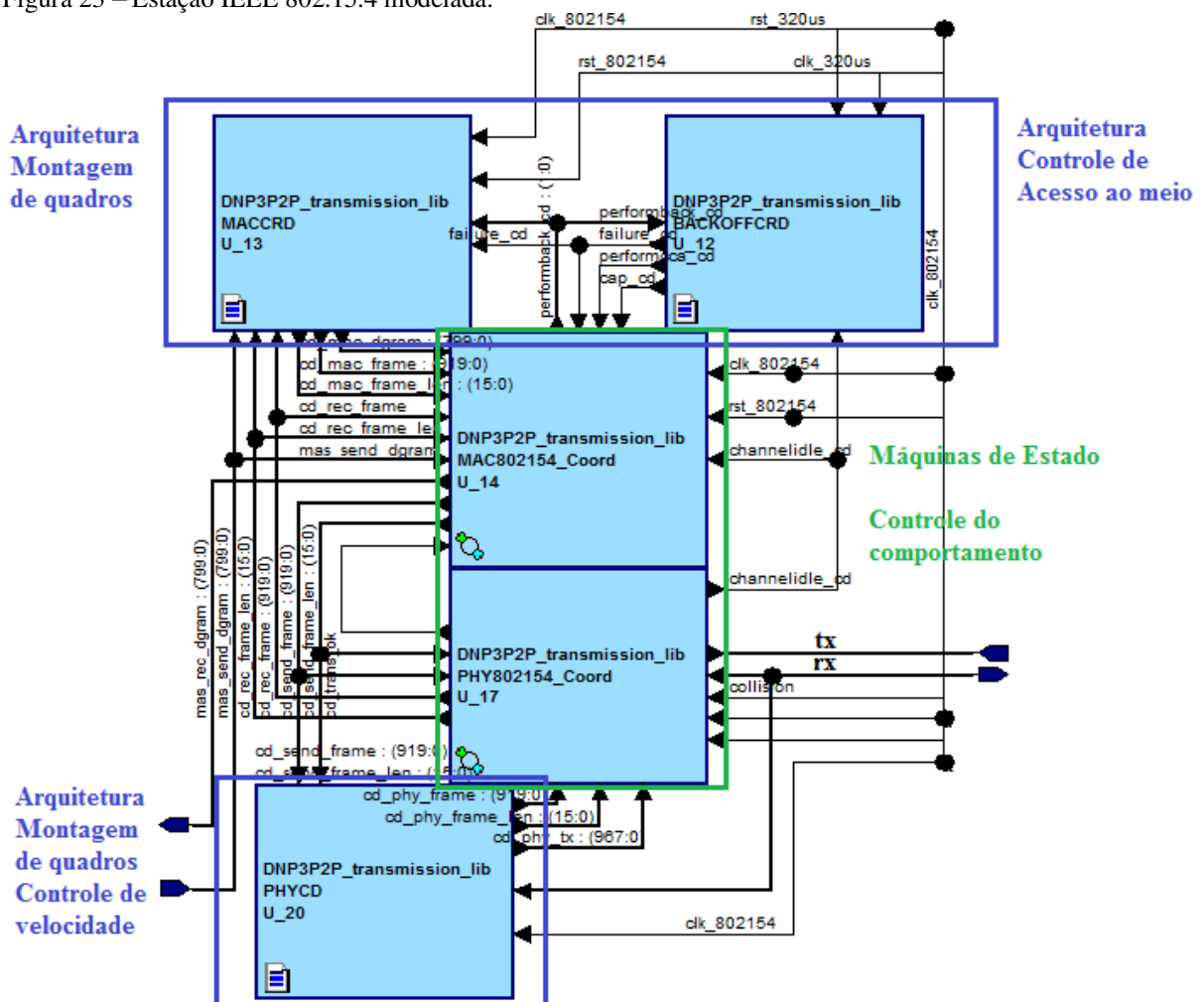
A camada PHY modelada realiza somente a transmissão de acordo com a velocidade do protocolo IEEE 802.15.4 (250 kbps), a montagem de cabeçalho PHY e *preamble*, descrito na Tabela 22, e o auxílio na utilização da função CCA. A modelagem de umas das estações IEEE 802.15.4 é ilustrada na Figura 23.

Tabela 22 – Quadro PHY do protocolo IEEE 802.15.4.

Campo	Tamanho	Descrição
<i>Preamble</i>	4 octetos	Sequência de zeros que antecedem o início do quadro.
<i>SFD</i>	1 octetos	<i>Start Frame Delimiter</i> representa o octeto x“a7” que indica início do quadro.
<i>Lenght</i>	1 octetos	Indica o número de octetos que estão presentes no <i>payload</i> .
<i>Payload</i>	Até 127 octetos	Dados provenientes das camadas superiores.

Fonte: (IEEE, 2011b).

Figura 23 – Estação IEEE 802.15.4 modelada.



Fonte: Elaborado pelo próprio autor.

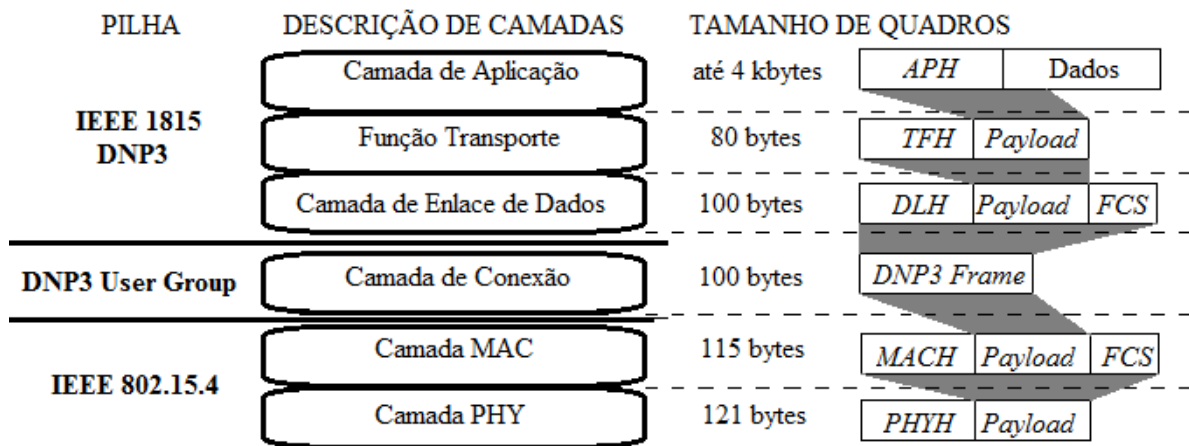
4.4 Integração IEEE 1815/802.15.4

A integração IEEE 1815/802.15.4 procurou manter as características desses protocolos. Entretanto algumas mudanças foram inevitáveis, como:

- Observando a recomendação do DNP3 (IEEE, 2012) e considerando que o protocolo IEEE 802.15.4 já utiliza quadros ACK para confirmação de recebimento, essa característica foi desabilitada no DNP3;
- Para tornar possível a integração dos protocolos, o tamanho máximo do datagrama DNP3 foi reduzido de 292 bytes para 100 bytes, conforme apresentado na Figura 24.

Segundo a recomendação de (IEEE, 2012), a integração deve desabilitar o sistema de confirmação da camada de enlace de dados do DNP3, ou seja, ACK. Essa funcionalidade está então sob responsabilidade do protocolo IEEE 802.15.4.

Figura 24 - Encapsulamento DNP3 / IEEE 802.15.4.



Fonte: Elaboração do próprio autor.

Após descrição da modelagem realizada das integrações IEEE 1815/802.11b e IEEE 1815/802.15.4, o Capítulo 5 efetua a validação das mesmas através de cenários de simulação de medição de latência em relação a disputa de acesso ao meio.

5 SIMULAÇÕES E RESULTADOS

Neste capítulo apresenta-se os cenários de simulação e validação das integrações modeladas no trabalho. A validação ocorreu através de seis cenários de execução do comando API DNP3 *Delay Measure*, utilizando as interfaces abordadas. Os cenários analisam a latência medida pelo comando e validam a utilização das interfaces em relação aos requisitos *Smart Grid* de latência previstos pela literatura, como descrito no Capítulo 3.

