

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

TIAGO FERNANDES BORTH

**ANALISANDO OS IMPACTOS DO USO DO PROTOCOLO
CAN FD EM APLICAÇÕES AUTOMOTIVAS – ESTUDO DE
CASO**

Porto Alegre

2016

TIAGO FERNANDES BORTH

**ANALISANDO OS IMPACTOS DO USO DO PROTOCOLO
CAN FD EM APLICAÇÕES AUTOMOTIVAS – ESTUDO DE
CASO**

Dissertação de mestrado apresentada
ao Programa de Pós-Graduação em Engenharia Elétrica,
da Universidade Federal do Rio Grande do Sul, como
parte dos requisitos para a obtenção do título de Mestre
em Engenharia Elétrica.

Área de concentração: Controle e
Automação

ORIENTADOR: Prof. Dr. Carlos Eduardo Pereira

Porto Alegre

2016

TIAGO FERNANDES BORTH

**ANALISANDO OS IMPACTOS DO USO DO PROTOCOLO
CAN FD EM APLICAÇÕES AUTOMOTIVAS – ESTUDO DE
CASO**

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia Elétrica e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____

Prof. Dr. Carlos Eduardo Pereira, UFRGS

Doutor pela Technische Universität Stuttgart, Alemanha

Banca Examinadora:

Prof. Dr. João Cesar Netto, UFRGS

Doutor pela Université Catholique de Louvain, Bélgica

Prof. Dr. João Manoel Gomes da Silva Junior, UFRGS

Doutor pela Université Toulouse III Paul Sabatier, Toulouse, França

Prof. Dr. Marcelo Götz, UFRGS

Doutor pela Universität Paderborn, Paderborn, Alemanha

Coordenador do PPGEE: _____

Prof. Dr. Valner João Brusamarello

Porto Alegre, setembro de 2016

AGRADECIMENTOS

Agradeço à minha esposa, Mônica da Silva, pelo apoio, paciência, amor e compreensão. Agradeço também aos meus pais e à minha irmã pelo incentivo e pela confiança em mim depositada.

Agradeço ao meu orientador, Carlos Eduardo Pereira, pelo aceite em me orientar e pela dedicação e suporte prestados.

Agradeço ao colega Thiago Michelin, pelos ensinamentos em CANoe e pelos pequenos favores que fizeram a diferença.

Agradeço aos meus colegas de trabalho e aos diretores da empresa LOHR Sistemas Eletrônicos, pelo incentivo e pela liberação para realização das disciplinas durante horário de trabalho.

Por fim, agradeço à empresa Vector Informatik, por conceder licenças para utilização do programa CANoe e também pelo suporte técnico prestado.

RESUMO

O protocolo de comunicação CAN destaca-se há mais de duas décadas como um dos protocolos de comunicação mais utilizados na indústria automotiva. Porém a demanda cada vez maior de novas tecnologias embarcadas e as limitações do padrão (máximo de 8 bytes por mensagem e velocidade de até 1 Mbps) provocam a elevação da taxa de ocupação do barramento de comunicação a ponto de comprometer a sua viabilidade em determinadas aplicações. Em situações como esta, a adoção de protocolos de comunicação de maior velocidade (como FlexRay por exemplo) pode causar impactos indesejados aos projetos, como elevação de custos e necessidade de redimensionamento completo de sistemas já existentes. Com o intuito de atender parte desta demanda, surge como proposta o protocolo CAN FD, uma evolução do CAN clássico desenvolvida pela empresa BOSCH no ano de 2012 e a ser definitivamente normatizada apenas no segundo semestre de 2016. O protocolo CAN FD mantém algumas características do CAN clássico e sua maior inovação está na manipulação do campo de dados da mensagem, de modo a oferecer maior velocidade de transmissão (até 10 Mbps) e maior quantidade de dados por mensagem (até 64 bytes). Por tratar-se de um protocolo novo, ainda não implementado em sistemas veiculares, propõe-se neste trabalho a realização de estudo para avaliação do desempenho do novo padrão CAN FD em substituição a um sistema CAN tradicional. O sistema a ser avaliado, neste caso, trata-se de uma rede de comunicação CAN de um caminhão fora-de-estrada, o qual possui duas redes de comunicação SAE J1939, uma para o chassi e outra para a carroceria, interligadas por uma interface de comunicação. Este trabalho apresentará estudo de caso para aplicação do CAN FD no sistema proposto, através da simulação de mensagens CAN e CAN FD e avaliação das taxas de ocupação das linhas de comunicação e avaliação de possíveis atrasos relacionados à transmissão das mensagens.

Palavras-chave: CAN. CANFD. Taxa de Ocupação. Intervalo de Transmissão. SAE J1939.

ABSTRACT

CAN communication protocol stands out for more than two decades as one of the communication protocols most commonly used in the automotive industry. But the increasing demand of new embedded technologies and the CAN limitations (maximum of 8 bytes per message and a baud rate of until 1 Mbps) cause the elevation of the bus load to the point of compromising its viability in certain applications. In these situations, the adoption of faster protocols (such as FlexRay for example) can cause unwanted impacts on projects, such as higher costs and the need of redesigning existing systems. In order to meet part of this demand comes the new CAN FD protocol, an evolution of the classic CAN, developed by BOSCH company in 2012 and to be definitely normalized only in the second semester of 2016. The CAN FD protocol maintains some characteristics of classic CAN and as a further innovation, the improvement of the message data field, in order to provide higher transmission rates (up to 10 Mbps) and greater amount of data per message (up to 64 bytes). Considering that CAN FD is a new protocol, not yet implemented in vehicular systems, it's proposed in this work to perform a study to evaluate performance of the new CAN FD standard to replace the traditional CAN system. The system to be evaluated, in this case, is the CAN communication system of an off-road truck, which own two SAE J1939 communication networks, one for the chassis and one for the body, connected by communication interface. This report presents a case study for the application of CAN FD in the proposed system, by simulating CAN and CAN FD messages and evaluation of bus load and eventual delays related to the messages transmission.

Keywords: CAN. CANFD. Busload. Transmission Time. SAE J1939.

SUMÁRIO

| | | |
|------------|--|------------|
| 1 | INTRODUÇÃO | 12 |
| 2 | FUNDAMENTAÇÃO TEÓRICA..... | 15 |
| 2.1 | SISTEMAS DE COMUNICAÇÃO AUTOMOTIVOS..... | 15 |
| 2.2 | PROTOCOLOS DE COMUNICAÇÃO PARA APLICAÇÕES AUTOMOTIVAS | 17 |
| 3 | PROTOCOLO CAN FD | 38 |
| 3.1 | FORMATO DA MENSAGEM CAN FD..... | 39 |
| 3.2 | CAMADA FÍSICA PARA O CAN FD | 44 |
| 4 | ANÁLISE DO ESTADO DA ARTE | 47 |
| 4.1 | ANÁLISE TEMPO DE RESPOSTA DE MENSAGENS CAN | 47 |
| 4.2 | ANÁLISE DA TAXA DE OCUPAÇÃO DO BARRAMENTO CAN | 51 |
| 4.3 | TRABALHOS SIMILARES | 53 |
| 5 | ANÁLISE DE IMPACTO DE MIGRAÇÃO DE SISTEMAS PARA CAN FD | 56 |
| 5.1 | CAN FD E EMC..... | 56 |
| 5.2 | INTEGRAÇÃO DE PADRÕES CAN E CAN FD | 57 |
| 5.3 | IMPACTOS EM OUTROS PROTOCOLOS AUTOMOTIVOS..... | 60 |
| 6 | ESTUDO DE CASO: SISTEMA ELETRÔNICO DE UM CAMINHÃO FORA DE ESTRADA | 62 |
| 6.1 | DESCRIÇÃO DO SISTEMA ELETRÔNICO | 63 |
| 6.2 | ESPECIFICAÇÃO DAS MENSAGENS CAN | 66 |
| 6.3 | DEFINIÇÃO DOS ENSAIOS..... | 71 |
| 7 | SIMULAÇÕES E RESULTADOS | 75 |
| 7.1 | DESCRIÇÃO DO PROGRAMA DE SIMULAÇÃO..... | 75 |
| 7.2 | IMPLEMENTAÇÃO DO SISTEMA | 76 |
| 7.3 | ENSAIOS E RESULTADOS OBTIDOS..... | 85 |
| 7.4 | ANÁLISE DE RESULTADOS | 99 |
| 8 | CONCLUSÕES E TRABALHOS FUTUROS..... | 103 |
| | REFERÊNCIAS | 106 |
| | APÊNDICE: ALGORITMOS CAPL (CANOE)..... | 110 |
| A.1 | MENSAGENS ECU MOTOR | 110 |
| A.2 | MENSAGENS ECU TRANSMISSÃO..... | 112 |
| A.3 | MENSAGENS ECU ABS..... | 113 |
| A.4 | MENSAGENS ECU CABINE..... | 115 |
| A.5 | MENSAGENS ECU CARROCERIA | 116 |
| A.6 | MENSAGENS ECU CHAVE-SETA..... | 116 |
| A.7 | MENSAGENS GATEWAY | 117 |

LISTA DE ILUSTRAÇÕES

| | |
|---|----|
| Figura 1 Formato mensagem LIN (LIN CONSORTIUM, 2010)..... | 18 |
| Figura 2 <i>Transceiver</i> LIN e representação dos níveis lógicos (LIN CONSORTIUM, 2010).. | 19 |
| Figura 3 Representação de uma rede LIN (LIN CONSORTIUM, 2010) | 20 |
| Figura 4 Topologias de rede FlexRay (NATIONAL INSTRUMENTS, 2009) | 21 |
| Figura 5 Formato mensagem FlexRay (FLEXRAY CONSORTIUM, 2010)..... | 21 |
| Figura 6 Ciclo de comunicação FlexRay (NATIONAL INSTRUMENTS, 2009) | 22 |
| Figura 7 Relação entre o modelo OSI e os padrões ISO 11898 e ISO 11519 | 26 |
| Figura 8 Representação de uma rede CAN conforme ISO11898..... | 27 |
| Figura 9 Representação de sinal diferencial comunicação CAN (RICHARDS, 2016)..... | 27 |
| Figura 10 Relação da velocidade por comprimento do cabo CAN | 28 |
| Figura 11 Representação do tempo de propagação no padrão CAN | 29 |
| Figura 12 Arbitragem no barramento CAN..... | 29 |
| Figura 13 Formatos mensagens CAN (BOSCH GmbH, 1991)..... | 31 |
| Figura 14 Identificador CAN no padrão J1939 (JUNGER, 2010) | 34 |
| Figura 15 Especificação PGN65102 norma SAE J1939 (SAE INTERNATIONAL, 2006).... | 36 |
| Figura 16 Estrutura de mensagens no Protocolo de Transporte J1939 (JUNGER, 2010)..... | 37 |
| Figura 17 Taxa de decaimento da velocidade CAN clássico | 38 |
| Figura 18 Segmentos de uma mensagem CAN (CAN IN AUTOMATION, 2013)..... | 39 |
| Figura 19 Formatação das mensagens CAN e CAN FD | 40 |
| Figura 20 Formatos de Identificador mensagem CAN FD (BOSCH, 2011) | 41 |
| Figura 21 Transmissão de mensagem CAN usando <i>bit stuffing</i> (RENESAS, 2006) | 43 |
| Figura 22 Inserção de <i>stuff bits</i> fixos no campo de CRC (RENESAS, 2006)..... | 44 |
| Figura 23 Atraso de propagação assimétrico <i>transceiver</i> CAN (HELL, 2013) | 45 |
| Figura 24 Comparação de EMC em mensagens CAN de 500 kbps e 2 Mbps (LOPEZ; MOUNIER, 2015) | 57 |
| Figura 25 Barramento CAN e CAN FD conectados via <i>gateway</i> (ADAMSON, 2015) | 58 |
| Figura 26 Implementação da solução <i>FD Shield</i> (ADAMSON, 2015)..... | 59 |
| Figura 27 Filtragem de uma mensagem CAN FD pelo <i>FD Shield</i> (ADAMSON, 2015)..... | 60 |
| Figura 28 Relação entre quantidade de dados por velocidade no CAN, CAN FD e FlexRay (QUIGLEY, WILLIAMS, MCLAUGHLIN, 2013). | 61 |
| Figura 29 Conexão elétrica dos módulos do caminhão | 63 |
| Figura 30 Estruturação da rede para ensaio 1 | 72 |
| Figura 31 Estruturação da rede para ensaio 2..... | 72 |
| Figura 32 Estruturação da rede para ensaio 3..... | 73 |
| Figura 33 Estruturação da rede para ensaio 4..... | 74 |
| Figura 34 Estruturação da rede para ensaio 5..... | 74 |
| Figura 35 Rede CAN da Carroceria do Caminhão | 77 |
| Figura 36 Rede CAN do Chassi do Caminhão | 77 |
| Figura 37 Formatação de Mensagens CANFD com mais de 8 bytes | 82 |
| Figura 38 Taxa de ocupação ensaio 1..... | 86 |
| Figura 39: Rede CAN no CANoe para ensaio 2..... | 88 |
| Figura 40 Taxa de ocupação ensaio 2..... | 89 |
| Figura 41 Taxa de Ocupação ensaio 3..... | 90 |
| Figura 42 Taxa de ocupação das redes CAN FD e CAN | 92 |
| Figura 43 Taxa de ocupação ensaio 5 CAN FD | 94 |

| | | |
|-----------|--|-----|
| Figura 44 | Variação percentual intervalos de transmissão das mensagens CAN/CAN FD | 96 |
| Figura 45 | Distribuição períodos de transmissão PGN61442 (Ensaio 1 a 5)..... | 97 |
| Figura 46 | Distribuição períodos de transmissão PGN65265 (Ensaio 1 a 5)..... | 98 |
| Figura 47 | Representação do <i>Jitter</i> na transmissão de mensagens assíncronas..... | 99 |
| Figura 48 | Proposta de distribuição de PGNs em mensagem CAN FD | 104 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 Especificação do campo DLC no CAN FD | 41 |
| Tabela 2 Exemplos de <i>transceivers</i> CAN FD..... | 46 |
| Tabela 3 Mensagens Módulo Motor Scania | 66 |
| Tabela 4 Mensagens CAN Módulo de Transmissão do Motor. | 67 |
| Tabela 5 Mensagens CAN Módulo do Freio ABS | 68 |
| Tabela 6 Mensagens CAN Módulo da Cabine | 68 |
| Tabela 7 Mensagens CAN Módulo Carroceria | 69 |
| Tabela 8 Mensagens CAN Dispositivo Chave-Seta..... | 69 |
| Tabela 9 Agrupamento de Mensagens do Motor..... | 83 |
| Tabela 10 Agrupamento de Mensagens Transmissão | 84 |
| Tabela 11 Agrupamento de Mensagens Freio ABS | 84 |
| Tabela 12 Agrupamento de Mensagens Módulo da Cabine..... | 85 |
| Tabela 13 Agrupamento de Mensagens Módulo da Carroceria | 85 |
| Tabela 14 Agrupamento de Mensagens Módulo da Chave Seta | 85 |
| Tabela 15 Intervalos de Transmissão Ensaio 1 | 87 |
| Tabela 16 Intervalos de Transmissão Ensaio 2 | 89 |
| Tabela 17 Intervalos de transmissão ensaio 3 | 91 |
| Tabela 18 Intervalos de atraso para ensaio 4..... | 93 |
| Tabela 19 Intervalo de atraso ensaio 5 | 94 |
| Tabela 20 Taxa de ocupação da rede para diferentes ensaios | 95 |
| Tabela 21 Intervalos de atraso mensagens CAN ensaios 1 a 5 | 95 |
| Tabela 22 Distribuição períodos PGN61442 (Ensaio 1 a 5)..... | 97 |
| Tabela 23 Distribuição períodos PGN61442 (Ensaio 1 a 5)..... | 98 |
| Tabela 24 <i>Jitter</i> na transmissão de mensagens assíncronas | 99 |

LISTA DE ABREVIATURAS

ABS: Anti-lock Breaking System

BAM: Broadcast Announce Message

CAN: Controller Area Network

CAN FD: Controller Area Network with Flexible Data-Rate

CAPL: Communication Access Programming Language

CSMA/CA: Carrier Sense Multiple Access/Collision Avoidance

EDL: Extended Data Length

ECU: Electronic Control Unit

EMC: Electromagnetic Compatibility

ESI: Error State Indicator

IHM: Interface Homem Máquina

ISO: International Standardization Organization

LIN: Local Interconnect Network

LLC: Logical Link Control

MAC: Medium Access Control

NRZ: Non Return to Zero

PGN: Parameter Group Number

SAE: Society of Automotive Engineers

SOF: Start Of Frame

SPN: Suspect Parameter Number

TDMA: Time Division Multiple Access

UART: Universal Asynchronous Receiver/Transmitter

USB: Universal Serial Bus

1 INTRODUÇÃO

Idealizado pela empresa Robert Bosch GmbH na década de 80 e padronizado pela ISO (*International Organization for Standardization*) no começo da década de 90, o protocolo de comunicação CAN (*Controller Area Network*) destaca-se desde então como um dos padrões de comunicação de dados mais utilizados na linha automotiva leve (carros), pesada (caminhões e ônibus) máquinas agrícolas e construção civil (CAN IN AUTOMATION, 2013). Por apresentar características como comunicação bidirecional em um único cabo tipo par trançado, velocidade de transmissão de até 1 Mbps e boa imunidade a ruídos, este protocolo mostra-se bastante versátil para a maioria das aplicações de controle do chassi e carroceria dos veículos (BOSCH, 1991). O protocolo CAN também possui grande confiabilidade na transmissão de dados e bom desempenho em aplicações de tempo real a um baixo custo final.