Os cenários simulados são:

- Cenário 1 – Medição de Latência através da interface IEEE 802.11b sem utilização do mecanismo *Backoff*. Simbolizando ambiente livre de disputa para acesso;
- Cenário 2 - Medição de Latência através da interface IEEE 802.11b com utilização de 8 *timeslots* de *Backoff*. Simbolizando disputa mediana por acesso;
- Cenário 3 - Medição de Latência através da interface IEEE 802.11b com utilização de 16 *timeslots* de *Backoff*. Simbolizando alta disputa por acesso;
- Cenário 4 – Medição de Latência através da interface IEEE 802.15.4 sem utilização do mecanismo *Backoff*. Simbolizando ambiente livre de disputa para acesso;
- Cenário 5 - Medição de Latência através da interface IEEE 802.15.4 com utilização de 8 *timeslots* de *Backoff*. Simbolizando disputa mediana por acesso;
- Cenário 6 - Medição de Latência através da interface IEEE 802.15.4 com utilização de 16 *timeslots* de *Backoff*. Simbolizando alta disputa por acesso;

5.1 Simulações dos cenários IEEE 1815/802.11b

A integração IEEE 1815/802.11b foi utilizada nas simulações dos Cenários 1, 2 e 3 ilustrados nas Figuras 25, 26 e 27, e de numeração de eventos descritas nas Tabelas 23, 24 e 25, respectivamente.

A fim de permitir maior compreensão das simulações, foram ilustradas nas Figuras apenas transições as mudanças de estado da camada MAC IEEE 802.11b. Entretanto a análise das simulações foi feita em conjunto com a formação dos quadros e eventos descritos nas tabelas de forma resumida às transições apresentadas na seção 4.1

As transições de estado provenientes do protocolo IEEE 1815 DNP3 são semelhantes ao descrito no Capítulo 3.

Figura 25 – Simulação Cenário 1 – IEEE 1815/802.11b.

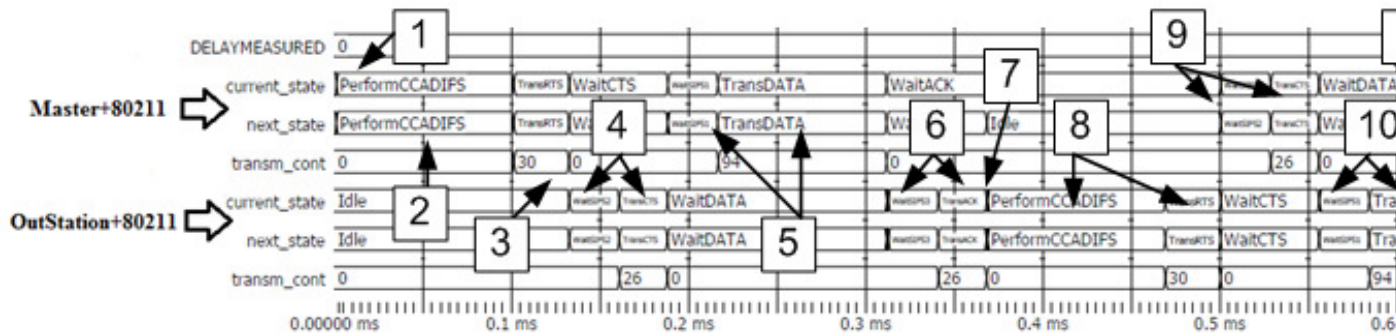


Figura 26 – Simulação Cenário 2 – IEEE 1815/802.11b.

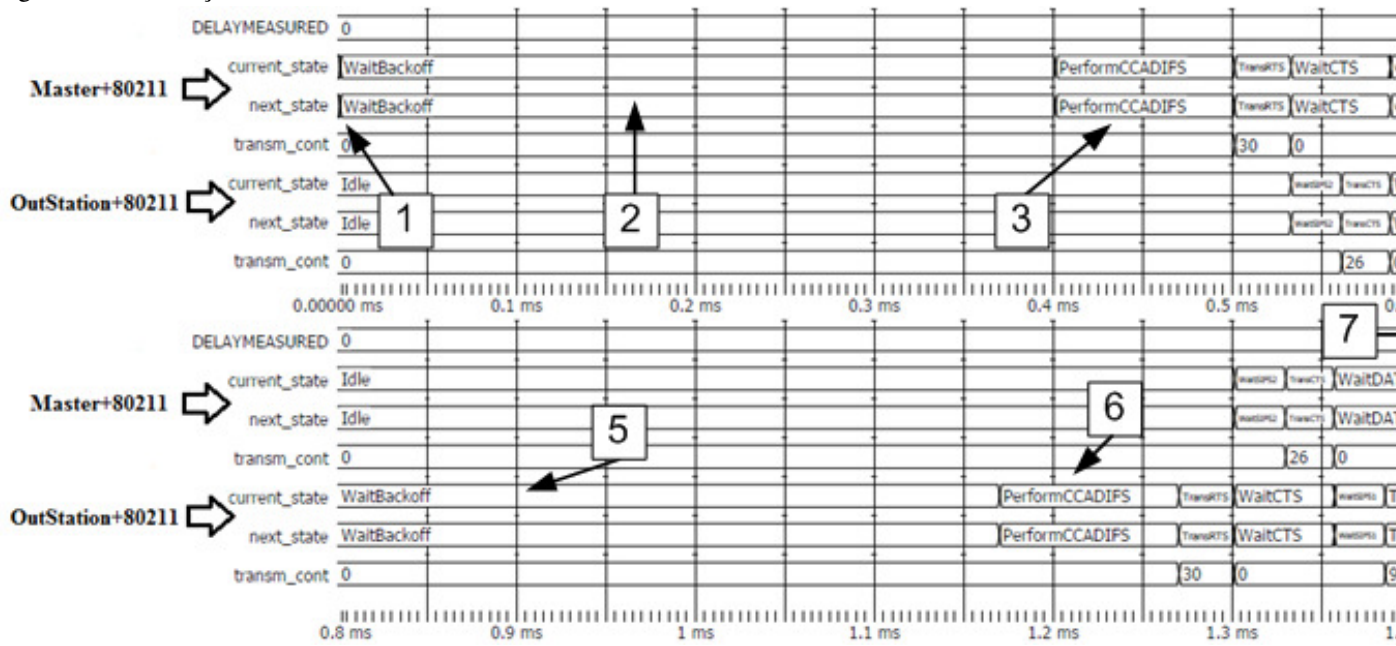


Figura 27– Simulação Cenário 3 – IEEE 1815/802.11b.