Porém, com o rápido crescimento da indústria automotiva surgem também novas aplicações que acabam por aumentar a complexidade e a quantidade de dados nos barramentos de comunicação. Como resultado, alguns sistemas estão sujeitos a problemas como limitação na largura de banda e também aumento de carga na rede (HARTWICH, 2012). Neste caso, se a carga da rede é demasiadamente elevada, algumas mensagens de baixa prioridade podem sofrer atrasos consideráveis na transmissão devido ao mecanismo existente para evitar colisões de mensagens. Desta forma, o protocolo CAN pode acabar deixando de atender requisitos como intervalos de transmissão fixos, condição essencial para sistemas de tempo real.

Para solucionar este tipo de problema, é comum realizar a estruturação de redes de comunicação CAN em várias sub-redes distintas, ou então, migrar para outro protocolo de maior velocidade, como o FlexRay, por exemplo (DECKER, 2013). Neste caso, ambas as soluções acabam agregando desvantagens como elevação do custo, incompatibilidade no padrão de mensagens, entre outros. Em função destas limitações, criou-se então um novo

padrão, o CAN FD (*CAN with Flexible Data-Rate*), o qual oferece maior velocidade (até 10 Mbps) e maior quantidade de dados em uma única mensagem (até 64 bytes).

O desenvolvimento do CAN FD foi baseado no CAN clássico e tem como princípio básico prover maior velocidade na transmissão de dados, porém mantendo o meio físico sem alterações (HARTWICH, 2012). Para isto, manteve-se a estrutura original do campo de arbitragem da mensagem, com uma velocidade de comunicação mais baixa e, adicionalmente, elevou-se a velocidade e tamanho do campo de dados da mensagem. Este tipo de estratégia garante as mesmas características funcionais do CAN clássico.

Este trabalho tem como objetivo avaliar o desempenho do novo padrão CAN FD quando utilizado em substituição ao tradicional padrão CAN, mais especificamente nos barramentos de comunicação do chassi e carroceria de um caminhão fora-de-estrada, utilizado atualmente na indústria de mineração. Este processo de migração para o novo padrão CAN FD visa obter melhora no desempenho de transmissão das mensagens CAN, além de avaliar possibilidade de unificação das redes do chassi e carroceria, as quais são atualmente separadas por uma interface de comunicação, também conhecida como *gateway*. Como parte deste estudo, faz-se necessário verificar parâmetros como taxa de ocupação dos barramentos de comunicação CAN atuais (chassi e carroceria) e intervalos de envio das mensagens, as quais possuem modo de transmissão síncrona. As redes de comunicação do veículo estão em conformidade com o protocolo de aplicação SAE J1939, o qual especifica a formatação, taxa de comunicação e intervalos de transmissão das mensagens, além de outros aspectos relacionados ao meio físico da rede.

No decorrer deste trabalho serão apresentados resultados de ensaios realizados em diferentes configurações das linhas de comunicação utilizando os padrões CAN e CAN FD. As avaliações da taxa de ocupação da rede, assim como os tempos de atraso das mensagens CAN e CAN FD, serão realizadas através de simulações utilizando-se programa CANoe, da empresa

Vector Informatik GmbH (VECTOR, 2015). Desta forma, pode-se descrever a organização desta dissertação da seguinte forma: o Capítulo 2 aborda conceitos teóricos que dizem respeito a este trabalho, sobretudo o que envolve sistemas de comunicação automotivos. O Capítulo 3 traz uma abordagem mais completa sobre a definição teórica do protocolo CAN FD. O Capítulo 4 traz uma análise sobre trabalhos importantes desenvolvidos por outros autores e que possuem relevância quanto às análises de desempenho dos protocolos CAN e CAN FD. O Capítulo 5 realiza uma análise sobre os possíveis impactos na introdução e adoção do novo padrão FD. No Capítulo 6 é apresentada descrição do sistema avaliado e metodologia utilizada. No Capítulo 7 são demonstrados os resultados obtidos nas simulações efetuadas no desenvolvimento do trabalho. Por fim, o Capítulo 8 apresenta uma discussão sobre os resultados obtidos, assim como propostas e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Nas próximas seções serão descritos conceitos importantes e necessários para a apresentação do estudo realizado neste trabalho.

2.1 SISTEMAS DE COMUNICAÇÃO AUTOMOTIVOS

A comunicação de dados digitais em veículos automotores utilizando técnicas de multiplexação de sinais iniciou-se por volta de 1970 e, a partir de então, tornou-se algo corriqueiro e fundamental para a evolução dos sistemas embarcados. Segundo Lupini (2010), idealizava-se inicialmente que um único protocolo seria suficiente para atender as necessidades de um veículo, porém gradualmente os protocolos foram se expandindo até chegarmos a uma quantidade superior a 40 diferentes tipos de protocolos. Desta forma, torna-se necessário uma categorização destes padrões, trabalho este realizado pela SAE (*Society of Automotive Engineers*) e que inclui a definição de diferentes classes para a classificação destes protocolos. De acordo com (JURGEN, 1999), sistemas de comunicação automotivos podem ser divididos em três classes distintas:

- a) classe A: aplicações gerais de comunicação, geralmente diagnóstico ou controles secundários no veículo. Possui baixa velocidade, em torno de 10 Kbps, e baixo valor de custo na solução. Dentre os protocolos envolvidos nesta classe o LIN destaca-se como padrão mais utilizado pelos fabricantes na atualidade;
- b) classe B: para aplicações que não envolvam diagnósticos e comunicação de sistemas críticos no veículo. Velocidade variando entre 10 Kbps e 125 Kbps, com suporte a transmissão assíncrona (vinculada a eventos) ou periódica. Nesta classe o CAN destaca-se como protocolo de maior utilização, mais especificamente sob a regulamentação da ISO 11898-3, com velocidades entre

80 e 125 Kbps e suporte a detecção de falhas na rede e transmissão sobre um único fio;

- c) classe C: para aplicações que exigem maior velocidade e largura de banda da rede, tipicamente associados a sistemas em tempo real, como controle do motor, caixa de transmissão, controle de velocidade, entre outros. A taxa de transmissão varia entre 125 Kbps e 1 Mbps. Nesta classe o CAN também se destaca como protocolo de maior utilização, porém sob regulamentação da J1939 (250 Kbps, aplicação em ônibus e caminhões) ou da ISO 11898-2 (500 Kbps, aplicação em veículos leves).

Considerando-se que nem todas as funções de um veículo apresentam o mesmo desempenho ou, os mesmos requisitos de segurança, diferentes premissas de “Qualidade de Serviço” podem ser esperadas dos sistemas de comunicação veiculares, como tempo de resposta, *jitter*, largura de banda, mecanismos de detecção e resolução de erros, entre outros. Segundo (NAVET e SIMONOT-LION, 2009), sistemas automotivos embarcados podem tipicamente ser divididos em distintos domínios funcionais que correspondem a diferentes características e diferentes restrições do sistema. Destes, dois deles dedicam-se especificamente ao controle de funções de tempo real e segurança veicular: o *powertrain* (controle do motor e transmissão) e o chassi (controle de suspensão, direção e freios). O terceiro, a carroceria, tipicamente gerencia funções de conforto como iluminação e climatização, por exemplo. Domínios como telemática (sistemas de localização veicular), multimídia e Interfaces Homem Máquina (IHM) também ganham cada vez mais destaque nos sistemas veiculares.

2.1.1 Modos de transmissão

A transmissão de mensagens em sistemas de comunicação veiculares geralmente baseia-se em dois distintos modos: o modo *event-triggered* e o modo *time-triggered*. O primeiro modo

(*event-triggered*) vincula a transmissão de mensagens à ocorrência de eventos significantes, como a abertura da porta do veículo, por exemplo. Isto significa que a transmissão ocorre de forma assíncrona e por isto o protocolo de comunicação deve utilizar alguma estratégia para evitar colisões de mensagens na linha. De acordo com Navet e Simonot-Lion (2009), sistemas *event-triggered* são muito eficientes na otimização da largura de banda de linhas de comunicação, pois somente mensagens realmente necessárias são transmitidas.

Para comunicações do tipo *time-triggered*, as mensagens são transmitidas em intervalos de tempo pré-determinados, com valores que podem variar de acordo com a relevância da informação a qual se está transmitindo e do tempo de resposta necessário para executar determinadas funções do veículo. Navet e Simonot-Lion (2009) também afirmam que este tipo de estratégia garante a recepção das informações em caso de perda da mensagem no ciclo anterior. Por outro lado, este tipo de estratégia pode ser menos eficiente em termos de ocupação da linha de comunicação e tempos de resposta das mensagens.

O protocolo de aplicação que especifica a formatação das mensagens CAN (SAE-J1939) utiliza como estratégia principal o modo *time-triggered*, porém também existem mensagens as quais podem ser transmitidas tanto em modo *time-triggered* como *event-triggered*.

2.2 PROTOCOLOS DE COMUNICAÇÃO PARA APLICAÇÕES AUTOMOTIVAS

Nas seções a seguir serão apresentados conceitos referentes a protocolos de comunicação mais comumente utilizados nas linhas automotivas leve e pesada (carros, caminhões, ônibus entre outros).

2.2.1 Protocolo LIN

O protocolo LIN (*Local Interconnect Network*) é um padrão de rede para comunicação de dados de baixo custo, desenvolvido por um consórcio de empresas, a fim de ser utilizado em

aplicações embarcadas na linha automotiva. Este protocolo foi desenvolvido com o intuito de criar uma padronização para itens de baixo valor agregado na estrutura de comunicação de dados em veículos automotores. O padrão LIN é uma boa alternativa para aplicações que não requerem tanta versatilidade e largura de banda como aplicações CAN, por exemplo. Com este modelo de comunicação é possível implementar uma rede LIN utilizando microcontroladores de baixo custo, pois o único recurso necessário neste caso é apenas um canal de UART (LIN CONSORTIUM, 2010).

O padrão LIN é baseado em uma arquitetura mestre-escravo a qual consiste em um dispositivo mestre e até quinze dispositivos escravos. Uma mensagem LIN é composta por um cabeçalho e um campo de respostas, conforme pode ser visualizado na Figura 1.

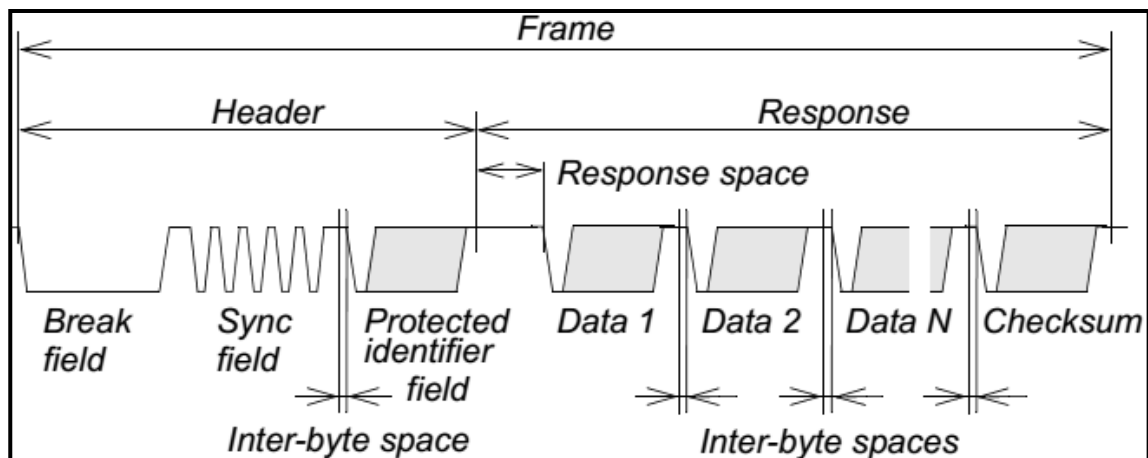


Figura 1 Formato mensagem LIN (LIN CONSORTIUM, 2010)

A iniciação da mensagem é feita pelo dispositivo mestre através da manutenção do barramento em nível lógico “0” durante 13 bits consecutivos (*BREAK*). Posteriormente o mestre envia o valor 85 (0x55 em hexadecimal) para que os dispositivos escravos consigam determinar a taxa de transmissão efetiva da mensagem (*SINCRONISMO*). O próximo valor a ser transmitido chama-se IDENTIFICADOR e serve para especificar o tipo de mensagem que se está transmitindo ou então determinar os nós na rede que devem receber ou responder cada mensagem. O padrão especifica um total de 64 valores para o campo do identificador, o qual é

transmitido utilizando os seis bits menos significativos, enquanto que os dois bits mais significativos são utilizados para transmitir a paridade do valor do identificador. No campo de respostas da mensagem podem ser transmitidos de 1 a 8 bytes de dados, tanto pelo dispositivo mestre como pelos dispositivos escravos. O último dado enviado é a verificação (*CHECKSUM*), o qual consiste na soma de todos os dados e do identificador, com a subtração por 255 cada vez que o valor for maior que 256 (LIN CONSORTIUM, 2010).

O meio físico utilizado pelo padrão LIN consiste basicamente em um único fio conectado através de um resistor de terminação ao positivo da bateria. O comprimento máximo da linha é de 40 metros e suporta taxas de transmissão de até 20 Kbps. O barramento opera com dois níveis lógicos:

- a) o valor dominante com uma tensão próxima a zero volts (terra) representa o valor lógico '0';
- b) o valor recessivo com uma tensão próxima à tensão da bateria representa o valor lógico '1';

A Figura 2 demonstra os níveis lógicos do barramento LIN bem como a representação típica de um *transceiver* LIN.

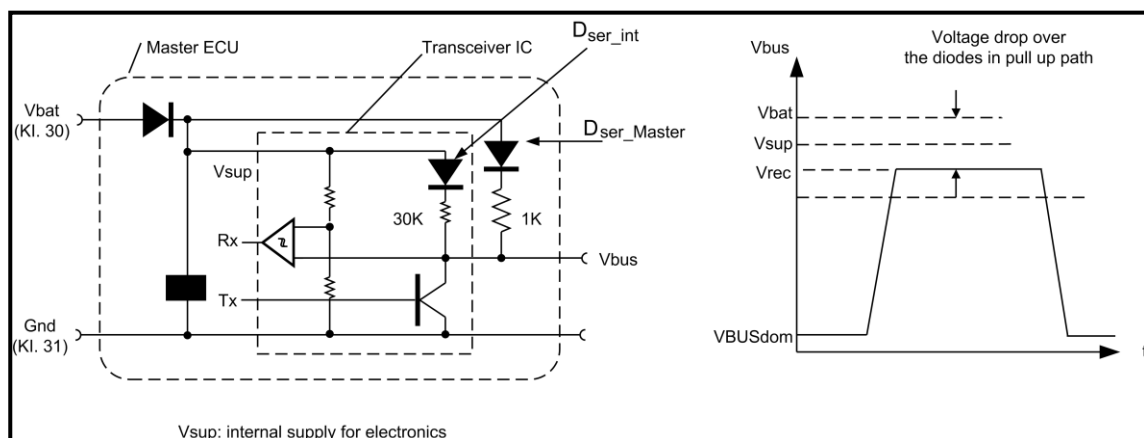


Figura 2 *Transceiver* LIN e representação dos níveis lógicos (LIN CONSORTIUM, 2010)

Pode-se afirmar que o padrão LIN não necessita lidar com colisões uma vez que o gerenciamento das mensagens na linha é sempre executado pelo mestre. Desta forma apenas um pacote de mensagens é transmitido por vez. As mensagens neste caso podem tanto ser transmitidas de modo síncrono (*time-triggered*) ou de modo assíncrono, ou seja, vinculadas a eventos (*event-triggered*) (LIN CONSORTIUM, 2010). Na Figura 3 pode-se visualizar uma rede LIN com o dispositivo mestre e os dispositivos escravos.

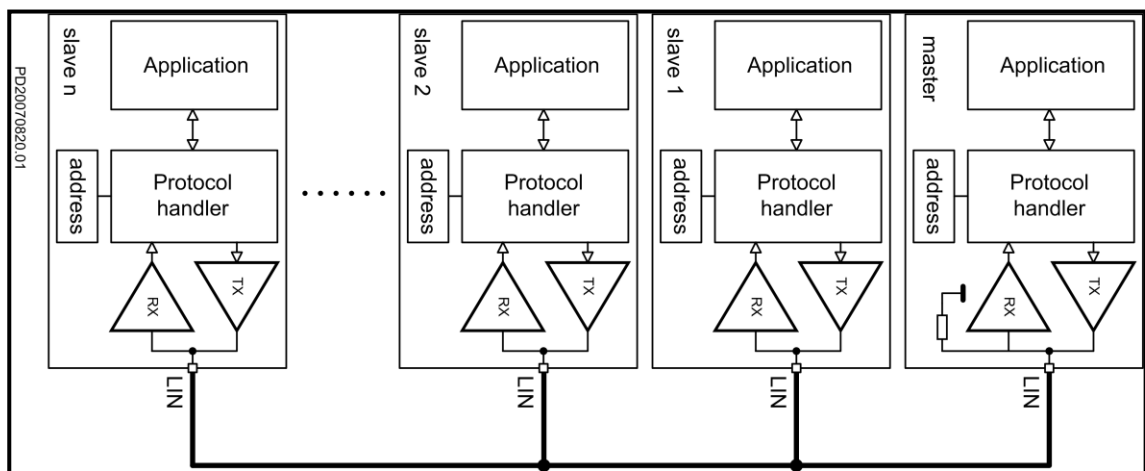


Figura 3 Representação de uma rede LIN (LIN CONSORTIUM, 2010)

2.2.2 Protocolo FlexRay

O protocolo FlexRay foi desenvolvido por um consórcio de fabricantes e fornecedores da cadeia automotiva com o intuito de criar uma rede de comunicação de dados que fosse capaz de atender as novas demandas na transmissão de dados entre os sistemas eletrônicos de um veículo. Esta demanda surge basicamente da necessidade de se aumentar velocidade e quantidade de dados para atender os mais modernos sistemas de segurança e controle do veículo, os quais requerem sincronização e desempenho superiores aos oferecidos por protocolos como CAN e LIN (NATIONAL INSTRUMENTS, 2009).

FlexRay utiliza como meio físico um par trançado não blindado para conexão dos nós e suporta configuração de um ou dois canais (um ou dois pares trançados, respectivamente). O

sinal transmitido é diferencial em cada par trançado, o que reduz o efeito de ruído externo na linha de comunicação. Configurações a dois canais também oferecem suporte melhorado a falhas na linha de comunicação e maior velocidade nas taxas de comunicação. Da mesma forma que outros padrões de comunicação, FlexRay também requer resistores de terminação ao final da linha de comunicação. A topologia de rede utilizada pode ser do tipo linear (*multi-drop*), tipo estrela ou híbrida (Figura 4). O protocolo FlexRay possui três taxas de comunicação padronizadas: 10 Mbps, 5 Mbps e 2.5 Mbps (FLEXRAY CONSORTIUM, 2010).

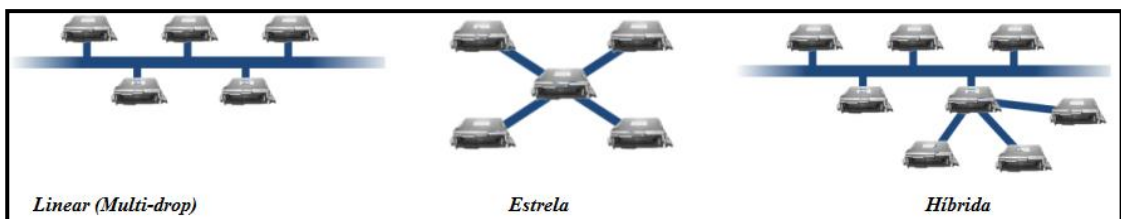


Figura 4 Topologias de rede FlexRay (NATIONAL INSTRUMENTS, 2009)

Na Figura 5 pode ser analisado o formato de uma mensagem FlexRay. A mensagem consiste de três segmentos, *Header Segment*, *Payload Segment* e *Trailer Segment*, transmitidos na linha de comunicação nesta mesma ordem (FLEXRAY CONSORTIUM, 2010).

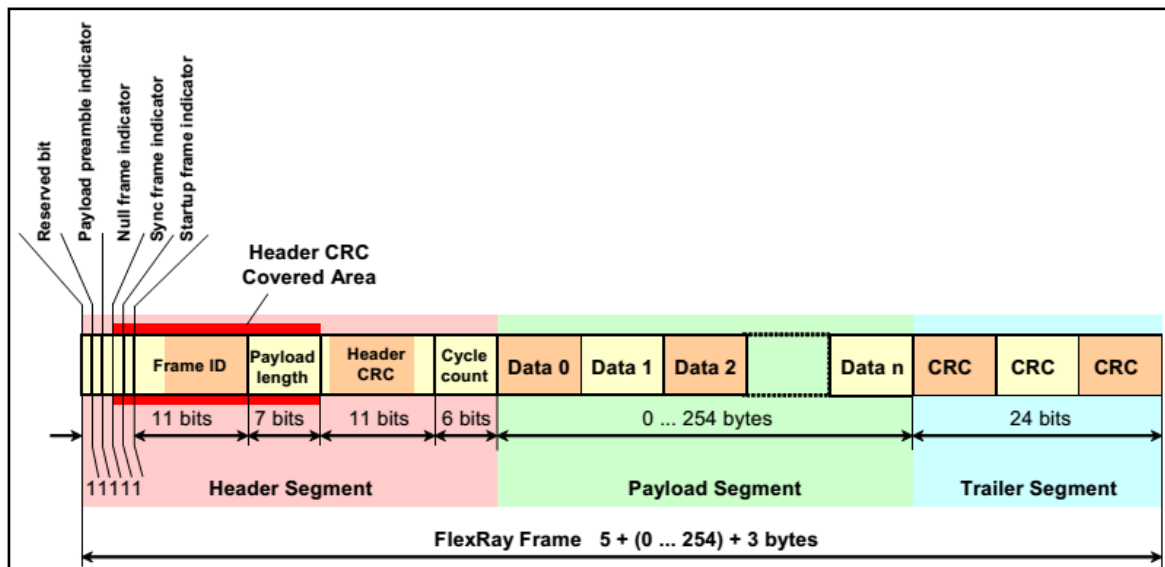


Figura 5 Formato mensagem FlexRay (FLEXRAY CONSORTIUM, 2010)

O protocolo FlexRay combina as estratégias *time-triggered* e *event-triggered* para a transmissão de mensagens na rede. Na primeira estratégia existe um tempo fixo pré-determinado em projeto, o qual especifica o intervalo de transmissão de uma mensagem enquanto que na segunda existe um tempo dinâmico variável, destinado às transmissões de mensagens baseadas em eventos. Outra característica importante do protocolo é o gerenciamento das mensagens via TDMA (*Time Division Multiple Access*), ou seja, cada nó de rede está sincronizado ao mesmo sinal de relógio (*clock*) e cada nó aguarda sua vez para escrever seus dados na mensagem de rede. Como resultado este protocolo consegue garantir determinismo e consistência na entrega dos dados aos nós da rede (NATIONAL INSTRUMENTS, 2009).

O ciclo de comunicação no FlexRay é o elemento fundamental para acesso ao meio. A duração de um ciclo é pré-determinada em projeto, mas tipicamente define-se algo entre 1 e 5 milissegundos. Em um ciclo de comunicação existem quatro diferentes segmentos, os quais podem ser visualizados na Figura 6.

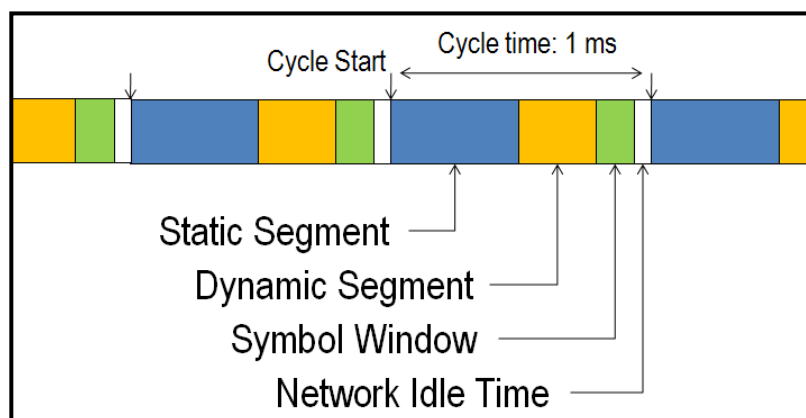


Figura 6 Ciclo de comunicação FlexRay (NATIONAL INSTRUMENTS, 2009)

O segmento estático (*static segment*) é o espaço no ciclo dedicado às mensagens do tipo *time-triggered*. Este segmento é subdividido em *slots*, cada um reservado para o envio de dados

por um módulo específico da rede. Desta maneira os dados somente são enviados a cada novo ciclo de uma mensagem, o que conseqüentemente os caracteriza como determinísticos.

O segmento dinâmico permite a transmissão ocasional de dados e possui um tamanho fixo. Por este motivo, há um limite na quantidade total de dados que pode ser alocada no segmento dinâmico para cada ciclo. A priorização de mensagens é pré-determinada por *minislots*, os quais geram um mecanismo de alocação de mensagens similar à arbitragem no protocolo CAN.

A janela de símbolo (*symbol window*) é inicialmente utilizada para manutenção e identificação de ciclos especiais. Aplicações de camadas superiores não interagem com esta janela. O tempo em repouso (*network idle time*) é pré-definido em projeto e conhecido pelos nós da rede.

2.2.3 Protocolo CAN

O padrão CAN é um protocolo de comunicação serial desenvolvido pela empresa BOSCH para aplicações na indústria automotiva. Quando foi desenvolvido, tinha como objetivo principal substituir e simplificar arranjos elétricos que utilizavam grande quantidade de fios e cabos por um único barramento de comunicação de dados. O CAN é um protocolo do tipo *broadcast*, ou seja, todos os nós de um barramento de comunicação estão aptos a receber as mensagens transmitidas no mesmo. Este protocolo suporta eficientemente sistemas distribuídos de tempo real e, dentre as aplicações no qual o CAN se enquadra, pode-se citar desde redes de alta velocidade (1 Mbps) até sistemas de controle multiplexados de baixo custo (BOSCH, 1991).

O CAN possui ampla utilização em sistemas de controle na linha automotiva leve (automóveis), automotiva pesada (caminhões e ônibus), máquinas agrícolas, construção civil e

indústria em geral. O fato de ser amplamente utilizado pode ter relação com as seguintes características (ANDERSSON e TOVAR, 2009):

- a) microcontroladores e controladores CAN estão disponíveis a um baixo custo;
- b) permite a utilização de cabos de comunicação de baixo custo;
- c) a rede CAN é flexível à novas configurações, ou seja, novos nós podem ser incluídos na rede sem a necessidade de alteração dos nós já existentes;

Nas seções seguintes serão abordadas maiores características do protocolo CAN.

2.2.3.1 Características gerais do protocolo CAN

O protocolo CAN utiliza o método CSMA/CA (*Carrier Sense Multiple Access/Collision Avoidance*) para gerenciamento de mensagens na rede, ou seja, antes de realizar a transmissão, o controlador CAN verifica se o barramento está em estado de repouso para somente assim iniciar a transmissão (PARET, 2007). Caso dois ou mais módulos iniciem a transmissão simultânea de um pacote de dados, persistirá na linha a mensagem que possuir o menor valor numérico no campo da arbitragem. No momento em que o barramento passar para o estado de repouso, ocorrerá então nova tentativa de transmissão das mensagens que estavam na fila. Nas seções seguintes será realizada uma abordagem com maiores detalhes da estratégia de arbitragem das mensagens CAN.

Outra característica importante do protocolo CAN é o fato de ele ser NRZ (*Non Return to Zero*), ou seja, o valor de um bit (0 ou 1) sempre é representado por um valor de tensão diferente de zero. Destacam-se no protocolo CAN as seguintes propriedades (BOSCH, 1991):

- a) priorização de mensagens;
- b) garantia nos tempos de latência;
- c) flexibilidade na configuração;
- d) recepção de mensagens por todos os nós com tempo de sincronização;

- e) detecção de erros e sinalização;
- f) retransmissões automáticas no caso de mensagens corrompidas;
- g) configuração de rede com arranjo multimestre;
- h) reconhecimento de leitura da mensagem por pelo menos um nó na rede.

Assim como outros protocolos, o CAN também está de acordo com o modelo OSI (*Open Systems Interconnection*), o qual é definido em sete distintas camadas. Neste caso, encontram-se especificadas pela ISO apenas as camadas inferiores do modelo, ou seja, a camada física e a camada de enlace de dados. As camadas físicas e de enlace de dados são padronizadas pela ISO 11898 (125 Kbps a 1 Mbps) e ISO 11519 (10 Kbps a 125 Kbps). Acima destas camadas encontra-se diretamente a camada de aplicação, a qual possui distintas padronizações como CANOpen, SAE J1939, SAE-J2284, entre outros.

A camada física define especificamente como os sinais são transmitidos, o meio físico utilizado (cabos e conectores), as características dos circuitos integrados de recepção e transmissão (*transceivers*) e a forma como os bits são codificados, temporizados e sincronizados dentro de uma mensagem. A camada de dados é subdividida em camada MAC (*Medium Access Control*) e LLC (*Logical Link Control*). A subcamada MAC gerencia o acesso ao barramento, sendo responsável pela serialização e deserialização dos dados, detecção e sinalização de erros e codificação da mensagem segundo os padrões do protocolo. A subcamada LLC é responsável por controlar filtros de aceitação de mensagens bem como monitorar o estado da rede e das mensagens em execução. A Figura 7 demonstra a relação entre as 7 camadas do modelo OSI e os padrões ISO 11898 e ISO 11519.

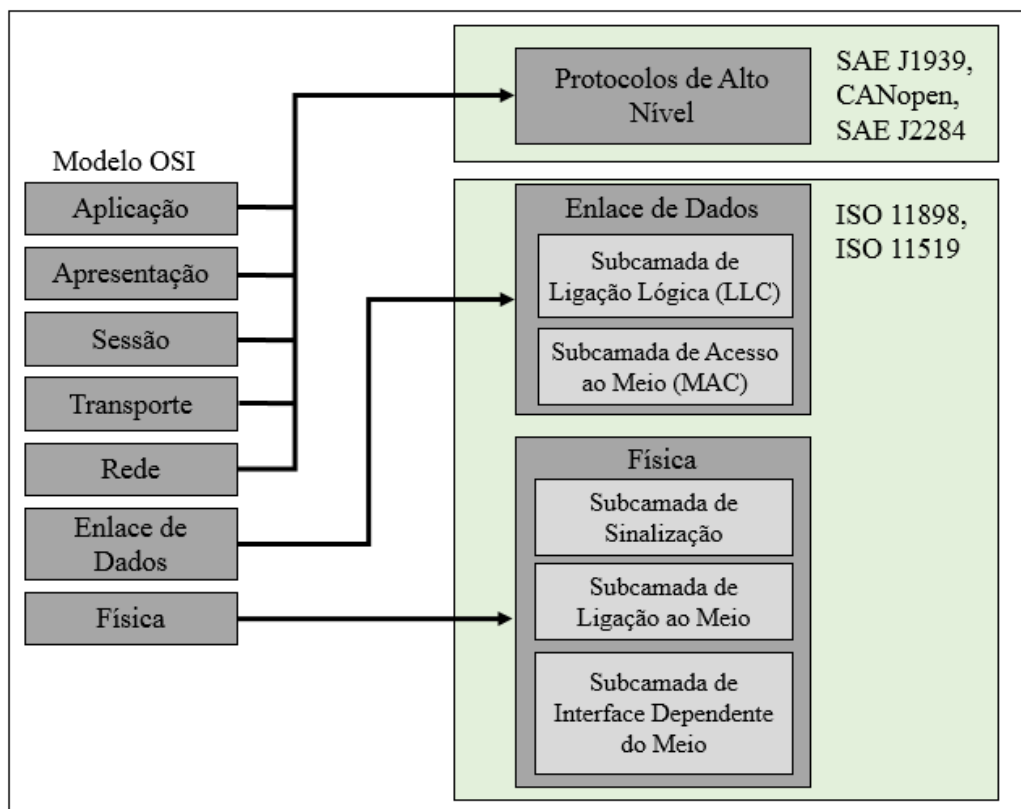


Figura 7 Relação entre o modelo OSI e os padrões ISO 11898 e ISO 11519

2.2.3.2 Camada física de rede CAN

A especificação ISO11898-2, padrão utilizado para redes CAN de alta velocidade, descreve uma rede CAN tradicional conforme indica a Figura 8. A linha de comunicação é constituída basicamente por dois fios trançados e dois resistores de terminação de 120Ω nas extremidades da rede. Ao longo da linha de comunicação estão conectados os módulos eletrônicos responsáveis pela transmissão e recepção das mensagens CAN. Os módulos, por sua vez, são constituídos basicamente por um *transceiver*, e um microcontrolador (ou microprocessador) com controlador CAN integrado. O *transceiver* tem como função realizar a conversão dos níveis de tensão da linha CAN para níveis de tensão do controlador CAN e vice-versa. O controlador CAN pode ser descrito como a interface entre a aplicação e a linha de comunicação, ou seja, o controlador tem como função básica converter os dados fornecidos

pela aplicação em uma mensagem formatada para ser transmitida pela rede CAN (LATTICE, 2013).

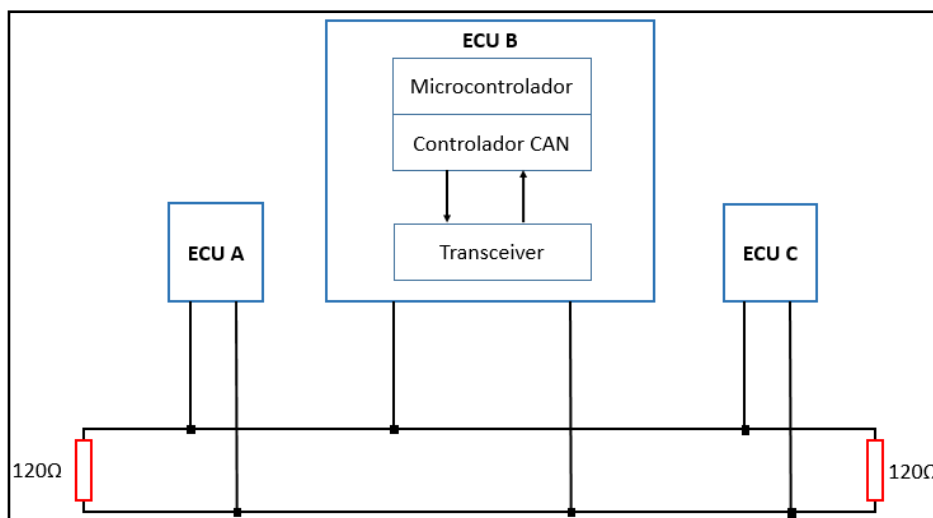


Figura 8 Representação de uma rede CAN conforme ISO11898

Em uma rede CAN a transmissão dos sinais elétricos é diferencial, o que garante grande imunidade a ruídos elétricos e boa tolerância a falhas (CORRIGAN, 2008). O protocolo especifica dois níveis lógicos: recessivo (bit lógico '1') e dominante (bit lógico '0'). A Figura 9 representa o sinal diferencial de uma linha de comunicação CAN. No estado recessivo a tensão dos sinais CANH e CANL apresenta um valor aproximado de 2.5V e no estado dominante apresenta um valor aproximado de 3.5V para CANH e 1.5V para CANL.