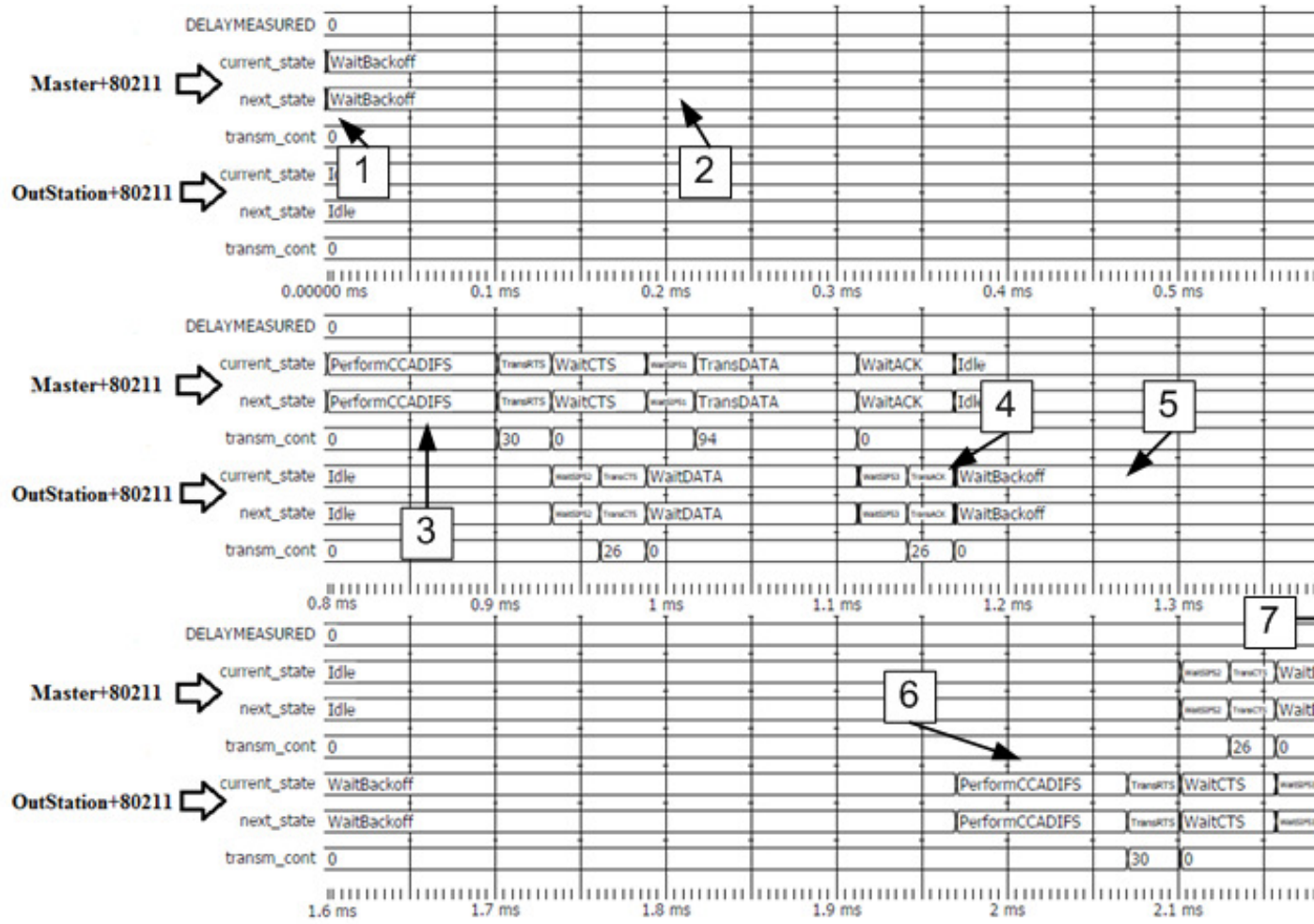


Tabela 23 – Descrição de eventos da simulação do Cenário 1.

Eventos	Transição de Estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>WaitNav</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Delay Measure</i> encapsulado em protocolos LLC, TCP e IPv4. Os dados são de 76 bytes que possuem separação em cabeçalhos IPH (20 bytes)+TCPH (20 bytes) +LLCH (8 bytes) +DNP3 Frame (28 bytes) (Figura 18).
2	<i>WaitNav</i>	<i>PerformCCADIFS</i>	Para efetuar o envio, a estação IEEE 802.11 precisa ativar a função CCADIFS para verificar a assinatura do canal.
3	<i>PerformCCADIFS</i>	<i>TransRTS</i>	Uma vez não detectada atividade no canal a estação IEEE 802.11 vinculada ao <i>Master</i> pode efetuar a transmissão do quadro RTS (44 bytes).
4	<i>Idle</i>	<i>WaitSIFS2</i>	A estação 802.11 vinculada ao <i>Outstation</i> recebe um quadro RTS endereçado ao <i>Master</i> e iniciado então um período de espera SIFS e posterior envio de quadro CTS (44 bytes).
	<i>WaitSIFS2</i>	<i>TransCTS</i>	
5	<i>WaitCTS</i>	<i>WaitSIFS1</i>	A estação 802.11 vinculada ao <i>Master</i> recebe o quadro CTS. É iniciado então um período de espera SIFS e posterior envio de quadro de dados (136 bytes).
	<i>WaitSIFS1</i>	<i>TransDATA</i>	
6	<i>WaitData</i>	<i>WaitSIFS3</i>	A estação 802.11 vinculada ao <i>Outstation</i> recebe um quadro de dados e responde com um quadro <i>payload</i> para as camadas superiores. É iniciado então um período de espera SIFS e posterior envio de quadro ACK (38 bytes).
	<i>WaitSIFS3</i>	<i>TransACK</i>	
7	<i>Idle</i>	<i>WaitNAV</i>	Dispositivo <i>Outstation</i> envia o comando DNP3 API <i>Response</i> encapsulado em protocolos LLC, TCP e IPv4. O quadro possui o mesmo tamanho do descrito no evento 1.
8	<i>WaitNav</i>	<i>PerformCCADIFS</i>	Para efetuar o envio, a estação IEEE 802.11 precisa ativar a função CCADIFS para verificar a assinatura do canal.
	<i>PerformCCADIFS</i>	<i>TransRTS</i>	Uma vez não detectada atividade no canal a estação IEEE 802.11 vinculada ao <i>Outstation</i> pode efetuar a transmissão do quadro RTS (44 bytes).
9	<i>Idle</i>	<i>WaitSIFS2</i>	A estação 802.11 vinculada ao <i>Master</i> recebe um quadro RTS endereçado ao <i>Outstation</i> e iniciado então um período de espera SIFS e posterior envio de quadro CTS.
	<i>WaitSIFS2</i>	<i>TransCTS</i>	
10	<i>WaitCTS</i>	<i>WaitSIFS1</i>	A estação 802.11 vinculada ao <i>Outstation</i> recebe o quadro CTS. É iniciado então um período de espera SIFS e posterior envio de quadro de dados.
	<i>WaitSIFS1</i>	<i>TransDATA</i>	
11	-----	-----	A latência medida pelo <i>Master</i> indica ser menor que 1 ms, quando de fato é maior conforme simulação realizada.
12	<i>WaitData</i>	<i>WaitSIFS3</i>	A estação 802.11 vinculada ao <i>Master</i> recebe um quadro de dados e responde com um quadro <i>payload</i> para as camadas superiores. É iniciado então um período de espera SIFS e posterior envio de quadro ACK (38 bytes).
	<i>WaitSIFS3</i>	<i>TransACK</i>	
13	-----	-----	Inicia-se o envio do comando de Confirmação API do <i>Master</i> . As etapas de comunicação são semelhantes aos eventos 1 ao 6 e se finalizam em 1,5 ms.

Tabela 24 – Descrição de eventos da simulação do Cenário 2.

Eventos	Transição de Estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>WaitNAV</i>	Evento semelhante ao descrito no evento 1 do Cenário 1.
2	<i>WaitNAV</i>	<i>PerformCCADIFS</i>	Ao ativar a função CCA, a estação IEEE 802.11 vinculada ao <i>Master</i> detecta tempo de espera <i>Backoff</i> (8 <i>timeslots</i> ou 400 <i>us</i>) para a próxima tentativa de a
	<i>PerformCCADIFS</i>	<i>WaitBackoff</i>	
3	<i>WaitBackoff</i>	<i>PerformCCADIFS</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.11 vinculada ao quadro RTS. E assim a transmissão ocorre de forma semelhante aos eventos 3
	<i>PerformCCADIFS</i>	<i>TransRTS</i>	
4	<i>Idle</i>	<i>WaitNAV</i>	Evento semelhante ao descrito no evento 7 do Cenário 1.
5	<i>WaitNAV</i>	<i>PerformCCADIFS</i>	Ao ativar a função CCA, a estação IEEE 802.11 vinculada ao <i>Outstation</i> detecta tempo de espera <i>Backoff</i> (8 <i>timeslots</i> ou 400 <i>us</i>) para a próxima tentativa de a
	<i>PerformCCADIFS</i>	<i>WaitBackoff</i>	
6	<i>WaitBackoff</i>	<i>PerformCCADIFS</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.11 vinculada ao quadro RTS. E assim a transmissão ocorre de forma semelhante aos eventos
	<i>PerformCCADIFS</i>	<i>TransRTS</i>	
7	-----	-----	A latência medida pelo <i>Master</i> indica ser menor que 1 <i>ms</i> , quando de fato é 0.
8	-----	-----	Inicia-se o envio do comando de Confirmação API do <i>Master</i> . As etapas eventos 1 ao 6 do Cenário 1 e se finalizam em 2 <i>ms</i> .

Tabela 25 – Descrição de eventos da simulação do Cenário 3.

Eventos	Transição de Estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>WaitNAV</i>	Evento semelhante ao descrito no evento 1 do Cenário 1.
2	<i>WaitNAV</i>	<i>PerformCCADIFS</i>	Ao ativar a função CCA, a estação IEEE 802.11 vinculada ao <i>Master</i> detecta tempo de espera <i>Backoff</i> (16 <i>timeslots</i> ou 800 <i>us</i>) para a próxima tentativa de
	<i>PerformCCADIFS</i>	<i>WaitBackoff</i>	
3	<i>WaitBackoff</i>	<i>PerformCCADIFS</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.11 vinculada ao quadro RTS. E assim a transmissão ocorre de forma semelhante aos eventos 3
	<i>PerformCCADIFS</i>	<i>TransRTS</i>	
4	<i>Idle</i>	<i>WaitNAV</i>	Evento semelhante ao descrito no evento 7 do Cenário 1.
5	<i>WaitNAV</i>	<i>PerformCCADIFS</i>	Ao ativar a função CCA, a estação IEEE 802.11 vinculada ao <i>Outstation</i> detecta tempo de espera <i>Backoff</i> (16 <i>timeslots</i> ou 800 <i>us</i>) para a próxima tentativa de
	<i>PerformCCADIFS</i>	<i>WaitBackoff</i>	
6	<i>WaitBackoff</i>	<i>PerformCCADIFS</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.11 vinculada ao quadro RTS. E assim a transmissão ocorre de forma semelhante aos eventos
	<i>PerformCCADIFS</i>	<i>TransRTS</i>	
7	-----	-----	A latência medida pelo <i>Master</i> foi de 1 <i>ms</i> , quando de fato é 1,2 <i>ms</i> conforme
8	-----	-----	Inicia-se o envio do comando de Confirmação API do <i>Master</i> . As etapas eventos 1 ao 6 do Cenário 1 e se finalizam em 3 <i>ms</i> .

5.2 Simulações dos cenários IEEE 1815/802.15.4

A integração IEEE 1815/802.15.4 foi utilizada nas simulações dos cenários 4, 5 e 6 ilustrados nas Figuras 28, 29 e 30, e de numeração de eventos descritas nas Tabelas 26, 27 e 28, respectivamente.

A fim de permitir maior compreensão das simulações, foram ilustradas nas Figuras apenas transições as mudanças de estado da camada MAC IEEE 802.15.4. Entretanto a análise das simulações foi feita em conjunto com a formação dos quadros e eventos descritos nas tabelas de forma resumida às transições apresentadas no seção 4.3.

As transições de estado provenientes do protocolo IEEE 1815 DNP3 são semelhantes ao descrito no Capítulo 3.

Figura 28 – Simulação Cenário 4 – IEEE 1815/802.15.4.

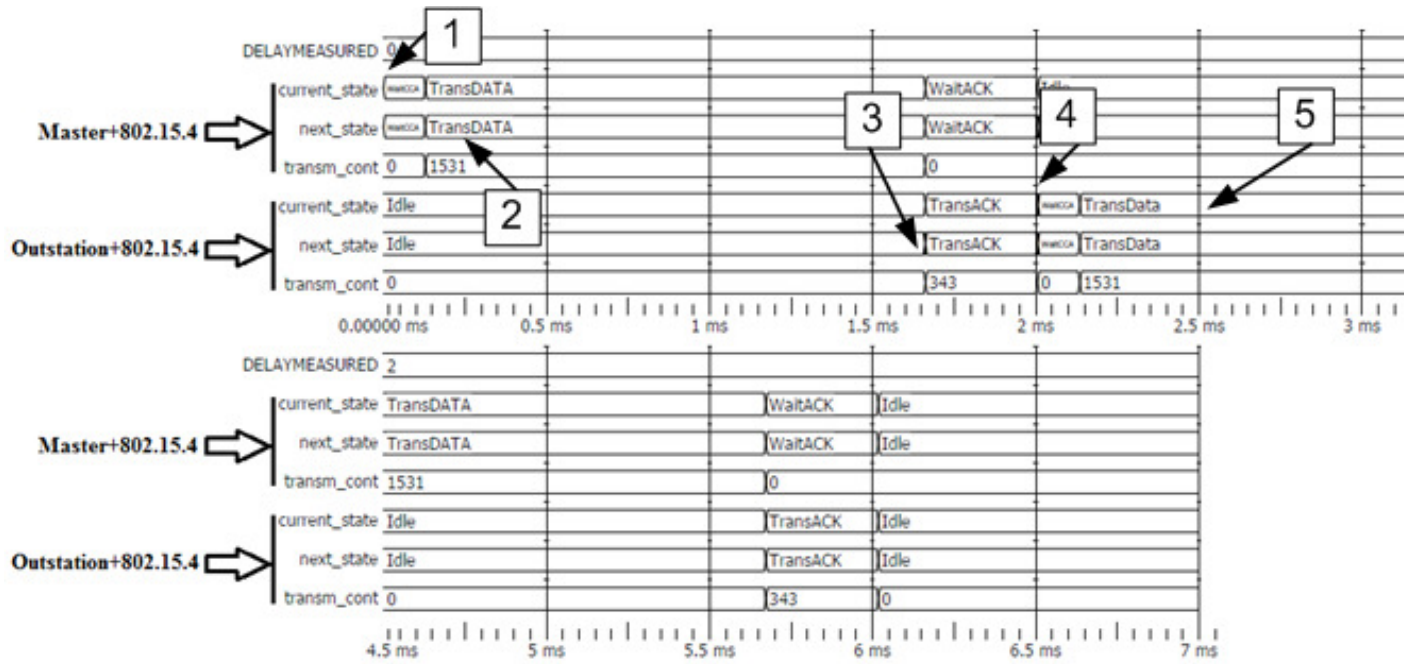


Figura 29 – Simulação Cenário 5 – IEEE 1815/802.15.4.

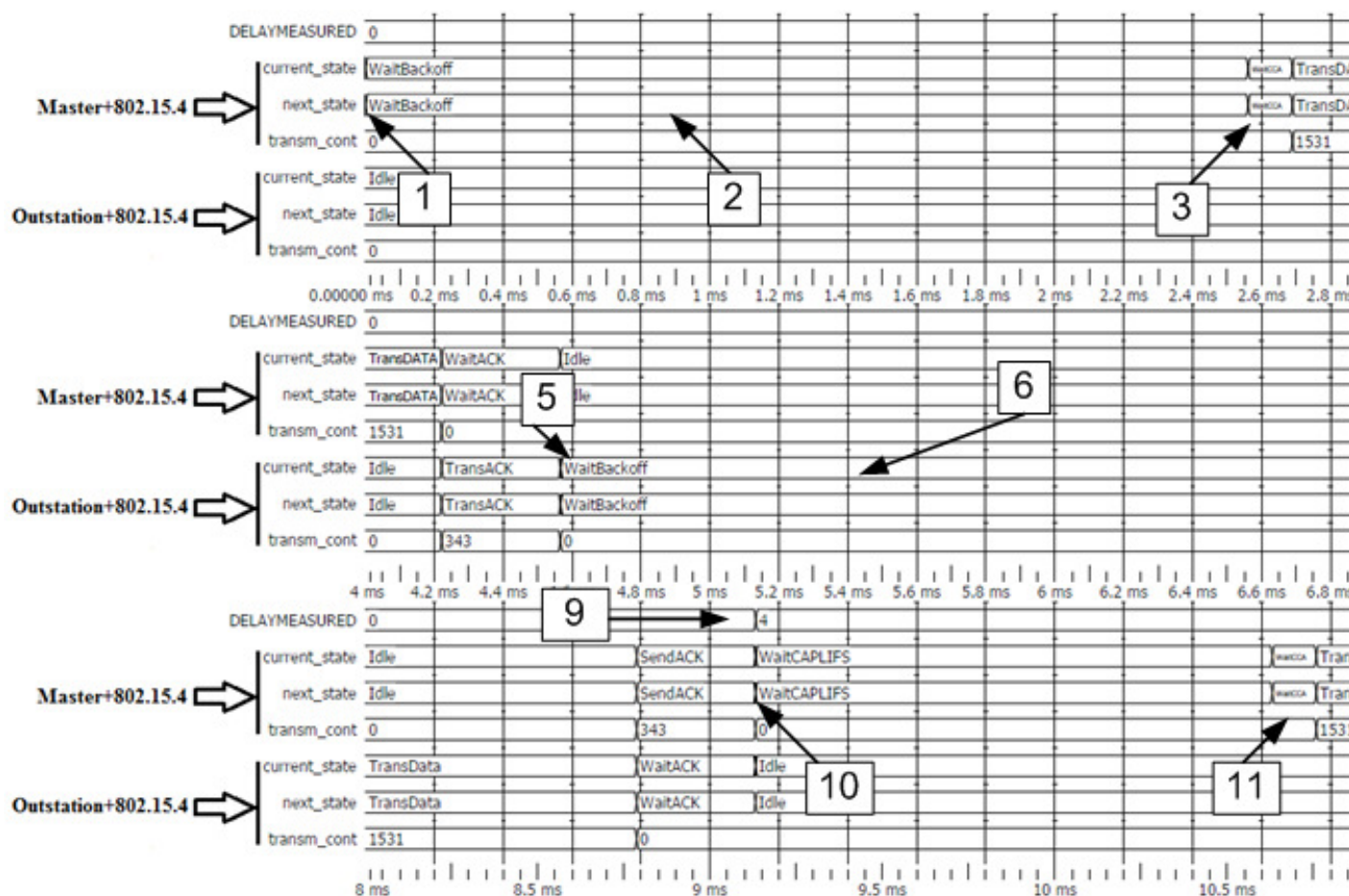


Figura 30 – Simulação Cenário 6 – IEEE 1815/802.15.4.