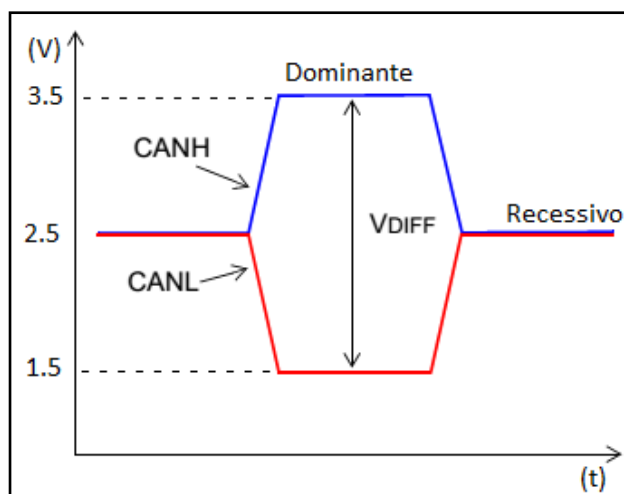


Figura 9 Representação de sinal diferencial comunicação CAN (RICHARDS, 2016)

O padrão ISO11898 também indica que o comprimento máximo do cabo de comunicação deve ser de 40 metros para uma taxa de transmissão de 1 Mbps. Maiores distâncias podem ser alcançadas utilizando-se taxas de comunicação inferiores a 1 Mbps, conforme pode ser verificado na Figura 10.

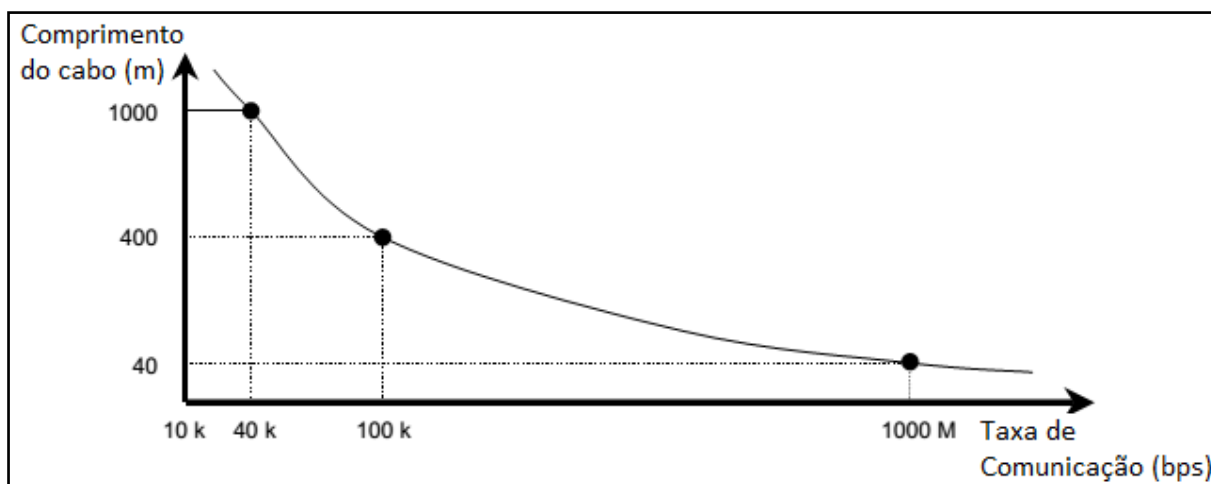


Figura 10 Relação da velocidade por comprimento do cabo CAN

Pode-se afirmar que o comprimento de uma rede CAN está diretamente relacionado com o tempo de propagação dos sinais, mais especificamente, o método de arbitragem não destrutivo do protocolo (CSMA/CA) que acaba sendo afetado pelos atrasos da comunicação (RICHARDS, 2002). Durante o processo de comunicação, cada nó conectado à rede CAN deveria ser capaz de amostrar os distintos níveis lógicos dos *bits* ao mesmo tempo, porém quando os módulos se encontram a uma distância elevada, existe a possibilidade de que a amostragem dos bits ocorra em tempos distintos (Figura 11). Desta forma, tempos de propagação extremos (além do tempo do ponto de amostragem) irão resultar em um processo de arbitragem inválido e consequentes falhas na transmissão e recepção de mensagens.

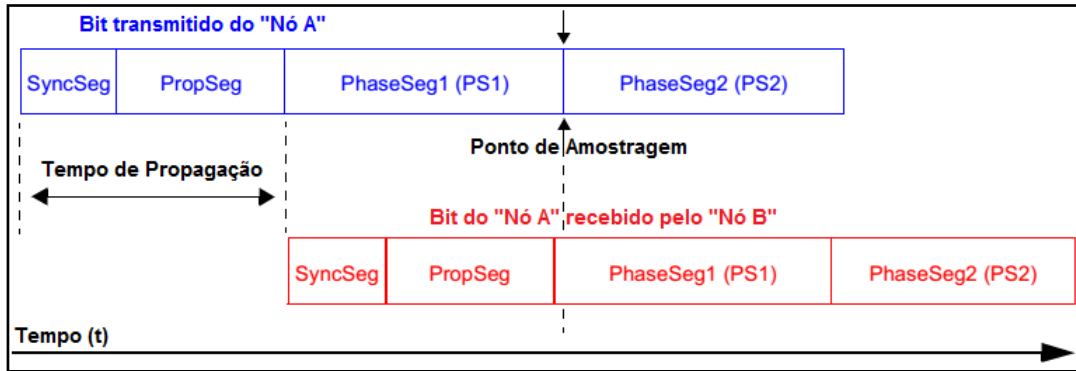


Figura 11 Representação do tempo de propagação no padrão CAN

O processo de resolução da arbitragem ocorre através da priorização da mensagem que possuir maior quantidade de bits em nível dominante ('0') no início da transmissão da mensagem. Quando um nó efetua a transmissão de um pacote de dados, ele deve ser capaz de transmitir e monitorar os sinais ao mesmo tempo. Quando o nó transmite um bit dominante, ele deve ler na linha o valor lógico referente ao mesmo bit dominante ('0'). Quando o nó transmite um bit recessivo e lê na linha de comunicação o valor lógico referente a um bit dominante, significa que ocorreu um conflito na transmissão com alguma mensagem de outro nó. Neste momento o nó que possui a mensagem menos prioritária aguarda o barramento ficar livre e retransmite a mensagem. A Figura 12 representa o funcionamento do esquema de arbitragem de mensagens CAN: três nós distintos iniciam a transmissão simultânea de suas mensagens e por fim prevalece na linha de comunicação apenas a mensagem transmitida pelo "Nó 2", a qual possui maior número de bits dominantes no identificador de sua mensagem.

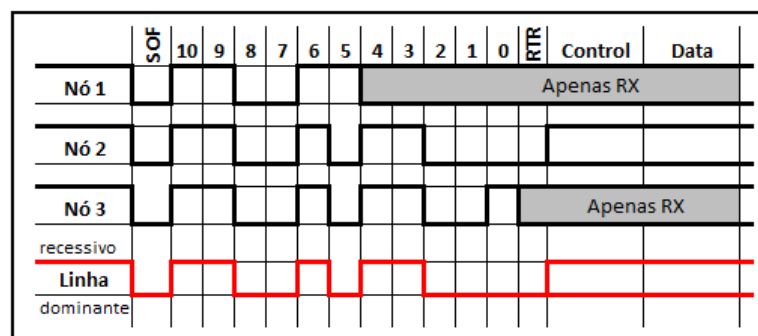


Figura 12 Arbitragem no barramento CAN

2.2.3.3 Tipos de mensagens CAN

Em um barramento de comunicação podem ser encontrados os seguintes tipos de mensagens CAN (RENESAS, 2006):

- a) *data frame*: mensagem utilizada para transmissão de dados de uso geral dos módulos da rede;
- b) *remote frame*: mensagem utilizada por algum módulo para solicitar da rede mensagem com identificador específico;
- c) *error frame*: quando um módulo detecta um erro na linha ele envia esta mensagem para notificar os demais módulos do erro detectado;
- d) *overload frame*: esta mensagem é utilizada por um módulo receptor para notificar que ele ainda não está preparado para receber outras mensagens;
- e) *interframe space*: é utilizado para separar uma mensagem tipo *Data Frame* ou *Remote Frame* da próxima mensagem a ser transmitida na linha de comunicação.

2.2.3.4 Data Frame CAN

O padrão CAN permite a utilização de dois tipos de formatos de mensagens: formato padrão (CAN 2.0A) e o formato estendido (CAN 2.0B) (BOSCH, 1991). A principal diferença entre estes dois formatos é a quantidade de bits presentes no campo identificador, ou seja, no formato padrão utiliza-se um campo de 11 bits enquanto que no formato estendido utiliza-se um campo de 29 bits. Na Figura 13 pode-se visualizar os dois formatos de uma mensagem CAN, além da representação dos demais campos da mensagem:

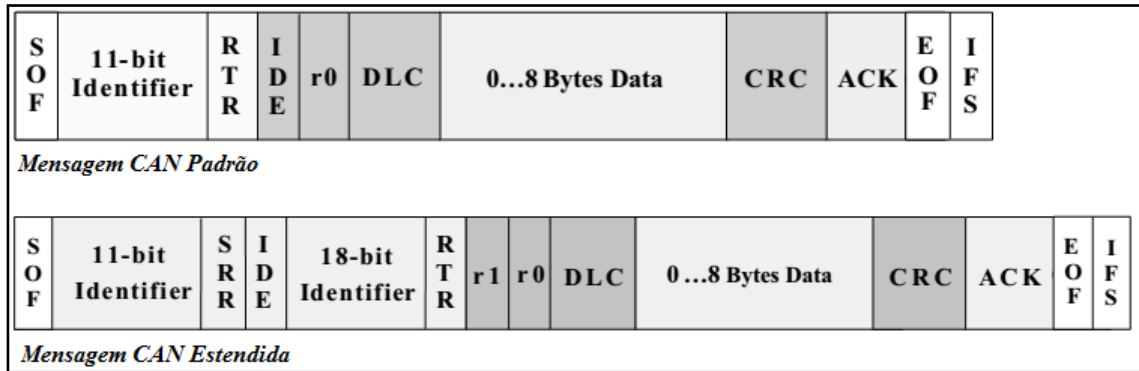


Figura 13 Formatos mensagens CAN (BOSCH GmbH, 1991)

Os campos de uma mensagem CAN apresentados na Figura 13 podem ser descritos como:

- a) SOF (*Start of Frame*): bit dominante responsável pela iniciação de uma mensagem e sincronização inicial dos módulos da rede;
- b) IDENTIFIER: o campo identificador estabelece a prioridade da mensagem, ou seja, quanto menor o valor, maior a prioridade da mesma;
- c) RTR (*Remote Transmission Request*): este bit é dominante quando uma informação é requisitada de outro nó na rede;
- d) SRR (*Substitute Remote Request*): este bit, utilizado apenas no formato estendido, é sempre recessivo e tem como função apenas alocar o espaço utilizado pelo RTR em mensagens padrão;
- e) IDE (*Identifier Extension Bit*): o bit dominante representa uma mensagem do tipo padrão, enquanto que o bit recessivo indica uma mensagem do tipo estendida;
- f) r1,r0: bits reservados para aplicações futuras, transmitidos na mensagem como valores dominantes;
- g) DLC (*Data Length Code*): contém o número de bytes de dados transmitidos na mensagem;

- h) *DATA*: campo que contém os dados da mensagem (até 8 bytes);
- i) *CRC (Cyclic Redundancy Check)*: campo que contém o byte de validação da mensagem (*checksum*);
- j) *ACK (Acknowledge)*: campo o qual indica o “reconhecimento” de uma mensagem, ou seja, quando algum nó receptor reconhece a mensagem como sendo válida, ele sobrescreve o bit originalmente recessivo para um bit dominante;
- k) *EOF (End of Frame)*: campo de sete bits recessivos o qual indica a finalização da mensagem;
- l) *IFS (Interframe Space)*: tempo em que a linha deve ficar em repouso para transmissão de nova mensagem, o que corresponde a sete bits de uma mensagem;

2.2.3.5 Método de *Bit Stuffing* no CAN

O processo de *bit stuffing* refere-se à estratégia utilizada no protocolo CAN para resincronizar periodicamente operações de transmissão e recepção e assim prevenir erros de interpretação das mensagens. O *bit stuffing* consiste basicamente em sempre transmitir um bit inverso aos 5 bits anteriores, caso eles tenham todos a mesma polaridade (RENESAS, 2006).

Quando um módulo da linha transmite uma mensagem CAN e o mesmo nível de sinal ocorre por 5 bits consecutivos (podendo ocorrer desde o campo de SOF até o campo de CRC), um bit de sinal inverso é adicionado à mensagem ocupado a posição do sexto bit consecutivo. Quando um módulo recebe uma mensagem e o mesmo nível de sinal ocorre em 5 bits consecutivos (também podendo ocorrer do SOF até o CRC), o sexto bit (*stuff bit*) é deletado da mensagem. Caso o sexto bit possua o mesmo nível dos 5 bits anteriores, detecta-se então um erro de *stuffing* na rede.

Segundo (NOLTE, HANSSON e NORSTRÖM, 2003), a quantidade máxima de *stuff* bits de uma mensagem pode ser dada pela equação 1, onde “*g*” representa a quantidade de bits antes do campo de dados sujeitos ao mecanismo de *stuffing* (34 para mensagens CAN 2.0A e 54 para mensagens CAN 2.0B) e L_i representa a quantidade de bytes no campo de dados. Como resultado, verifica-se uma quantidade de 24 bits para uma mensagem CAN 2.0A e 29 bits para uma mensagem CAN 2.0B.

$$n = \frac{g+8L_i-1}{4} \quad (1)$$

2.2.4 Protocolo CAN FD

Por se tratar de assunto de maior relevância para este trabalho, o protocolo CAN FD será abordado especificamente no Capítulo 3.

2.2.5 Protocolo de Aplicação SAE J1939

O protocolo J1939 é um conjunto de padronizações para sistemas CAN, baseado na especificação ISO11998 e elaborado pela *Society of Automotive Engineers* (SAE) para projetos de sistemas eletroeletrônicos de veículos pesados como caminhões e ônibus. Em outras palavras, pode-se afirmar que a J1939 é um guia de práticas recomendadas que define os tipos de dados e como os mesmos são transmitidos entre as unidades de controle eletrônico (ECU ou *Electronic Control Units*) em uma linha de comunicação do veículo (JUNGER, 2010).

Algumas características importantes do protocolo J1939:

- a) identificador CAN estendido de 29 bits;
- b) taxa de comunicação padronizada em 250 Kbps;
- c) comunicação ponto a ponto e do tipo *broadcast*;
- d) protocolo de transporte de mensagens suportando envio de até 1785 bytes de dados;

- e) gerenciamento da rede e diagnóstico do sistema;
- f) definição de grupos de parâmetros para veículos comerciais e outros;

Os grupos de parâmetros, denominados como PGN (*Parameter Group Number*), são um conjunto de informações referentes a um grupo de funções ou à alguma função específica do veículo. A quantidade de dados de um PGN não é limitada pelo comprimento de uma mensagem CAN. Tipicamente um PGN tem no mínimo 8 bytes e no máximo 1785 bytes, ou seja, quando o PGN é maior do que 8 bytes (quantidade máxima de dados de uma mensagem CAN), utiliza-se um método definido como protocolo de transporte.

2.2.5.1 Formato das mensagens J1939

Muitas das mensagens definidas pela J1939 são do tipo *broadcast*, ou seja, as mensagens são anunciadas na linha de forma que todos os módulos tenham acesso à leitura da mesma. Esta estratégia permite que qualquer ECU tenha acesso aos dados da mensagem sem a necessidade de efetuar a requisição da mesma. Este protocolo utiliza um identificador de 29 bits, conforme especificado na revisão do padrão CAN 2.0B, no qual encontram-se informações específicas, como o endereço do módulo transmissor (*Source Address*), prioridade (*Priority*), bit de página de dados (*Data Page*), bit de página de dados estendida (*Extended Data Page*) e endereço de destino (apenas para mensagens ponto a ponto). Na Figura 14 é apresentada a estrutura dos bits que formam o identificador de uma mensagem CAN no formato J1939.

| Priority | Extended Data Page | Data Page | PDU Format | PDU Specific | Source Address |
|----------|--------------------|-----------|------------|--------------|----------------|
| 3 bit | 1 bit | 1 bit | 8 bit | 8 bit | 8 bit |

Figura 14 Identificador CAN no padrão J1939 (JUNGER, 2010)

Os três primeiros bits do identificador da mensagem são utilizados para controlar a prioridade da mensagem durante o processo de arbitragem, neste caso a maior prioridade é dada

para o valor '0'. Valores de alta prioridade são usualmente atribuídos a mensagens de maior velocidade, como por exemplo mensagens de controle de torque da transmissão para o motor. Mensagens que contém dados de menor criticidade, como velocidade do veículo, por exemplo, recebem valores de menor prioridade. Os bits de *Data Page* da mensagem servem como índice de expansão da mensagem, ou seja, para uma mesmo PGN utilizam-se diferentes valores de “página” que permitem aumentar a quantidade de dados transmitidos para um mesmo grupo de mensagens (SAE INTERNATIONAL, 2006).