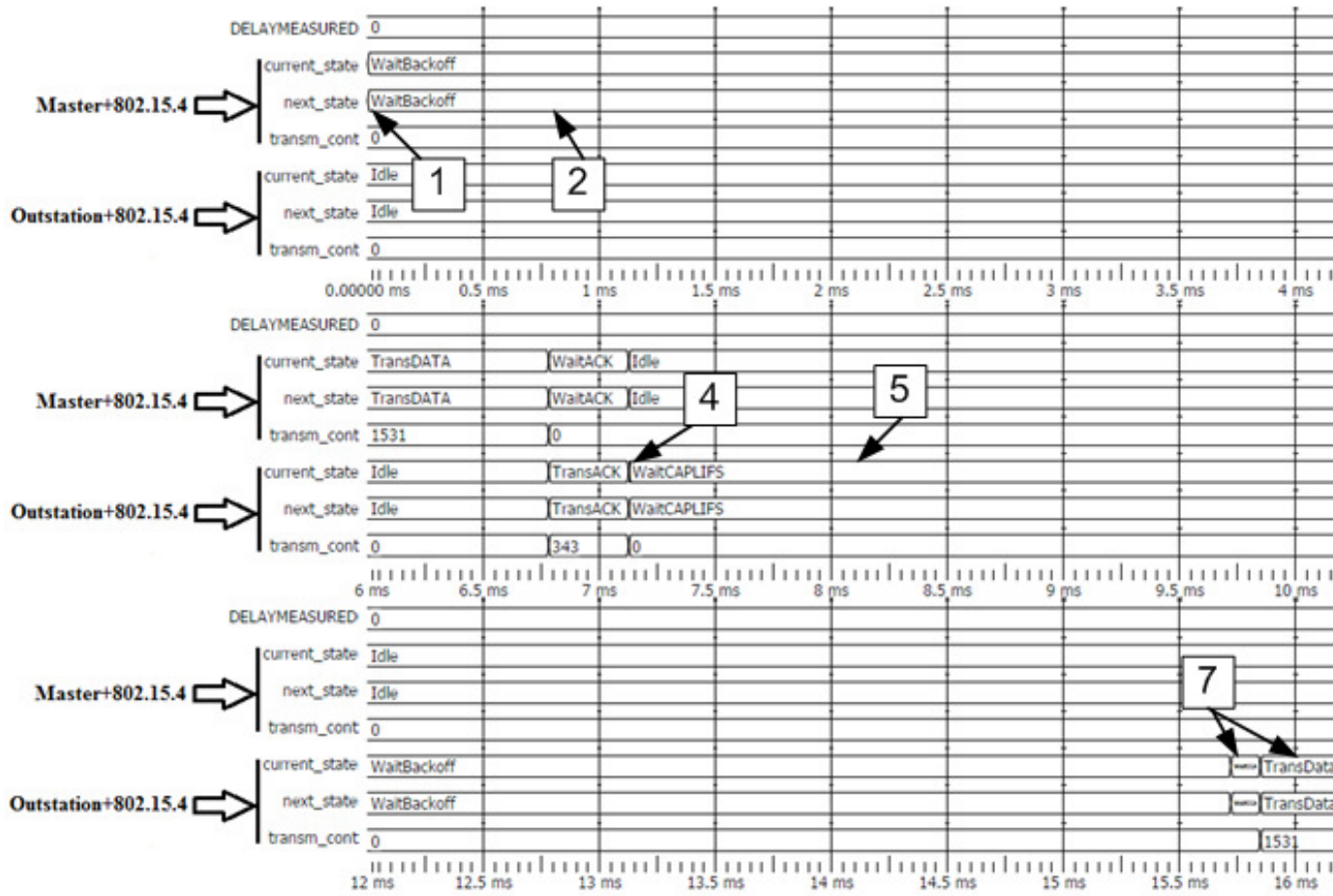


Tabela 26 – Descrição de eventos da simulação do Cenário 4.

Eventos	Transição de Estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>WaitCCA</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Delay Measure</i> . Os dados são de 28 bytes. Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Master</i> precisa ativar a função CCA para verificar a assinatura do canal.
2	<i>WaitCCA</i>	<i>TransDATA</i>	Uma vez não detectada atividade no canal a estação IEEE 802.15.4 vinculada ao <i>Master</i> pode efetuar a transmissão do quadro de dados (39 bytes).
3	<i>Idle</i>	<i>RecDATA</i>	A estação IEEE 802.15.4 vinculada ao <i>Outstation</i> recebe o quadro de dados e repassa o <i>payload</i> para as camadas superiores e envia o ACK (5 bytes).
	<i>RecDATA</i>	<i>TransACK</i>	
	<i>TransACK</i>	<i>Idle</i>	
4	<i>Idle</i>	<i>WaitCCA</i>	Dispositivo <i>Outstation</i> envia o comando DNP3 API <i>Solicited Response</i> de 28 bytes. Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> precisa ativar a função CCA para verificar a assinatura do canal.
5	<i>WaitCCA</i>	<i>TransDATA</i>	Uma vez não detectada atividade no canal a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> pode efetuar a transmissão do quadro de dados (39 bytes).
6	<i>Idle</i>	<i>RecDATA</i>	A estação IEEE 802.15.4 vinculada ao <i>Master</i> recebe o quadro de dados e repassa o <i>payload</i> para as camadas superiores e envia o ACK.
	<i>RecDATA</i>	<i>TransACK</i>	
	<i>TransACK</i>	<i>Idle</i>	
7	-----	-----	A latência medida pelo <i>Master</i> foi de 2 ms.
8	<i>Idle</i>	<i>WaitCCA</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Confirm</i> . Os dados são de 28 bytes. Inicia-se assim um processo semelhante aos eventos 1, 2 e 3. O envio de mensagens se finaliza em 6 ms.

Tabela 27 – Descrição de eventos da simulação do Cenário 5.

Eventos	Transição de Estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>WaitCCA</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Delay Measure</i> . Os envio, a estação IEEE 802.15.4 vinculada ao <i>Master</i> precisa ativar a função canal.
2	<i>WaitCCA</i>	<i>WaitBackoff</i>	Ao ativar a função CCA, a estação IEEE 802.15.4 vinculada ao <i>Master</i> segue assim o tempo de espera <i>Backoff</i> (8 <i>timeslots</i> ou 2560 <i>us</i>) para a próxima transmissão.
3	<i>WaitBackoff</i>	<i>WaitCCA</i>	Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Master</i> precisa verificar a assinatura do canal.
4	<i>WaitCCA</i>	<i>TransDATA</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.15.4 vinculada ao <i>Master</i> segue um processo semelhante aos eventos 2 e 3 do Cenário 4.
5	<i>Idle</i>	<i>WaitCCA</i>	Dispositivo <i>Outstation</i> envia o comando DNP3 API <i>Solicited Response</i> . Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> precisa verificar a assinatura do canal.
6	<i>WaitCCA</i>	<i>WaitBackoff</i>	Ao ativar a função CCA, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> segue assim o tempo de espera <i>Backoff</i> (8 <i>timeslots</i> ou 2560 <i>us</i>) para a próxima transmissão.
7	<i>WaitBackoff</i>	<i>WaitCCA</i>	Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> precisa verificar a assinatura do canal.
8	<i>WaitCCA</i>	<i>TransDATA</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> segue um processo semelhante aos eventos 5 e 6 do Cenário 4.
9	-----	-----	A latência medida pelo <i>Master</i> foi de 4 <i>ms</i> .
10	<i>Idle</i>	<i>WaitCAPLIFS</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Confirm</i> . A estação <i>Outstation</i> inicia um período ativo de disputa (CAP).
11	<i>WaitCAPLIFS</i>	<i>WaitCCA</i>	Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Master</i> precisa verificar a assinatura do canal. Inicia-se assim um processo semelhante aos eventos 2 e 3 do Cenário 4. O processo se finaliza em 13 <i>ms</i> .