Os dois campos formados por *PDU Format* e *PDU Specific* constituem o valor que será atribuído ao PGN da mensagem e também definem o modo como as mensagens são endereçadas na linha. Para valores de *PDU Format* entre 0 e 239, transmite-se a mensagem para um endereço indicado pelo valor de *PDU Specific*. Para valores de *PDU Format* entre 240 e 255, a mensagem será somente do tipo *broadcast* e o dado contido no campo *PDU Specific* fornece um índice de referência para a mensagem chamado de *Group Extension*. A formatação do PGN agrupa estes dois valores, por exemplo, para uma determinada mensagem com *PDU Format* de valor 232 (0xE8 em hexadecimal) e *PDU Specific* de valor 220 (0xDC em hexadecimal), o PGN da mensagem terá o valor 59612 (0xE8DC em hexadecimal).

Todas as informações especificadas pela norma são apresentadas de forma descritiva, conforme pode ser verificado na Figura 15, que representa a especificação do PGN 65102 para controle da porta do veículo. Nela verificam-se as informações referentes aos dados pertencentes ao identificador da mensagem, taxa de transmissão da informação e especificação das informações existentes no campo de dados da mensagem. As informações presentes no campo de dados são descritas na norma através de um índice numérico denominado SPN (*Suspect Parameter Number*) (SAE INTERNATIONAL, 2006).

| PGN 65102 (R) Door Control 1 | | | - DC1 |
|-------------------------------|--------|----------------------------------|-------|
| Used for door information. | | | |
| Transmission Repetition Rate: | 100 ms | | |
| Data Length: | 8 | | |
| Extended Data Page: | 0 | | |
| Data Page: | 0 | | |
| PDU Format: | 254 | | |
| PDU Specific: | 78 | PGN Supporting Information: | |
| Default Priority: | 6 | | |
| Parameter Group Number: | 65102 | (0xFE4E) | |
| Start Position | Length | Parameter Name | SPN |
| 1.1 | 4 bits | Position of doors | 1821 |
| 1.5 | 2 bits | Ramp / Wheel Chair Lift Position | 1820 |
| 1.7 | 2 bits | Status 2 of doors | 3411 |

Figura 15 Especificação PGN65102 norma SAE J1939 (SAE INTERNATIONAL, 2006)

2.2.5.2 Protocolo de Transporte

Mensagens J1939 que possuem mais do que 8 bytes de dados são transmitidas na forma de pacotes de dados, ou seja, são necessárias n mensagens para transmitir a informação na linha CAN. O limite de transmissão neste caso é de 1785 bytes de dados, divididos em 255 distintos pacotes. Para realizar este tipo de estratégia, torna-se necessário realizar o gerenciamento da rede, de forma que os demais módulos permaneçam sem enviar suas mensagens e permitam a transmissão plena da mensagem via protocolo de transporte. O protocolo estabelece então dois gerenciamentos distintos para transmissões ponto a ponto e transmissões do tipo *broadcast*. Transmissões ponto a ponto utilizam-se dois PGNs distintos, um para gerenciamento da conexão (TP.CM) e outro para transmissão dos dados (TP.DT). Em transmissões do tipo *broadcast* utiliza-se um comando denominado BAM (*Broadcast Announce Message*). Em transmissões do tipo ponto-a-ponto, o transmissor inicia a conexão através do comando TP.CM_RTS (*Request to Send*). O receptor então controla o protocolo de transporte utilizando os comandos TP.CTS (*Clear to Send*) e TP.EoMA (*End of Message Acknowledge*). Os intervalos de transmissão das mensagens no protocolo de transporte também são definidos pela

J1939 e utilizam valores mínimos de 50 milissegundos. A Figura 16 demonstra a estrutura sequencial de envio de mensagens no protocolo de transporte para mensagens *broadcast* e ponto a ponto.

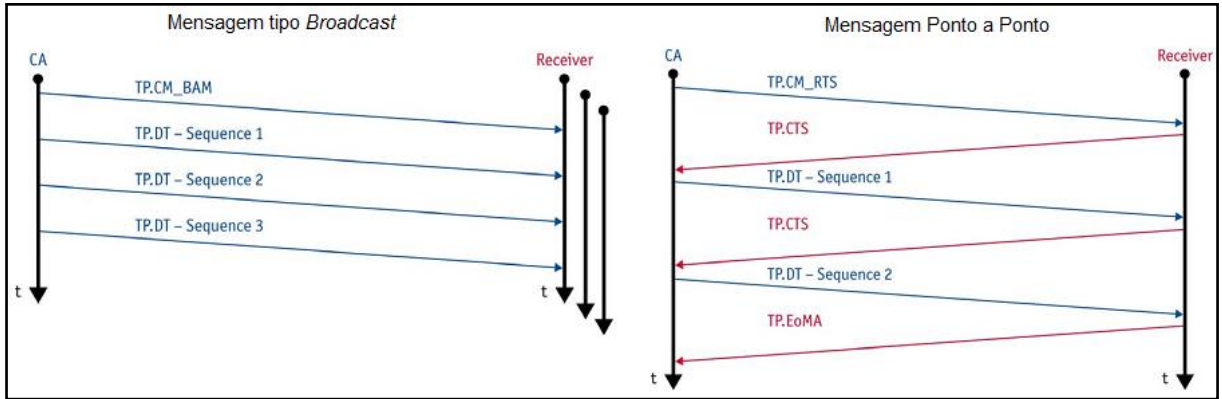


Figura 16 Estrutura de mensagens no Protocolo de Transporte J1939 (JUNGER, 2010)

3 PROTOCOLO CAN FD

O CAN FD (*CAN with Flexible Data-Rate*) é uma evolução do clássico padrão CAN e surgiu a fim de atender novas demandas da indústria automotiva, mais especificamente, a necessidade de uma maior quantidade e maior velocidade de transmissão de dados (BOSCH, 2011). O seu desenvolvimento foi iniciado no ano de 2011 pela empresa Bosch em parceria com fabricantes automotivos e, apenas no ano de 2015, foi revisado e publicado pela ISO 11898-1. As principais inovações do CAN FD incluem o aumento da taxa de transmissão de dados para velocidades superiores a 1 Mbps e o aumento na quantidade de dados transmitidos de 8 para até 64 bytes.

Atualmente o CAN clássico possui limitações na largura de banda (até 1Mbps) em função do atraso de propagação de uma mensagem na linha (LENNARTSSON, 2015). A Figura 17 demonstra a relação de velocidade da mensagem pelo comprimento do cabo de comunicação.

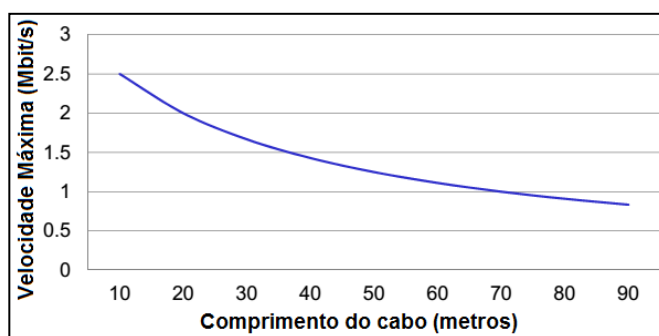


Figura 17 Taxa de decaimento da velocidade CAN clássico

O atraso de propagação da mensagem CAN afeta os campos de arbitragem (*Arbitration Phase*) e reconhecimento (*ACK Phase*), porém não limita os campos intermediários da mensagem (Figura 18). Como consequência deste efeito verifica-se que é possível aumentar a velocidade do pacote de dados, desde que a transmissão já tenha sido iniciada e os nós da rede já tenham sido sincronizados (LENNARTSSON, 2015).

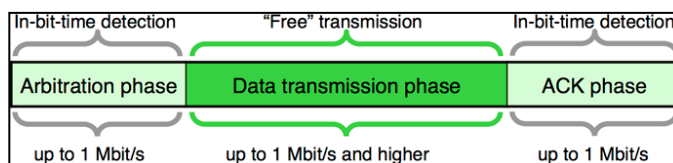


Figura 18 Segmentos de uma mensagem CAN (CAN IN AUTOMATION, 2013)

Através da manutenção dos mecanismos de arbitragem e reconhecimento do padrão de mensagens CAN clássico, torna-se possível aumentar a velocidade de transmissão dos demais campos da mensagem: *Data Length Code*, *Data Field* e *Checksum* (CRC). Além do aumento da taxa de transmissão, também é possível aumentar a quantidade de dados transmitidos e inclusive, diminuir o tamanho da mensagem (tempo de ocupação na rede).

3.1 FORMATO DA MENSAGEM CAN FD

A mensagem CAN FD possui os mesmos elementos de uma mensagem CAN, a principal diferença é que no padrão CAN FD o campo de dados e o campo de CRC podem ser maiores que no clássico padrão CAN. A validação de uma mensagem CAN FD requer, assim como no CAN clássico, a inserção de um bit de reconhecimento (*Acknowledge Bit*) por pelo menos um dos módulos da linha (BOSCH, 2011).

Como os dois padrões possuem basicamente a mesma estrutura, pode-se afirmar que a diferenciação entre uma mensagem CAN e uma CAN FD ocorre inicialmente no bit reservado “*res*”, localizado logo após a arbitragem da mensagem (Figura 19). No CAN FD este bit é substituído pelos bits EDL (*Extended Data Length*), “*r0*”, BRS (*Bit Rate Switch*) e ESI (*Error State Indicator*). Na Figura 19 pode ser visualizado o campo destacado da mensagem CAN FD, a qual representa os bits com taxa de transmissão distinta de uma mensagem CAN tradicional. Nas próximas seções será melhor descrita a composição dos bits que compõem a mensagem CAN FD.

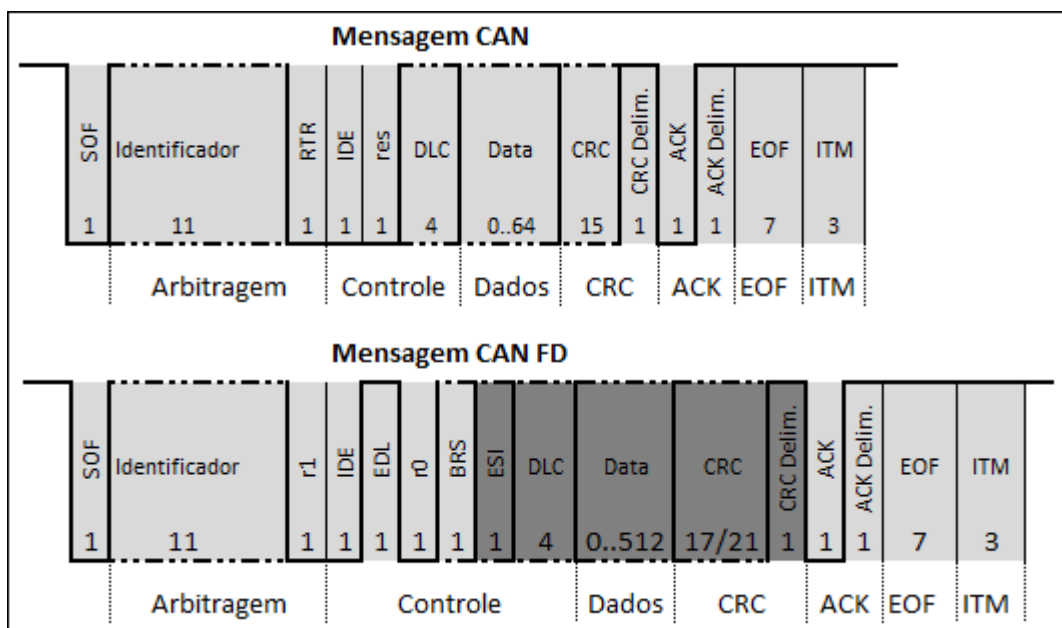


Figura 19 Formatação das mensagens CAN e CAN FD

3.1.1 SOF, Arbitragem e Controle

Da mesma forma que no CAN clássico, no CAN FD o bit de SOF (*Start of Frame*) possui um valor dominante que inicializa o pacote de dados e sincroniza a recepção da mensagem.

A primeira parte de uma mensagem CAN FD, até o bit BRS, é transmitida na mesma taxa de comunicação de uma mensagem CAN clássica. O bit BRS separa a arbitragem do campo de dados e indica se os dados serão transmitidos na mesma taxa de comunicação da arbitragem ou se a taxa de comunicação dos dados será alternada para valores maiores. Caso o bit BRS apresente o valor recessivo na mensagem, a velocidade dos bits seguintes será aumentada até o bit delimitador de CRC. Caso o bit BRS apresente o valor dominante os dados serão transmitidos na mesma velocidade do campo de arbitragem.

Assim como no CAN clássico, o CAN FD também comporta a transmissão de mensagens com identificadores de 11 bits e 29 bits (formato estendido). Em ambos os formatos (padrão de 11 bits e estendido de 29 bits), a troca de velocidade é efetuada no bit BRS, conforme

indica a Figura 20. O bit ESI é o bit que indica se o nó transmissor é do tipo “erro-ativo” (bit dominante) ou “erro-passivo” (bit recessivo).

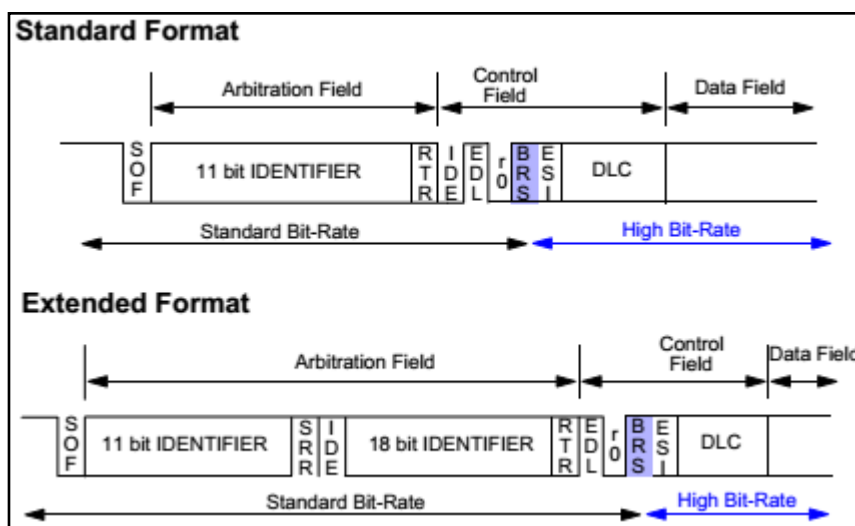


Figura 20 Formatos de Identificador mensagem CAN FD (BOSCH, 2011)

Outra particularidade da mensagem CAN FD refere-se ao campo DLC (*Data Length Code*), o qual utiliza 4 bits distintos para indicar a quantidade de bytes transmitidas no campo de dados da mensagem. Diferentemente do CAN clássico, que comporta até 8 bytes de dados, o CAN FD permite a transmissão de até 64 bytes, conforme demonstra a Tabela 1.

Tabela 1 Especificação do campo DLC no CAN FD

| Campo de DLC | | | | |
|--------------|------|------|------|-------|
| DLC3 | DLC2 | DLC1 | DLC0 | Dados |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 2 |
| 0 | 0 | 1 | 1 | 3 |
| 0 | 1 | 0 | 0 | 4 |
| 0 | 1 | 0 | 1 | 5 |
| 0 | 1 | 1 | 0 | 6 |
| 0 | 1 | 1 | 1 | 7 |
| 1 | 0 | 0 | 0 | 8 |
| 1 | 0 | 0 | 1 | 12 |
| 1 | 0 | 1 | 0 | 16 |

| Campo de DLC | | | | |
|--------------|------|------|------|-------|
| DLC3 | DLC2 | DLC1 | DLC0 | Dados |
| 1 | 0 | 1 | 1 | 20 |
| 1 | 1 | 0 | 0 | 24 |
| 1 | 1 | 0 | 1 | 32 |
| 1 | 1 | 1 | 0 | 48 |
| 1 | 1 | 1 | 1 | 64 |

3.1.2 Campo de CRC

O campo de CRC, o qual está presente na mensagem após a transmissão do campo de dados, também possui diferenças em relação ao CAN clássico. O CRC, ou *checksum*, trata-se de um valor calculado pelo módulo transmissor e inserido na mensagem CAN para posterior comparação com o valor determinado pelo módulo receptor a partir dos dados recebidos na mensagem e de um polinômio conhecido. Para o CAN clássico utilizam-se 15 bits no campo de CRC, enquanto que no CAN FD utilizam-se 17 bits para mensagens menores que 16 bytes no campo de dados e 21 bits para mensagens com mais de 16 bytes no campo de dados. A equação 22 representa o polinômio utilizado para cálculo do CRC do CAN clássico (BOSCH, 1991), a equação 23 representa o polinômio utilizado para cálculo do CRC de 17 bits do CAN FD e a equação 24 representa o polinômio utilizado para cálculo do CRC de 21 bits do CAN FD (HARTWICH, 2012).

$$X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^4 + X^3 + 1 \quad (22)$$

$$X^{17} + X^{16} + X^{14} + X^{13} + X^{11} + X^6 + X^4 + X^3 + X^1 + 1 \quad (23)$$

$$X^{21} + X^{20} + X^{13} + X^{11} + X^7 + X^4 + X^3 + 1 \quad (24)$$

3.1.3 Método de *Bit Stuffing* no CAN FD

Assim como no CAN clássico, CAN FD também utiliza uma técnica de inserção de *stuff bits* na mensagem. Neste método o transmissor da mensagem insere, sempre após 5 bits consecutivos de mesmo valor, um bit adicional de valor inverso, chamado *stuff bit*. O receptor da mensagem, neste caso, verifica a presença do *stuff bit* e o remove da mensagem após a recepção da mesma (Figura 21). O objetivo deste método é garantir que existam níveis de descida e subida suficientes na mensagem para ressincronizar os receptores CAN (HARTWICH; MUTTER, 2015).