Tabela 28 – Descrição de eventos da simulação do Cenário 6.

Eventos	Transição de Estado		Descrição
	Atual	Próximo	
1	<i>Idle</i>	<i>WaitCCA</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Delay Measure</i> . Os envio, a estação IEEE 802.15.4 vinculada ao <i>Master</i> precisa ativar a função CCA para verificar a assinatura do canal.
2	<i>WaitCCA</i>	<i>WaitBackoff</i>	Ao ativar a função CCA, a estação IEEE 802.15.4 vinculada ao <i>Master</i> aguarda assim o tempo de espera <i>Backoff</i> (16 <i>timeslots</i> ou 5120 <i>us</i>) para a próxima tentativa de envio.
3	<i>WaitBackoff</i>	<i>WaitCCA</i>	Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Master</i> precisa verificar a assinatura do canal. Inicia-se um processo semelhante aos eventos 1, 2 e 3 do Cenário 4.
4	<i>Idle</i>	<i>WaitCAPLIFS</i>	Dispositivo <i>Outstation</i> envia o comando DNP3 API <i>Solicited Response</i> . A estação IEEE 802.15.4 espera o reinício do período ativo de disputa (CAP).
5	<i>WaitCAPLIFS</i>	<i>WaitCCA</i>	Para efetuar o envio, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> precisa verificar a assinatura do canal.
6	<i>WaitCCA</i>	<i>WaitBackoff</i>	Ao ativar a função CCA, a estação IEEE 802.15.4 vinculada ao <i>Outstation</i> aguarda assim o tempo de espera <i>Backoff</i> (16 <i>timeslots</i> ou 5120 <i>us</i>) para a próxima tentativa de envio.
7	<i>WaitBackoff</i>	<i>WaitCCA</i>	Uma vez não detectada atividade no canal, a estação IEEE 802.15.4 inicia um processo semelhante aos eventos 5 e 6 do Cenário 4.
	<i>WaitCCA</i>	<i>TransDATA</i>	
8	-----	-----	A latência medida pelo <i>Master</i> foi de 7 <i>ms</i> .
9	<i>Idle</i>	<i>WaitCAPLIFS</i>	Dispositivo <i>Master</i> envia o comando DNP3 API <i>Confirm</i> . A estação IEEE 802.15.4 aguarda o período ativo de disputa (CAP). Para efetuar o envio, a estação IEEE 802.15.4 precisa ativar a função CCA para verificar a assinatura do canal. Inicia-se assim o processo semelhante aos eventos 1, 2 e 3 do Cenário 4. A troca de mensagens se finaliza em 20 <i>ms</i> .
	<i>WaitCAPLIFS</i>	<i>WaitCCA</i>	

5.3 Análise de Resultados

A Tabela 29 apresenta os resultados obtidos nos seis cenários e os classificam de acordo com os requisitos de latência, apresentados na Tabela 1. O período de *clock* das entidades modeladas, por terem o valor de 100 *ns*, não apresentam grande influência entre a latência de código sobre a latência de transmissão. Os resultados são analisados sobre a latência medida, entre a tentativa de transmissão e a recepção dos dados, e o final de cenário, simbolizando o fim de todas as trocas de mensagens entre os dispositivos DNP3 cujo comunicação é descrita no Capítulo 3.

Tabela 29 - Resultados Obtidos nas simulações e comparação com requisitos de latência.

	Atividade de canal ¹	Período <i>Backoff</i> (time slots) ¹	Latência Medida (ms) ¹	Final do Cenário (ms) ¹	Classe Nível ²	Aplicações SG ³
Cenário 1	Ocioso	----	0,4	1,5	1 (<3 ms)	Teleproteção
Cenário 2	Ocupado	8	0,8	2	1 (<3 ms)	Teleproteção
Cenário 3	Ocupado	16	1,2	3	1 (<16 ms)	Teleproteção
Cenário 4	Ocioso	----	2	6	1 (<16 ms)	Teleproteção
Cenário 5	Ocupado	8	4	13	1 (<16 ms)	Teleproteção
Cenário 6	Ocupado	16	7	20	2 (<160 ms)	Dados SCADA, Comunicação WAN, WACS, PMU

Fonte: ¹Dados da pesquisa do autor; ²Adaptado de (IEEE 2030, 2011); ³Adaptado de (LU et al, 2013; SADEGHI et al, 2012)

Os cenários 1, 2 e 3 simularam medições de atraso, na comunicação P2P DNP3 encapsulado no protocolo IEEE 802.11b, com valores menores que 3 *ms*. Ao analisar a Tabela 29, verifica-se que a abordagem, de utilização da interface IEEE 802.11b efetuando a comunicação entre dois Smart Meters DNP3, é promissora. Segundo a tabela de requisitos, esta interface de comunicação poderia ser utilizada até mesmo na teleproteção de subestações.

A latência encontrada neste primeiro enlace intra-subestação, será potencialmente aumentada na comunicação WAN entre a própria subestação e a Central de Controle. Por isso, a latência de cada enlace do fluxo de dados deve ser a menor possível para que atenda os requisitos previstos para menor caso de 16 *ms*.

Por esse motivo, a recomendação de utilização em teleproteção de subestações através da interface IEEE 802.15.4 não pode ser feita. Afinal em cenários de média ou alta disputa de acesso ao meio, ou seja, Cenários 5 e 6, a latência encontra-se muito elevada em relação a outra interface utilizada, inviabilizando a transmissão de dados de natureza digital e crítica,

conforme análise realizada no Capítulo 3 seção 1. Além disso, mesmo se aplicada na transmissão de dados analógicos, que não exigem baixa latência, a interface possui baixa capacidade de vazão de dados, o que pode criar dificuldades e atrasos de transmissão.

Apesar disso, pode-se observar que para transmitir uma mesma quantidade de informação DNP3, o protocolo IEEE 802.15.4 tem menor complexidade de formação de quadros. Ou seja, enquanto que para a integração IEEE 1815/802.11b efetuar a transmissão de 28 bytes DNP3 é gerado um quadro com 136 bytes, a integração IEEE 1815/802.15.4 produz o quadro de mesmo conteúdo com 39 bytes. Através da baixa complexidade de transmissão de dados, aliada ao baixo consumo de energia, inerente à natureza do protocolo (SANTOS, 2003), pode-se sugerir integração terá sua utilização em aplicações SG que demandem alguma economia de energia, como no monitoramento de última milha ou enlaces de *Backbone* de menores distâncias (Leitura AMI, HAN, BAN e NAN).

As simulações realizadas, não consideram possíveis erros de transmissão, distância entre nós, entre outras características naturais ao ambiente real de aplicação. Para que sejam aplicadas, as integrações modeladas no trabalho necessitam ainda passar por testes que avaliem sua robustez em relação às interferências eletromagnéticas do ambiente, confiabilidade, disponibilidade, escalabilidade e segurança. Além disso, simulações em arquiteturas ponto-multiponto devem trazer maior representatividade de resultados.

Contudo, a modelagem realizada contribui como projeto inicial para implementações futuras, tanto para que contemple cenários de simulação escaláveis, quanto para possibilitar a síntese de dispositivo FPGA.

6 CONCLUSÕES

A modelagem VHDL das integrações IEEE 1815/ 802.11b e IEEE 1815/ 802.15.4 realizadas neste trabalho se mostraram efetivas em descrever os protocolos de comunicação. A abordagem atual do código VHDL dividiu as funções das camadas em duas entidades. Uma delas efetuou apenas a montagem dos blocos de bits. A outra, modelada em máquina de estado, representa o comportamento do protocolo no envio/ recepção de dados. Essa divisão de tarefas da camada diminui o tempo de resposta de código.

Para futura síntese de dispositivo FPGA, melhorias serão realizadas para que o *hardware* seja capaz de suportar topologias ponto-multiponto. Espera-se assim que seu tempo de resposta seja ainda menor e além de possibilitar que um *hardware* sintetizado seja utilizados em ambientes reais.