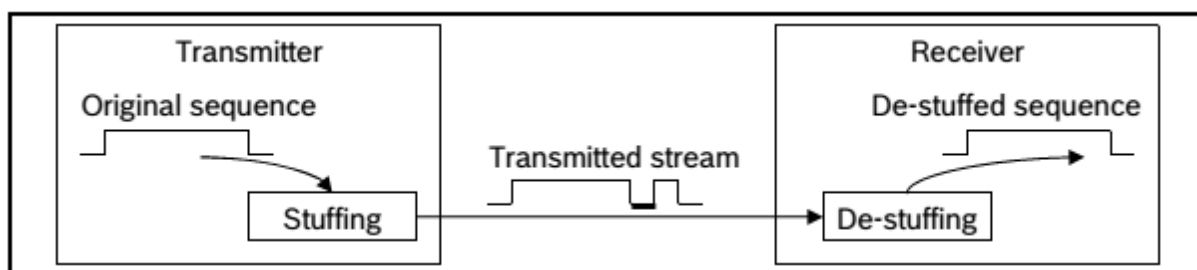


Figura 21 Transmissão de mensagem CAN usando *bit stuffing* (RENESAS, 2006)

O número de *stuff bits* transmitidos em uma mensagem tem relação direta com o valor dos dados contidos na mensagem e nos casos mais extremos pode aumentar o comprimento da mensagem em até 25%. Para estas situações extremas, existe a possibilidade de módulos receptores interpretarem de maneira errônea a combinação dos *stuff bits* e gerarem assim falsos bits na leitura da mensagem. Mensagens do CAN clássico não incluem estes bits adicionais no cálculo de CRC, porém no CAN FD estes bits dinâmicos são considerados (HARTWICH; MUTTER, 2015).

Um método diferente de *stuff bit* é utilizado no campo de CRC do CAN FD. Este método consiste na inserção de 3 *stuff bits* permanentes no campo de CRC, inclusive se o sequenciamento dos bits anteriores não atender a condição de 5 bits consecutivos de mesmo valor. Estes *stuff bits* ocupam posições fixas no campo de CRC (Figura 22) e sempre possuem

o valor inverso ao bit anterior. No receptor os *stuff bits* são verificados e removidos do campo de CRC antes de efetuar a verificação do valor de *checksum*. Esta técnica garante que no máximo 5 bits de mesmo valor serão transmitidos no campo de CRC da mensagem.

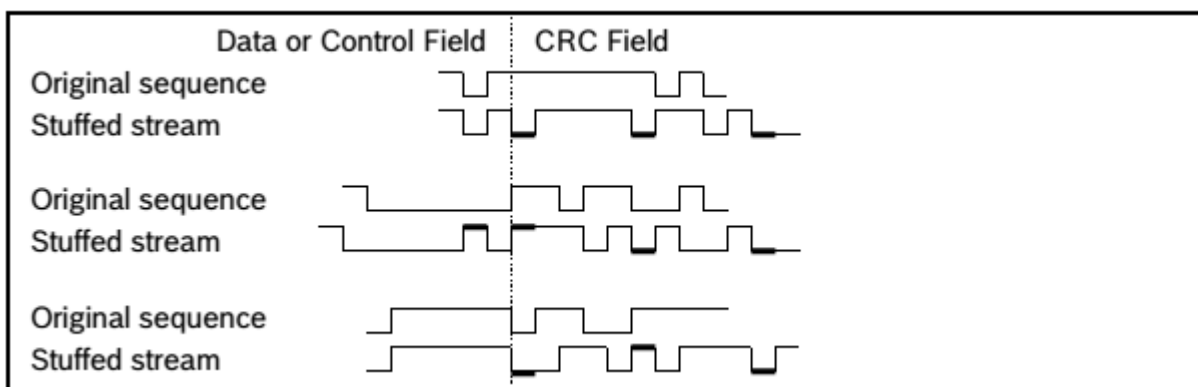


Figura 22 Inserção de *stuff bits* fixos no campo de CRC (RENESAS, 2006)

Outra inovação agregada com a introdução do CAN FD é a contagem do número de *stuff bits* antes do campo de CRC. A contagem é realizada pelo transmissor da mensagem e é transmitida no começo do campo de CRC utilizando codificação *Gray* (com um bit de paridade). Os bits de *stuff* fixos do CRC não são calculados. Após o recebimento da mensagem, os módulos comparam o valor recebido da quantidade de bits de *stuff* e comparam com o valor calculado. Caso os valores sejam distintos, o receptor gera um erro de CRC e trata o pacote de dados como inválido.

3.2 CAMADA FÍSICA PARA O CAN FD

Atualmente as especificações da camada física do CAN clássico são determinadas pelo padrão internacional ISO 11898-2/5/6, porém para o CAN FD ainda não há uma revisão definitiva desta norma. Está previsto o lançamento de nova revisão da ISO 11898-2 para o final de 2016, documento este que contemplará parâmetros adicionais para o CAN FD.

Mesmo não havendo ainda uma normatização formal para a especificação da camada física do CAN FD, existem algumas características relevantes já mencionadas na bibliografia.

A taxa de transmissão máxima do CAN FD por exemplo, possui limitantes que dificilmente permitirão o uso de velocidades superiores a 5 Mbps (HELL, 2013).

Durante a fase de arbitragem da mensagem CAN, quando dois ou mais módulos competem pela ocupação da rede para transmissão da mensagem, a velocidade de comunicação acaba sendo limitada por atrasos de propagação na rede, atrasos de propagação nos *transceivers* e reflexões dos sinais. Depois da fase de arbitragem, os atrasos de propagação entre os nós da rede não são mais importantes, porém as variações de largura de bit, causadas por características da rede e desempenho do *transceiver*, devem ser consideradas. Para casos em que estes atrasos são assimétricos (Figura 23), verifica-se redução da faixa de amostragem do bit CAN (*sampling point*), o que pode vir a causar falhas na decodificação das mensagens.

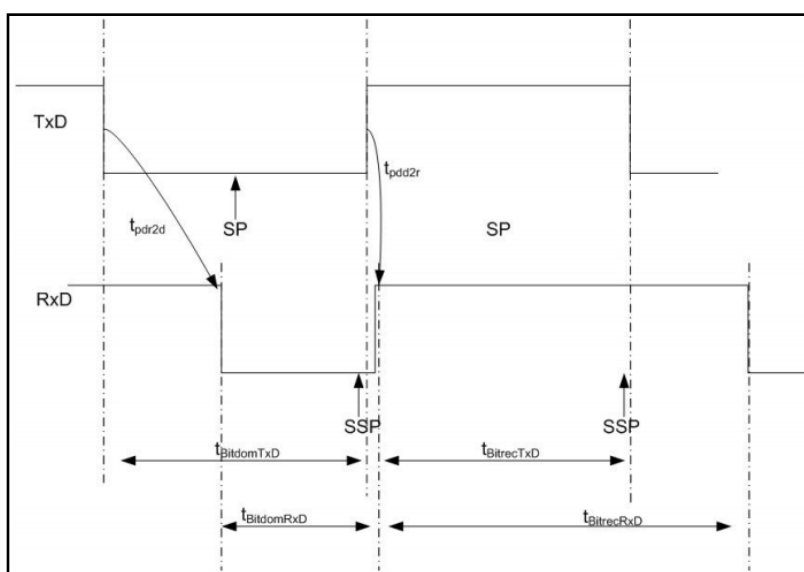


Figura 23 Atraso de propagação assimétrico *transceiver* CAN (HELL, 2013)

Para manutenção da estabilidade de uma rede CAN FD de alta velocidade, recomenda-se (HELL, 2013):

- a) utilização de *transceivers* especificados para o padrão CAN FD;
- b) utilização de barramento de comunicação com estrutura linear e ramificações curtas;

c) baixa capacitância no barramento de comunicação e nos módulos CAN;

Até o momento não existe ainda determinação sobre quais os valores de velocidade devem ser adotados no campo de dados de uma mensagem CAN FD, porém duas taxas de transmissão estão em discussão: 2 Mbps e 5 Mbps. A determinação destes valores torna-se importante para que os fabricantes de semicondutores possam oferecer *transceivers* adequados às taxas de comunicação a serem realmente utilizadas pelos fabricantes automotivos (HELL, 2013). Apesar de não haver ainda uma definição formal que indique as velocidades a serem adotadas no CAN FD, alguns fabricantes já disponibilizam soluções de *transceivers* para este padrão e que comportam velocidades de transmissão superiores a 1 Mbps. Na Tabela 2 são apresentadas algumas soluções de *transceivers* CAN FD recentemente disponibilizados para comercialização.

Tabela 2 Exemplos de *transceivers* CAN FD

| Componente | Velocidade | Fabricante |
|-------------|------------|------------|
| NCV7351F | 2 Mbps | OnSemi |
| SN65HVD26 | 2 Mbps | TI |
| MCP2561/2FD | 2 Mbps | Microchip |
| TJA1145TK | 2 Mbps | NXP |
| TLE7250SJ | 2 Mbps | Infineon |
| TLE9251VSJ | 5 Mbps | Infineon |
| ATA6560 | 5 Mbps | ATMEL |

4 ANÁLISE DO ESTADO DA ARTE

O protocolo CAN destaca-se atualmente como uma tecnologia bastante difundida em produtos e equipamentos da linha automotiva e industrial. Após mais de 25 anos de especificação do padrão, o CAN ofereceu a engenheiros e pesquisadores inúmeros temas para estudos e desenvolvimentos na área. Já o padrão CAN FD, por outro lado, é um protocolo novo, com poucas aplicações comerciais e ainda pouco difundido na comunidade acadêmica.

Tendo em vista estes aspectos, serão apresentados nas próximas seções estudos relacionados ao desempenho do padrão CAN para aplicações de tempo real, análises de tempos de resposta das mensagens e técnicas de escalonamento de mensagens. Após apresentação dos temas relacionados à utilização e desempenho do padrão CAN clássico, serão descritos os estudos até então realizados abordando o padrão CAN FD e que possuem relevância com este trabalho.

4.1 ANÁLISE TEMPO DE RESPOSTA DE MENSAGENS CAN

O modelamento temporal do CAN clássico tem como principais referências os trabalhos realizados por (TINDELL e BURNS, 1994) e (TINDELL, BURNS e WELLINGS, 1995).

No primeiro trabalho os autores propõem uma análise sobre os algoritmos utilizados em sistemas de tempo real de processadores computacionais e demonstram como esta análise pode ser aplicada a controladores CAN. No segundo trabalho, os autores aprofundam as análises sobre os tempos de transmissão de mensagens CAN em intervalos regulares e demonstram como determinar o pior caso do tempo de resposta de uma mensagem específica. Ambos os trabalhos utilizam o controlador CAN Intel 82527 como referência para determinação dos modelos apresentados. Para especificação dos modelos, os autores apresentam as seguintes considerações:

- a) o tempo de entrega (*deadline*) de uma mensagem "*m*" (denotado por D_m) não pode ser maior que o período da mensagem (denotado por T_m);
- b) o controlador CAN não deve permitir a transmissão de mensagens de baixa prioridade nos casos em que a transmissão de mensagens de alta prioridade está pendente;
- c) o pior caso no tempo de resposta de uma dada mensagem "*m*" é denotada por R_m e definida como o maior tempo entre a colocação da mensagem "*m*" na fila de transmissão e o momento em que as mensagens chegam nos nós receptores;

Desta forma, o tempo de resposta R_m de uma mensagem "*m*", no pior dos casos, é dado por (TINDELL e BURNS, 1994):

$$R_m = J_m + w_m + C_m \quad (2)$$

Os parâmetros da equação 2 podem ser definidos como:

- a) J_m é especificado como o atraso de processamento e representa o intervalo para processar a informação física e preparar a mensagem para ser alocada na fila de transmissão;
- b) w_m é o tempo de aguardo na fila de transmissão, no pior dos casos;
- c) C_m é o tempo total de transmissão de todos os bits da mensagem;

Posteriormente, (TINDELL, BURNS E WELLINGS, 1995), suprimem o elemento J_m da equação 2 e a apresentam novamente conforme equação 3:

$$R_m = w_m + C_m \quad (3)$$

O termo C_m , especificado como o tempo de transmissão de uma mensagem CAN padrão de identificador de 11 bits, pode ser representado pela equação 4, incluindo os bits da arbitragem e bits de *stuff*.

$$C_m = \left(\left\lceil \frac{34+8s_m}{5} \right\rceil + 47 + 8s_m \right) \tau_{bit} \quad (4)$$

O termo " s_m " da equação 4 representa a quantidade de bytes no campo de dados da mensagem. O termo τ_{bit} representa o período de 1 bit no barramento, por exemplo, para uma velocidade de 1 Mbps, o τ_{bit} é igual a $1\mu s$.

O termo w_m é representado pela equação 5:

$$w_m^{n+1} = B_m + \sum_{\forall j \in hp(m)} \left[\frac{w_m^n + J_j + \tau_{bit}}{T_j} \right] C_j \quad (5)$$

Onde:

- a) τ_{bit} é o tempo de um bit em milissegundos;
- b) $hp(m)$ é o grupo de mensagens com prioridade maior do que a mensagem "m";
- c) J_j é o atraso de processamento para cada mensagem j ;
- d) T_j é a taxa de transmissão de cada mensagem j ;
- e) C_j é o tempo de transmissão de cada mensagem j .

Da equação 5, tem-se o termo B_m , que representa o tempo máximo de bloqueio por mensagens de menor prioridade e apresentado como (equação 6):

$$B_m = \max_{\forall k \in lp(m)} (C_k) \quad (6)$$

Da equação 6, o termo $lp(m)$ pode ser definido como o grupo de mensagens de menor prioridade em relação à mensagem "m". Já a segunda parte da equação 5 representa a contribuição de cada mensagem de alta prioridade para o atraso de uma mensagem "m" na fila de transmissão.

Outro trabalho importante relacionado à análise temporal das mensagens CAN é o desenvolvido por (DAVIS et al, 2007). Neste trabalho, realiza-se uma revisão da teoria inicialmente proposta por (TINDELL e BURNS, 1994) com objetivo de resolver problemas relacionados ao não cumprimento de *deadlines* de mensagens. Neste trabalho, considera-se o

comportamento não preemptivo da rede CAN quanto à transmissão das mensagens e apresenta-se um novo conceito denominado de “período de não disponibilidade” (*busy period*), o qual, segundo o autor, pode ser maior que o período da mensagem T_m . Em outras palavras pode-se afirmar que este período representa o tempo em que a linha está ocupada com a transmissão de mensagens de maior prioridade do que dada mensagem “ m ”. Este período pode ser expresso pela equação 7, na qual “ t_m ” é o “tamanho do período de não disponibilidade” e “ hep ” é o conjunto de mensagens de maior prioridade em relação à dada mensagem “ m ”.

$$t_m^{n+1} = B_m + \sum_{\forall k \in hep(m)} \left\lceil \frac{t_m^n + J_k}{T_k} \right\rceil C_k \quad (7)$$

Ainda segundo (DAVIS et al, 2007), existe a possibilidade de um conjunto de instâncias Q_m ficarem disponíveis para transmissão durante o “período de não disponibilidade”. O conjunto de instâncias Q_m é dado por (equação 8):

$$Q_m = \left\lceil \frac{t_m + J_m}{T_m} \right\rceil \quad (8)$$

Para determinar o tempo de resposta no pior caso de uma mensagem “ m ”, torna-se necessário calcular o tempo de resposta de cada uma das instâncias Q_m . O valor máximo encontrado dará o tempo de resposta no pior caso. Na análise realizada o autor utiliza o índice “ q ” para representar uma instância da mensagem “ m ”. A primeira instância no período não disponível corresponde a $q = 0$, e a última instância corresponde a $q = Q_m - 1$. O maior tempo desde o começo do período não disponível até o começo da transmissão da instância “ q ” é dada pela equação 9:

$$w_m^{n+1}(q) = B_m + qC_m + \sum_{\forall k \in hep(m)} \left\lceil \frac{w_m^n + J_k + \tau_{bit}}{T_k} \right\rceil C_k \quad (9)$$

Por fim, o autor demonstra que o pior caso no tempo de resposta de uma mensagem “ m ” é dado por pela equação 10, cujo parâmetro $R_m(q)$ pode ser verificado na equação 11.

$$R_m = \max_{q=0..Q_m-1} (R_m(q)) \quad (10)$$

$$R_m(q) = J_m + w_m(q) + C_m \quad (11)$$

4.2 ANÁLISE DA TAXA DE OCUPAÇÃO DO BARRAMENTO CAN

A taxa de ocupação máxima de uma rede CAN clássica é um dos aspectos que possui distintas abordagens segundo informações verificadas nos estudos da área. Por exemplo, no trabalho elaborado por (ANDERSSON e TOVAR, 2009), os autores afirmam que a taxa de ocupação máxima de uma rede CAN 2.0A (identificador de 11 bits) deve ser de no máximo 25% a fim de que os *deadlines* de todas as mensagens sejam atendidos. Para o CAN 2.0B (identificador de 29 bits) este valor sobe para 29% da taxa de ocupação. Estes valores máximos de taxa de ocupação aplicam-se a redes CAN que utilizam unicamente estratégias de comunicação do tipo *time-triggered*.

Os resultados apresentados por (ANDERSSON e TOVAR, 2009) baseiam-se nos estudos anteriormente realizados por (TINDELL, BURNS e WELLINGS, 1995). A partir destes, os autores retomam os conceitos de intervalo de transmissão entre mensagens (aqui denominado de T_i), tempo de total de transmissão de uma mensagem (denominado de C_i) e tempo de resposta de uma mensagem (denominado de R_i) e, a partir destes, apresentam equacionamento para determinação da taxa de ocupação da rede dado por (equação 12):

$$U = \sum_{i=1}^n \frac{C_i}{T_i} \leq \frac{C_{MIN}}{C_{MIN} + C_{MAX}} \quad (12)$$

Da equação 12, tem-se os termos C_{MIN} e C_{MAX} , representados pelas equações 13 e 14, respectivamente:

$$C_{MIN} = (g + 13) \cdot QUANTUM \quad (13)$$

$$C_{MAX} = \left(g + 8 \cdot S_i + 13 + \left\lceil \frac{g + 8 \cdot S_i - 1}{4} \right\rceil \right) \cdot QUANTUM \quad (14)$$

Das equações 13 e 14, $QUANTUM$ representa o tempo de duração de um único bit e S_i representa o número de bytes no campo de dados da mensagem. Considerando então uma

mensagem CAN 2.0B (29 bits no identificador) e 8 bytes no campo de dados, os autores indicam os valores (equações 15 e 16):

$$C_{MIN} = 67 \cdot QUANTUM \quad (15)$$

$$C_{MAX} = 160 \cdot QUANTUM \quad (16)$$

Desta forma, a taxa de ocupação máxima para uma rede CAN 2.0B pode ser representada por (equação 17):

$$U = \sum_{i=1}^n \frac{C_i}{T_i} \leq \frac{67}{227} \leq 0.2951 \quad (17)$$

A taxa de ocupação de 29.5% para uma rede CAN 2.0B, apresentada como resultado na equação 17, é equivalente aos valores tradicionalmente utilizados como regra geral para taxa de ocupação máxima de uma rede CAN. Por exemplo, segundo (DAVIS et al, 2007) valores entre 30% e 40% de taxa de ocupação eram tidos como regra geral para definição de um sistema CAN até o fim dos anos 90 a de modo a garantir os *deadlines* das mensagens. Posteriormente, com técnicas mais refinadas no escalonamento de mensagens, taxas de até 80% acabaram sendo utilizadas. Porém, mais recentemente, autores como (NAVET e PERRAULT, 2012), voltam a afirmar que valores entre 35% e 40% são mais indicados para elaboração de redes de comunicação CAN sem a necessidade de elaboração de um projeto de gerenciamento eficiente da rede e garantia de cumprimento nas latências e tempos de transmissão das mensagens.

No que se refere à estimação da taxa de ocupação de uma determinada rede CAN, em (JOHANSSON e TORIN, 2002) é apresentada metodologia destinada ao cálculo eficiente da taxa de ocupação de uma rede CAN baseada no protocolo SAE J1939. Nesta metodologia, considera-se inicialmente a ocupação representada por uma única mensagem m , com período p_m , quantidade de bits s_m e comprimento do bit τ (equação 18):

$$U_m = \frac{s_m \tau}{p_m} \quad (18)$$

Considerando-se então um conjunto de mensagens (equação 19):

$$U = \tau \left(\sum_{m=1}^N \frac{s_m}{p_m} \right) \quad (19)$$

Tendo em conta as incertezas nos intervalos de transmissão e eventuais distúrbios na rede CAN, chega-se à expressão final dada por (equação 20):

$$U = \tau \left(\sum_{m=1}^N \frac{s_m}{p_m(1-\varepsilon_m)} \right) + x \frac{2s_D+20}{P_D} \quad (20)$$

Da equação 20 apresentam-se os seguintes termos:

- a) U é a ocupação máxima da rede para N mensagens;
- b) τ é a largura do bit (tempo);
- c) s_m é o número de bits da mensagem m ;
- d) p_m é o período da mensagem m ;
- e) ε_m é a fração de desvio máximo do período de tempo da mensagem m ;
- f) s_D é o maior número de bits da mensagem m ;
- g) P_D é o período de tempo onde o maior número de distúrbios x são esperados;

Por fim, os autores consideram uma situação onde todas as mensagens fossem iguais em comprimento (s) e não houvessem desvios no período das mensagens, de forma que a equação 21 pudesse ser utilizada em formato simplificado:

$$U = s\tau \left(\sum_{m=1}^N \frac{1}{p_m} \right) \quad (21)$$

4.3 TRABALHOS SIMILARES

Por se tratar de um tema novo, o CAN FD ainda possui poucos trabalhos publicados pela comunidade acadêmica. Dentre os trabalhos relacionados com esta dissertação, pode-se inicialmente mencionar o trabalho desenvolvido por (PÉREZ et al, 2003). Utilizando plataforma de desenvolvimento baseada em um processador Intel 386, os autores deste trabalho avaliam o desempenho de um sistema de controle baseado na transmissão e recepção de

mensagens CAN do tipo *event-triggered* e *time-triggered*. Como resultado, o artigo demonstra a influência que os atrasos e latências das mensagens CAN podem acarretar na resposta de um sistema de controle de tempo real. Como conclusão os autores sugerem a implementação de uma topologia de rede do tipo *time-triggered*, como os utilizados em TTCAN ou FTTCAN, por exemplo, nas quais os tempos de atrasos da rede tornam-se variáveis conhecidas para implementação do sistema de controle.

Outro trabalho relevante é o desenvolvido por (NGUYEN, CHEON e JEON, 2014), no qual os autores realizam procedimento de análise comparativa entre o CAN FD e o CAN clássico durante processo de reprogramação de uma ECU. Os ensaios foram realizados utilizando-se a ferramenta de simulação CANOe, da empresa Vector GmbH, e consistiram na transmissão de 960 kbytes de dados de programação de uma ECU específica. Os dados foram transmitidos utilizando o CAN clássico a uma taxa de 500 kbps e utilizando o CAN FD a uma taxa de 500 kbps (velocidade para a arbitragem) e 4 Mbps (velocidade no campo de dados) com distintos tamanhos de mensagens (de 8 a 64 bytes no campo dados). Como resultado, os autores constataram uma redução de até 89% no tempo de reprogramação de uma ECU quando efetuado o processo utilizando o CAN FD com 64 bytes no campo de dados. Na prática este resultado representou uma redução de tempo de 29,83 segundos para 3,31 segundos no tempo de reprogramação de 960 kbytes de uma ECU específica.

De maneira similar ao trabalho anterior, no artigo de (JANG e JEON, 2015) é apresentado estudo de simulação e implementação de um sistema de reprogramação de ECUs via CAN FD e via FlexRay, exclusivamente para fins de comparação de desempenho de ambas tecnologias. O programa utilizado para implementação do sistema também foi o CANOe e as plataformas de *hardware* utilizaram *transceivers* e microcontroladores específicos para CAN FD e FlexRay. Para configuração da rede CAN FD, os autores utilizaram taxa de comunicação de 500 kbps (arbitragem da mensagem) e 5 Mbps (campo de dados) e para a configuração do

FlexRay utilizaram uma metodologia que permite utilizar número variável de *slots* do segmento dinâmico da mensagem. Como resultado, os autores apresentaram um desempenho para o CAN FD de até 558 Kbytes/s (melhor cenário) utilizando 64 bytes no campo de dados e de 843 Kbytes/s utilizando 18 *slots* do segmento dinâmico da mensagem.

Por fim, pode-se afirmar que o trabalho de maior relevância é o de (CHEON e JEON, 2013), o qual também utilizou o programa CANOe como base para simulação de um sistema de comunicação CAN. Neste caso, o sistema simulado não é baseado em uma aplicação real, ou seja, foi definido pelos autores com mensagens fictícias, com uma taxa de ocupação de 80% e aproximadamente 40 mensagens transmitidas por 4 distintas ECUs. O sistema inicialmente foi simulado utilizando o CAN clássico (333 kbps) e, posteriormente, utilizando o CAN FD (333 kbps e 3Mbps no campo de dados). Como resultado, o artigo indica uma redução de 30% para uma taxa de ocupação da rede de 80%. Além disto demonstra grande redução nos tempos de atraso das mensagens de baixa e alta prioridade, com consequente cumprimento dos *deadlines* pré-definidos.

5 ANÁLISE DE IMPACTO DE MIGRAÇÃO DE SISTEMAS PARA CAN FD

A introdução do CAN FD em aplicações automotivas é um tema novo para os fabricantes da área e tende a ser um assunto muito discutido nos próximos anos. Esta afirmação pode ser embasada na necessidade atual por maior quantidade de dados e maior velocidade nas aplicações atuais que utilizam o CAN clássico como meio de transmissão de dados. Operações que demandam implementação de mensagens criptografadas ou então gravação e atualização rápida de códigos fonte das ECUs (durante processo produtivo) são exemplos destas demandas (ADAMSON, 2015).

Nas próximas seções serão apresentados alguns temas referentes aos possíveis impactos a serem verificados pelos fabricantes durante processo de migração para o padrão CAN FD.

5.1 CAN FD E EMC

O aumento da velocidade de transmissão das mensagens CAN FD pode apresentar como consequência a elevação dos níveis de emissões eletromagnéticas em um sistema veicular e dessa forma ultrapassar os limites normatizados para EMC (*Electromagnetic Compatibility*). Emissões eletromagnéticas são definidas pelo formato do sinal e pelo número de oscilações do sinal em função do tempo. Assim sendo, pode-se afirmar que o total de emissões aumenta em 6 dB quando o número de oscilações dobra de valor em determinado intervalo de tempo. Esta é uma questão física e independe do *transceiver* utilizado ou do padrão de medição adotado (ELEND, 2014). A Figura 24 demonstra este aumento no espectro de emissões eletromagnéticas para mensagens CAN a 500 kbps (níveis de emissão entre 40 dB μ V e 50 dB μ V) e mensagens a 2 Mbps (níveis de emissão entre 50 dB μ V e 60 dB μ V).

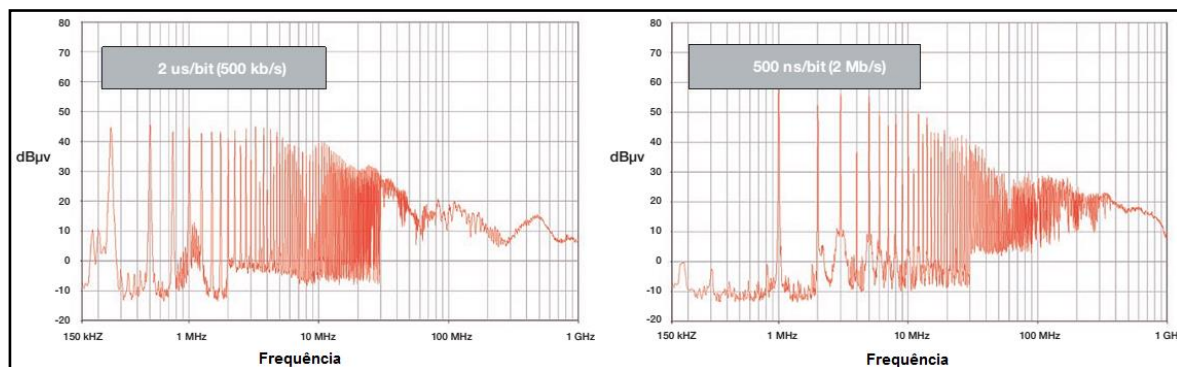


Figura 24 Comparação de EMC em mensagens CAN de 500 kbps e 2 Mbps (LOPEZ; MOUNIER, 2015)

Este aumento da emissão eletromagnética gerado pela elevação da taxa de comunicação tende a ultrapassar os limites estabelecidos pelos órgãos regulamentadores da indústria automotiva. Estudos prévios indicam que velocidades superiores a 5 Mbps já ultrapassaram os níveis de EMC permitidos, a não ser que técnicas de supressão do ruído sejam utilizadas (LOPEZ; MOUNIER, 2015).

5.2 INTEGRAÇÃO DE PADRÕES CAN E CAN FD

Nas seções seguintes serão abordadas algumas soluções previstas para a integração de redes do tipo CAN clássico e CAN FD.

5.2.1 Barramentos Independentes

Considerando-se que a substituição do CAN clássico pelo novo CAN FD acarretará em uma mudança de protocolo, pode-se afirmar então que novos controladores CAN FD serão necessários para gerenciamento da comunicação. Enquanto os novos controladores permitem receber e transmitir tanto mensagens CAN FD quanto mensagens CAN clássicas, os atuais controladores CAN irão reportar erro no recebimento de mensagens CAN FD. Como consequência imediata, verifica-se que será necessário realizar a separação de redes CAN e CAN FD, de forma que na rede CAN FD apenas controladores preparados para este protocolo

sejam utilizados. A curto prazo, é possível que os investimentos para implementação de uma rede CAN FD sejam superiores aos atuais custos de uma rede CAN clássica em função da menor disponibilidade de componentes preparados para este novo padrão (ADAMSON, 2015).

A fim de minimizar maiores impactos, sugere-se como opção mais eficiente a introdução do CAN FD em redes de comunicação que realmente demandem maior largura de banda, ou então, realizar a derivação dos módulos de uma única rede em duas redes distintas separadas por um dispositivo *gateway* (Figura 25).

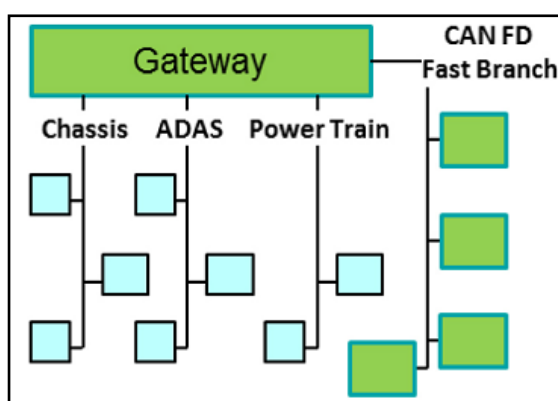


Figura 25 Barramento CAN e CAN FD conectados via *gateway* (ADAMSON, 2015)

5.2.2 Revisão dos controladores CAN clássico.

Para as novas gerações de controladores CAN clássico está prevista uma alteração na próxima revisão da ISO11898-2 que permitirá a integração dos dois padrões. Nesta alteração será utilizada nova nomenclatura para o bit EDL, o qual passará a ser denominado bit FDF. Este bit deverá ser utilizado para os novos controladores CAN ignorarem mensagens do tipo CAN FD. Como o bit FDF é transmitido antes do campo de dados da mensagem CAN FD, caberá aos novos controladores CAN avaliar este se este bit está em estado recessivo e paralisar a decodificação da mensagem até que a linha de comunicação fique em estado de repouso. Este tipo de procedimento evitará que os controladores do CAN clássico interpretem as mensagens CAN FD como erro de comunicação e acabem por inibir a utilização da rede (ELEND, 2014).

5.2.3 Blindagem contra CAN FD

Considerando-se o fato de que atualmente muitos dos controladores CAN são embarcados nos mesmos circuitos integrados de microcontroladores e processadores, talvez a adaptação destes componentes à nova determinação da ISO11898-2 não seja imediata. Para resolver este problema, alguns fabricantes de componentes eletrônicos optaram por oferecer uma solução integrada ao próprio *transceiver* CAN, a qual denominaram de *FD Shield*. Esta solução consiste em um *transceiver* CAN modificado, que possui a habilidade de bloquear mensagens CAN FD presentes na rede e entregar ao controlador CAN apenas mensagens padrão (bit FDF dominante). A Figura 26 demonstra a implementação de uma rede híbrida utilizando a tecnologia mencionada.

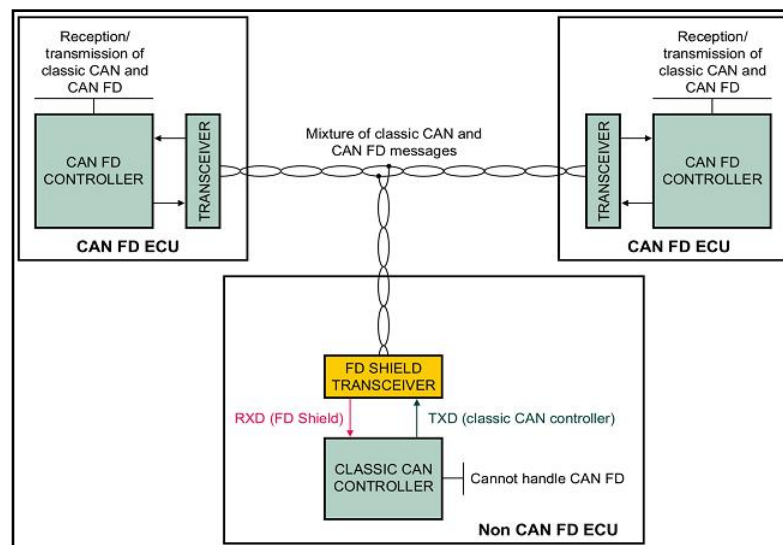


Figura 26 Implementação da solução *FD Shield* (ADAMSON, 2015)

Estes componentes possuem integrados um controlador CAN FD próprio, responsável por gerenciar a filtragem das mensagens CAN clássicas. Uma vez que uma mensagem CAN FD chega ao *transceiver*, os bits de SOF e o identificador são passados ao controlador, porém quando verifica-se a presença do bit FDF, o *transceiver* “mascara” a saída do pino RX mantendo o mesmo em nível baixo (dominante). Após 6 bits, o controlador CAN clássico,

embarcado no microcontrolador ou processador, gera um erro de *stuff* no pino de TX, o qual acaba sendo isolado da linha pelo *transceiver* modificado. O controlador CAN clássico aguarda então o pino de RX retornar ao estado recessivo, enquanto que o *transceiver FD shield* continua a monitorar a mensagem CAN FD. Ao final da mensagem CAN FD (durante o bit de *acknowledge*), o *transceiver* libera o pino de RX para reproduzir o estado real da linha. Neste momento, o controlador CAN clássico possui novamente permissão para enviar informação no delimitador de erro (bits recessivos), o que acaba ocorrendo em paralelo com a indicação de fim de mensagem CAN FD (campo EOF). Este processo ocorre simultaneamente, o que permite normalizar a comunicação ao fim da mensagem. A Figura 27 demonstra graficamente como ocorre o processo de filtragem mencionado.