Apesar dos cenários terem utilizado o comando DNP3 *Delay Measure* na comunicação, a modelagem atual já contempla a utilização de outros comandos, cuja representação da comunicação realizada é semelhante à apresentada neste trabalho.

Através das simulações e medições de latência dos Cenários 1, 2 e 3, observa-se que a integração IEEE 1815/ 802.11b é promissora e está dentro dos requisitos de latência descritos na literatura. Entretanto, para utilização em ambientes críticos como subestações, testes de segurança, confiabilidade e escalabilidade devem ser feitos.

Os Cenários 4, 5 e 6 demonstram que, apesar de atender os requisitos de teleproteção de subestações, a integração IEEE 1815/ 802.15.4 possui latência muito elevada quando exposta à um ambiente de média e alta disputa de acesso ao meio. Entretanto, devido à baixa complexidade para transmitir informações, essa abordagem pode ser utilizada em aplicações SG que demanda maior economia de energia para o monitoramento e transmissão de dados.

Os trabalhos futuros estarão voltados em adaptar a modelagem para cenários escaláveis em número de medidores e flexíveis nos tipos de interfaces de comunicação que podem ser utilizados. O objetivo é representar com maior fidelidade o comportamento e desempenho do fluxo de dados *Smart Grid*.

REFERÊNCIAS

- AAD, I; CASTELLUCCIA, C. Differentiation mechanisms for IEEE 802.11. **IEEE INFOCOM**, Anchorage, v. 1, n. 1, p. 209 - 218, 2001.
- ANTUNES, R. **Instalação de uma rede mesh metropolitana utilizando o padrão IEEE 802.11a e implementação do serviço VOIP (WMAN-VOIP)**. 2012. 95 f. Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia, Universidade Estadual Paulista- UNESP, Ilha Solteira, 2012.
- ARAÚJO, N. V. de S. **Kappa-PSO-ARTMAP fuzzy: Uma metodologia para detecção de intrusos baseado em seleção de atributos e otimização de parâmetros numa rede neural artmap fuzzy**. 2013. 112 f. Tese (Doutorado em Engenharia Elétrica)- Faculdade de Engenharia, Universidade Estadual Paulista- UNESP, Ilha Solteira, 2013. p. 112.
- ARAVINTHAN, V.; KARIMI, B.; NAMBOODIRI, V.; JEWELL, W. Wireless communication for smart grid applications at distribution level: feasibility and requirements. **Power and Energy Society General Meeting**, San Diego, v. 1, n. 1, p. 24-29, 2011.
- BHAVIKATTI, A. M.; HAMEED, M.; DESHPANDE, D.; DESHPANDE, L. M. FPGA implementation of MAC header block of transmitter for Wi-fi. **International Journal of Engineering Science and Technology- IJEST**, Chennai, v. 16, n. 2, p. 3327-3335, 2011.
- BIANCHI, G. Performance analysis of the IEEE 802.11 distributed coordination function. **IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS**, New York, v. 18, n. 3, p. 535-547, 2000.
- BRITO, M. A. G. de, **Pré-regulador Retificador Boost com Controle Digital por Valores Médios, para Sistema de Iluminação Fluorescente Multi-lâmpadas, utilizando Dispositivo FPGA e VHDL**. 2008. 216 f. Dissertação (Mestrado)- Faculdade de Engenharia, Universidade estadual Paulista- UNESP, Ilha Solteira, 2008.
- BURATTI, C.; VERDONNE, R. Performance analysis of IEEE 802.15.4 non beacon-enabled mode. **IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY**, New York, v. 58, n. 7, p. 3480-3494, 2009.
- COX, R. J.; BRAGER R. F.; COFFEL, J. A.; BRANDT, N. A. **Reconfigurable Load-Control Receiver - Specific application, Apparatus or Process Electrical Power Generation Or Distribution System Power Allocation Management**. U.S Patent EP20120029717, 02 Fe. 2012, Disponível em: <<http://www.faqs.org/patents/app/20120029717>>, Acesso: 17 dez. 2013.
- FADLULLAH, Z.M.; FOUDA, M.M.; KATO, N.; TAKEUCHI, A.; IWASAKI, N.; NOZAKI, Y. Toward intelligent machine-to-machine communications in smart grid. **Communications Magazine, IEEE**, Piscataway, v. 49, n. 4, p. 60-65, April 2011.
- FANG, X.; MISRA, S.; ZUE, G.; YANG, D. Smart grid. The New and Improved Power Grid: A Survey. **Communications Surveys & Tutorials, IEEE**, New York, v. 14, n. 4, p. 944-980, 2012.

GONÇALVES, F. A. S. **Pré-regulador retificador entrelaçado (interleaved) ZCS-FM boost, com controle digital através de dispositivo FPGA e linguagem VHDL**. 2005. 277 f. Tese (Doutorado em Engenharia Elétrica)- Faculdade de Engenharia, Universidade Estadual Paulista- UNESP, Ilha Solteira, 2005.

HASSAN, R.; RADMAN, G. Survey on Smart Grid. **Proceedings of the IEEE Southeast con**, New York, v. 1, n. 1, p. 210-213, March, 2010.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 802.11 TM - 2007**: wireless LAN medium access control- MAC and physical layer- PHY specifications. New York: IEEE, 2007. 1233 p.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 802.15.4 TM - 2011**: low-rate wireless personal area networks- LR-WPANs. New York: IEEE, 2011b. 334 p.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 1076 TM - 2002**: VHDL language reference manual. New York: IEEE, 2002. 309 p.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 1815 TM - 2012**: exchanging information between networks implementing IEC 61850 and IEEE Std 1815 (Distributed Network Protocol - DNP3). New York: IEEE, 2012. 283 p.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 2030 TM - 2011**: Guide for smart grid interoperability of energy technology and information technology operation with the electric power system (EPS), end-use applications, and loads. New York: IEEE, 2011a. 126 p.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 2030 TM - 2011**: logical link control. New York: IEEE, 1984. 116 p.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS – IEEE. **IEEE Std. 802.3 TM - 2003**: carrier sense multiple access with collision detection- CSMA/CD access method and physical layer specifications. New York: IEEE, 2003. 504 p.

JING, H.; AIDA, H. An analytical approach to optimization of throughput for IEEE 802.15.4 slotted CSMA/CA networks. In: **CONSUMER COMMUNICATIONS AND NETWORKING CONFERENCE - SMART SPACES AND PERSONAL AREA NETWORKS**, 8., 2011, New York. Annual... New York: IEEE, 2011. p. 1021-1025.

KALKI TECH. **Intelligent system stack for smart grid**. Bangalore: Kalki tech, 2013. Disponível em: <<http://www.kalkitech.com/solutions/smart-grid>>. Acesso: 17 dez. 2013.

KOHVAKKA, M.; KUORILEHTO, M.; HANNIKAINEN, M.; HAMALAINEN, T. D. Performance analysis of IEEE 802.15.4 and ZigBee for large-scale wireless sensor network

applications. In: INTERNATIONAL WORKSHOP ON PERFORMANCE EVALUATION OF WIRELESS AD HOC, SENSOR AND UBIQUITOUS NETWORKS, 3., 2006, Malaga. Proceedings... New York: ACM, 2006. p. 48-57.

LU, X.; WANG, W.; MA, J. An empirical study of communication infrastructures towards the smart grid: design, implementation, and evaluation. **IEEE TRANSACTIONS ON SMART GRID**, New York, v. 4, n. 1, p. 170-183, 2013.

MENTOR GRAPHICS. **HDL designer 2012**. Wilsonville: Mentor Graphics. 2012. Disponível em: <<http://www.mentor.com>>. Acesso em: 9 nov. 2012.

MODELSIM PE student edition: HDL simulation. Wilsonville: Mentor Graphics. 2010. Disponível em: <<http://model.com>>. Acesso em: 24 out. 2012.

MURTHY, C. S. R.; MANOJ, B. S. **Ad hoc wireless networks**: architectures and protocols. Old Tappan: Prentice Hall, 2004. p. 857.

NEICHIN, G.; CHENG D. **Smart grid vendor ecosystem**: report on the companies and market dynamic shaping the current U.S. smart grid landscap. [S.l.: s.n.], 2010. Available at: <<http://energy.gov/sites/prod/files/edg/news/documents/Smart-Grid-Vendor.pdf>>. Acesso: 17 dez. 2013.

OFFICE OF ELECTRICITY DELIVERY AND ENERGY RELIABILITY. **Smart grid**. [S.l.: s.n.], 2010. Disponível em: <<http://www.oe.energy.gov/smartgrid.htm>>. Acesso: 17 dez. 2013.

PADILLA, E.; CARDENAS, A.; AGBOSSOU, K., Hardware Implementation of ethernet based DNP3 data link. In: IEEE ELECTRICAL POWER AND ENERGY CONFERENCE, 1., 2012, London. **Conference ...** London: IEEE, 2012. p. 8-13.

PANG, W. L.; CHEW, K. W.; CHOONG, F.; TEOH, E. S. VHDL modeling of the CSMA/CA. **International Journal of Communications**, Nova Delhi, v. 1, n. 2, p. 81-87, 2007.

PARIKH, P. P.; KANABAR, M. G.; SIDHU, T. S. Opportunities and challenges of wireless communication technologies for smart grid applications. **Power and Energy Society General Meeting**, Minneapolis, v. 1, n. 1, p. 1-7, 25-29, July 2010.

PERRY, D. L. **VHDL**: programming by example. 4. ed. New York: McGraw-Hill, 2002. p. 497.

PFITSCHER, L. L; BERNARDON, D. P; KOPP, L. M; HECKLER, M. V. T.; THOMÉ, B.; MONTANI, P. Automação da irrigação para melhoria da eficiência energética em lavouras de arroz. In: CONGRESSO BRASILEIRO DE EFICIÊNCIA ENERGÉTICA, 4., 2011, Juiz de Fora. **Congresso...** Juiz de Fora: ABEE, 2011. p.1-7.

REQUEST FOR COMMENTS- RFC 793. **Transmission control protocol - DARPA internet program protocol specification**. Marina del Rey: IETF, 1981. Disponível em: <<http://tools.ietf.org/search/rfc793>>. Acesso: 17 dez. 2013.

REQUEST FOR COMMENTS- RFC 791. **Internet protocol - DARPA internet program protocol specification**, Marina del Rey: IETF, 1981. Disponível em: <<http://www.ietf.org/rfc/rfc791.txt>>. Acesso: 17 dez.2013.

ROLFES, C. **Side-channel analysis aspects of lightweight block ciphers**. 2009. 109 f. Tese (Doutorado)- Universität Bochum, Ruhr, 2009.

SADEGHI, S.; MASHAD, I.; YAGHMAEE, M. M. H.; BAHEKMAT, M; HEYDARI, Y. A. S. Modeling of smart grid traffics using non-preemptive priority queues. **Iranian Conference on Smart Grids**, Tehran, v. 2, n. 1, p. 1-4, 2012.

SANTOS, S. T. dos. **Redes de sensores sem fio em monitoramento e controle**. 2007. 87 f. Dissertação (Mestrado em Engenharia Elétrica)- Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007.

SHAH, P.; SHAIKH, T.; GHAN, K.; SHILASKAR, S. Power Management using zigbee wireless sensor network. In: INTERNATIONAL CONFERENCE ON EMERGING TRENDS IN ENGINEERING AND TECHNOLOGY- ICETET, 1., 2008, Nagpur. Conference... Nagpur: IEEE, 2008. p. 242-245.

TELESCADA. **Advanced metering solutions nexgen- AMI**. Pembroke: Telescada, 2010. Disponível em: <<http://www.telescada.com/Product%20Briefs/Telescada%20NeXGen%20AMI%20Triple%20Product%20Brief.pdf>>. Acesso: 17 dez. 2013.

ULLO, S. L.; VACCARO, A.; VELLOTO, G. Performance analysis of IEEE 802.15.4 based sensor networks for smart grids communications. **Journal of Electrical Engineering: Theory and Application**, Bratislava, v.1, n. 3, p. 129-134, 2010.

WANG, S.; ZHAO, H.; HUANG, S.; HAFID, A. Modeling FPGA-based IEEE 802.11 DCF. In: INTERNATIONAL CONFERENCE ON MOBILE AD-HOC AND SENSOR NETWORKS, 7., 2011, Beijing. Conference... Beijing: IEEE Computer Society, 2011. p. 194-200.

WANG, Y. **Wireless sensing and decentralized control for civil structures: theory and implementation**. 2007. 185 f. Tese Doutorado (Filosofia)- Department of Civil and environmental, Engineering and of Stanford University, EUA, 2007.

ZHOU, L.; RODRIGUES, J. J. P. C.; OLIVEIRA, L.M. QoE-driven power scheduling in smart grid: architecture, strategy, and methodology. **Communications Magazine IEEE**, New York: IEEE, v. 50, n. 5, p. 136-141, May 2012.

ZIGBEE ALLIANCE. **ZigBee specification**. San Ramon: Zigbee Alliance, 2007. Disponível em: <<http://www.zigbee.org/Standards/ZigBeeSmartEnergy/Specification.aspx>>. Acesso: 10 fev. 2014.

APÊNDICE A - ARTIGOS PUBLICADOS RELACIONADOS AO PRESENTE TRABALHO.

RAMALHO, L. A; SHINODA, A. A.; NASCIMENTO, V. E. do; OLIVEIRA, R. de; FERREIRA, E. T. Modeling of state machines in VHDL for encapsulation of DNP3 protocol in P2P Zigbee network. In: CONFERÊNCIA SOBRE INNOVATIVE SMART GRID TECHNOLOGIES LATIN AMERICA- ISGTLA, 2., 2013, São Paulo. **Conferência...** São Paulo: IEEE, 2013. p. 1-8.

RAMALHO, L. A; SHINODA, A. A.; NASCIMENTO, V. E. do; OLIVEIRA, R. de; FERREIRA, E. T. Modelagem em VHDL de encapsulamento do protocolo DNP3 em rede P2P IEEE 802.11b em modo Ad Hoc. In: LATIN-AMERICAN CONGRESS ON ELECTRIC POWER GENERATION, TRANSMISSION AND DISTRIBUTION- CLAGTEE, 9., 2013, Chile. **Congress...**Chile: CLAGTEE, 2013.

RAMALHO, L. A; SHINODA, A. A.; NASCIMENTO, V. E. do; OLIVEIRA, R. de; FERREIRA, E. T. Modelagem de Comandos de Sincronização DNP3 em rede P2P por interface IEEE 802.15.4. In: CONFERÊNCIA BRASILEIRA DE DINÂMICA, CONTROLE E APLICAÇÕES- DINCON, 11., Ceará, 2013. **Conferência ...** Ceará: SBAI, 2013. p. 1-6.

RAMALHO, L. A; SHINODA, A. A.; NASCIMENTO, V. E. do; OLIVEIRA, R. de; FERREIRA, E. T., Modelagem de Máquinas de Estado do Protocolo de Comunicação DNP3 do Smart Grid em VHDL. In: CONGRESSO BRASILEIRO DE QUALIDADE ENERGIA ELÉTRICA (X CBQEE), 10., Minas, 2013. **Congresso ...** Minas Gerais: SBQEE, 2013. p. 1-6.

RAMALHO, L. A; SHINODA, A. A.; NASCIMENTO, V. E. do; OLIVEIRA, R. de; FERREIRA, E. T. Uso de Linguagem de Descrição de Hardware e Dispositivos de Alto Desempenho na Educação Tecnológica. In: JORNADA CIENTIFICA DO IFMT, 1., 2013, Mato Grosso. **Jornada...**Mato Grosso: IFMT, 2013. p. 1-4.

RAMALHO, L. A; SHINODA, A. A.; NASCIMENTO, V. E. do; OLIVEIRA, R. de; FERREIRA, E. T. Modelagem do protocolo IEEE 802.11 em VHDL através de máquinas de estado. In: CONGRESSO DE MATEMÁTICA APLICADA E COMPUTACIONAL-CMAC-SE, 2., 2013, São Paulo. **Congresso ...** São Paulo: SBMAC, 2013. p. 1-6.

APÊNDICE B – CÓDIGOS VHDL

O Apêndice B tem o objetivo de mostrar os principais códigos VHDL utilizados na dissertação separando as funcionalidades conforme descrito abaixo:

- Códigos VHDL do protocolo DNP3 modelado;
- Códigos VHDL do protocolo IEEE 802.11b modelado;
- Códigos VHDL do protocolo IEEE 802.15.4 modelado;

Os códigos modelados estão disponíveis no link:

<https://www.dropbox.com/s/d6var1lg05lk97d/Apendice%20B%20Ramalho%202014.rar>