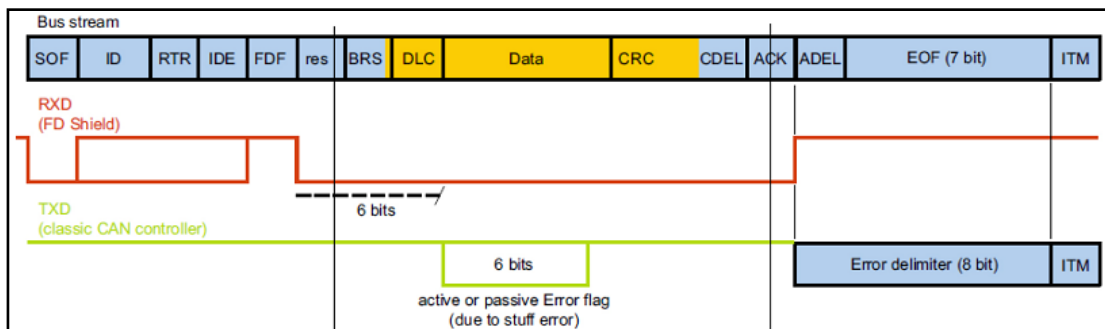


Figura 27 Filtragem de uma mensagem CAN FD pelo FD Shield (ADAMSON, 2015)

5.3 IMPACTOS EM OUTROS PROTOCOLOS AUTOMOTIVOS

A introdução do CAN FD como alternativa de protocolo de comunicação surge como opção para preencher uma lacuna hoje existente entre o CAN clássico e o FlexRay. Tradicionalmente, redes automotivas que utilizam o CAN clássico trabalham utilizando taxas de comunicação entre 250 kbps e 666 kbps com até 8 bytes no campo de dados (SCHREINER et al, 2013) enquanto que sistemas FlexRay podem trabalhar desde 2.5 Mbps até 10 Mbps e 255 bytes no campo dados. O CAN FD por sua vez, oferece transmissão de dados a uma taxa de até

5 Mbps (sem prejuízos relacionados à EMC) com transmissão de até 64 bytes no campo de dados.

Inicialmente a substituição de protocolos de baixa velocidade pelo CAN FD (como o LIN por exemplo) não tende a ser viável pois não existem vantagens econômicas para tal proposta.

Com relação ao CAN clássico, existe grande probabilidade de que redes de comunicação que possuam taxa de ocupação superior a 50% utilizem o padrão CAN FD em função dos benefícios ligados ao aumento da taxa de transmissão e quantidade de dados por mensagem (LOTOCZKY, 2012).

Com relação ao FlexRay pode-se afirmar que o padrão CAN FD apresenta-se como uma opção mais barata, especialmente para sistemas voltados à transmissão de mensagens *event-triggered*, uma vez que FlexRay continua apresentando maiores benefícios para aplicações *time-triggered* (LOTOCZKY, 2012). Sistemas que utilizam FlexRay a uma taxa de 2.5 Mbps poderiam ser substituídos pelo padrão CAN FD caso a velocidade da fase de arbitragem fosse de 1 Mbps e a velocidade no campo de dados fosse superior a 4 Mbps, conforme demonstra a Figura 28 (QUIGLEY, WILLIAMS, MCLAUGHLIN, 2013).

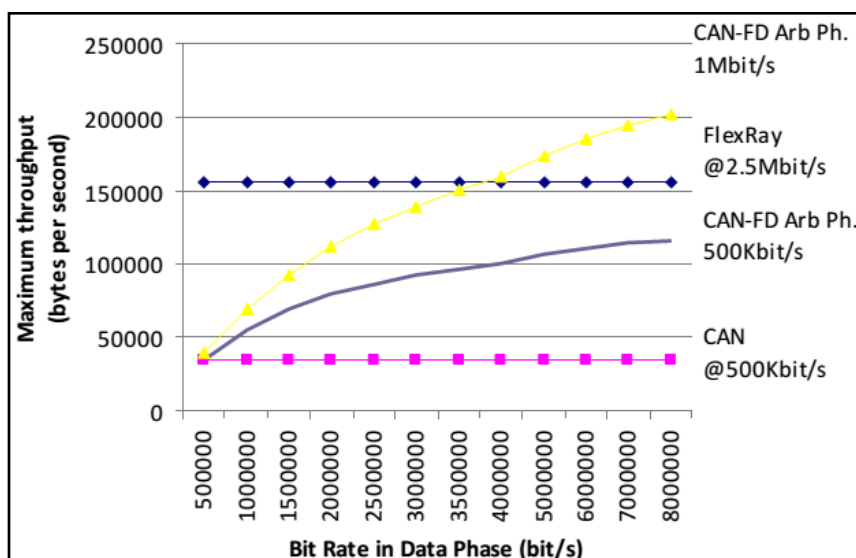


Figura 28 Relação entre quantidade de dados por velocidade no CAN, CAN FD e FlexRay (QUIGLEY, WILLIAMS, MCLAUGHLIN, 2013).

6 ESTUDO DE CASO: SISTEMA ELETRÔNICO DE UM CAMINHÃO FORA DE ESTRADA

Este estudo explora a introdução do protocolo CAN FD em um caminhão basculante fora-de-estrada para aplicações em mineração. Este caminhão possui atualmente duas redes de comunicação CAN, uma delas dedicada ao controle do chassi (motor, transmissão automática e sistema de freios) e a outra dedicada ao controle da carroceria (iluminação interna e externa, controle do ar condicionado, ventilação, controle do basculamento, entre outros). Em ambas as redes CAN (chassi e carroceria) trafegam mensagens formatadas pelo protocolo de aplicação SAE J1939.

Um dos principais objetivos na implementação do CAN FD refere-se à melhora no desempenho obtido nas linhas de comunicação do veículo com a utilização deste protocolo. Para isto serão avaliadas três características que indicam o bom desempenho de uma rede de comunicação: taxa de ocupação da rede, pior tempo de resposta na transmissão de uma mensagem e *jitter* de transmissão.

A taxa de ocupação da rede é um importante indicador para determinação do desempenho da linha de comunicação. Consiste basicamente na quantificação de dados transmitidos nas linhas de comunicação durante determinado intervalo de tempo. Se a taxa de ocupação é demasiadamente alta, é possível que as mensagens aguardem um tempo maior na fila de transmissão até que o barramento CAN fique em repouso (CHEON e JEON, 2013).

O pior caso no tempo de resposta de uma mensagem, ou “*worst-case response time*” (TINDELL, BURNS e WELLINGS, 1995), refere-se ao maior intervalo necessário para envio de uma mensagem com período de transmissão fixo. O *jitter*, por sua vez, refere-se à variação do atraso na entrega de uma mensagem na linha de comunicação. Nas seções seguintes será apresentada descrição mais detalhada dos sistemas de controle e redes de comunicação utilizados no caminhão fora-de-estrada.

6.1 DESCRIÇÃO DO SISTEMA ELETRÔNICO

O caminhão fora-de-estrada possui duas redes de comunicação CAN SAE J1939, sendo uma para o chassi e outra para a carroceria. O barramento CAN do chassi do veículo possui três módulos eletrônicos: o módulo de controle do motor Scania, o módulo de controle da transmissão eletrônica e o módulo de controle do sistema de freios ABS (*Anti-lock Breaking System*). O barramento CAN da carroceria possui conectados um módulo para controle de cargas da cabine (ECU dianteira), um módulo para controle de cargas externas (ECU traseira), o comando da chave-seta, um painel de teclas para acionamento de cargas e uma IHM para operação do caminhão. Ambas as redes são interconectadas através de um módulo denominado *gateway*, o qual tem por função selecionar e retransmitir mensagens específicas de um barramento a outro.

6.1.1 Estruturação das Redes de Comunicação

Segundo orientação da própria norma SAE J1939, as linhas de comunicação CAN instaladas no caminhão utilizam uma topologia em barramento com derivações de tamanho limitado. A Figura 29 representa o sistema eletrônico instalado no veículo.

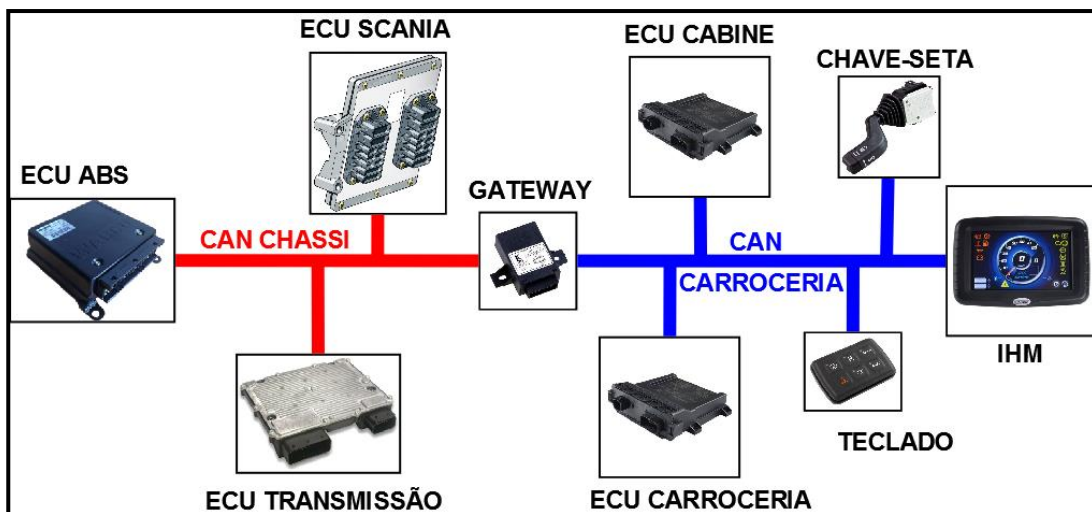


Figura 29 Conexão elétrica dos módulos do caminhão

6.1.2 Descrição dos Módulos de Controle do Chassi

A linha indicada pela cor vermelha na Figura 29 representa a rede de comunicação CAN dos módulos que efetuam o controle do chassi do caminhão. Apesar de possuir apenas três módulos conectados, esta rede possui uma taxa de ocupação elevada, conforme será apresentado nas próximas seções. Estes três equipamentos são fornecidos por fabricantes distintos.

A ECU Scania é o módulo responsável pelo controle direto do motor do veículo, transmitindo mensagens CAN relativas ao torque e temperatura do motor, pressão do óleo, temperatura do líquido de arrefecimento, consumo de combustível, entre outros.

A ECU da transmissão eletrônica envia principalmente informações relativas à caixa de câmbio, sistema de embreagem e sistema diferencial do veículo.

A ECU do ABS transmite informações relativas ao controle eletrônico do freio, informações do sistema hidráulico, velocidade das rodas e informações para controle da estabilidade dinâmica do veículo.

Todos os módulos desta rede de comunicação adotam o padrão SAE J1939, com mensagens e intervalos de transmissão descritos na norma, além de algumas mensagens proprietárias com especificação da mensagem e período de transmissão definidos pelo próprio fabricante.

6.1.3 Descrição dos Módulos de Controle da Carroceria

A linha indicada pela cor azul na Figura 29 representa a rede de comunicação CAN dos módulos que efetuam o controle da carroceria do veículo. Estes módulos transmitem menor quantidade de informações que a rede do chassi (nas próximas seções serão indicadas as mensagens e taxas de transmissão de cada módulo). Todos os equipamentos conectados a esta linha, exceto a chave-seta, são fornecidos pelo mesmo fabricante. As mensagens nela

transmitidas também estão padronizadas de acordo com o protocolo de aplicação SAE J1939, sendo algumas específicas da norma e outras proprietárias do fabricante.

A ECU da cabine controla cargas externas localizadas na parte frontal e no teto do veículo, além de cargas e sistemas da cabine. Entre estes podem-se citar faróis, luzes de posição, sistema de ar condicionado e ventilação, sensores de temperatura interna, posição do pedal do acelerador e freio, luzes indicadoras de direção e limpador de para-brisas. Este módulo realiza a transmissão síncrona de mensagens em intervalo específico a ser apresentado nas próximas seções.

A ECU da carroceria controla cargas localizadas na parte traseira do caminhão, sobretudo luzes externas e sistema de basculamento da caçamba. Este módulo também apresenta a transmissão síncrona de mensagens.

O módulo denominado “teclado” é responsável por transmitir mensagens de acionamento de cargas efetuadas pelo operador. A transmissão das mensagens ocorre apenas por evento, ou seja, somente quando alguma das teclas é pressionada.

O módulo denominado IHM possui como função permitir a visualização de dados provindos do chassi e carroceria, como por exemplo: temperatura motor, nível de combustível, velocidade do veículo, estado de cargas, entre outros. Este dispositivo não transmite mensagens de maneira síncrona.

O dispositivo denominado “chave-seta” indica apenas os acionamentos manuais de luzes e faróis externos, além do motor do limpador de para-brisas.

6.1.4 Descrição do *Gateway*

O dispositivo aqui denominado “*gateway*” trata-se de uma interface de comunicação que separa fisicamente as duas linhas de comunicação CAN (chassi e carroceria) e retransmite mensagens necessárias de uma rede à outra. Um exemplo que pode ser citado são as mensagens

referentes ao motor, as quais trafegam na linha CAN do chassi, mas são retransmitidas pelo *gateway* para a linha da carroceria para visualização de informações na IHM.

6.2 ESPECIFICAÇÃO DAS MENSAGENS CAN

A seguir serão apresentadas as mensagens transmitidas por cada módulo do sistema eletrônico e seus respectivos períodos de transmissão (em milissegundos). As mensagens estão identificadas utilizando-se o valor de PGN, de acordo com a norma SAE J1939. O valor de PGN, conforme indicado na Figura 14 da seção 2.2.5.1, possui 16 bits (8 bits de *PDU Format* e 8 bits de *PDU Specific*) e está contido dentro do próprio identificador da mensagem CAN. Os demais 13 bits do identificador são referentes à prioridade da mensagem, páginas de dados e endereço fonte do módulo transmissor (*PDU Source Address*).

Nas descrições das mensagens a seguir (Tabela 3 a Tabela 8) poderá ser verificado que eventualmente módulos distintos transmitem mensagens com o mesmo valor de PGN. Para situações como esta como esta, o identificador da mensagem em questão possui valor único determinado pelo endereço fonte do módulo transmissor, conforme anteriormente mencionado (campo *Source Address*, Figura 14). Desta forma, o estabelecimento de prioridade em mensagens com o mesmo PGN será dado pelo próprio endereço CAN dos módulos, ou seja, quanto menor o valor do endereço do módulo transmissor, maior a prioridade da mensagem na linha de comunicação.

Na Tabela 3 são apresentadas as mensagens transmitidas pela ECU do motor Scania, com seu respectivo PGN, descrição e período de transmissão.

Tabela 3 Mensagens Módulo Motor Scania

| PGN | Descrição da Mensagem | Período (ms) |
|----------|--|--------------|
| PGN61444 | Controlador Eletrônico do Motor (1) | 20 |
| PGN61443 | Controlador Eletrônico do Motor (2) | 50 |
| PGN65409 | Proprietária (DLN2) | 100 |
| PGN65265 | Controle de Cruzeiro/Velocidade do Veículo | 100 |
| PGN65266 | Economia de Combustível (Líquido) | 100 |

| PGN | Descrição da Mensagem | Período (ms) |
|----------|---|--------------|
| PGN65247 | Controlador Eletrônico do Motor (3) | 250 |
| PGN65263 | Nível/Pressão Fluido do Motor | 500 |
| PGN65270 | Condição de Admissão/Exaustão (1) | 500 |
| PGN65262 | Temperatura do Motor (1) | 1000 |
| PGN65415 | Mensagem Proprietária (1) | 1000 |
| PGN65416 | Mensagem Proprietária (2) | 1000 |
| PGN60670 | Mensagem de anúncio de transmissão mensagens motor | 1000 |
| PGN60671 | Mensagem de anúncio de transmissão mensagens exaustão | 1000 |
| PGN65257 | Consumo de combustível | 1000 |
| PGN65253 | Horímetro | 1000 |
| PGN65225 | Códigos de Erro (1) | 1000 |
| PGN65226 | Códigos de Erro (2) | 1000 |
| PGN65271 | Potencial Elétrico do Veículo | 1000 |
| PGN60415 | Mensagens de Configuração do Motor | 5000 |
| PGN65414 | Configuração do Motor (Proprietário) | 5000 |
| PGN65376 | Mensagens de Uso Geral (1) | 5000 |

Fonte: CAN Specification – EMS/6

Na Tabela 4 é possível verificar as mensagens enviadas pelo módulo da transmissão do motor. Para esta ECU verifica-se maior número de mensagens sendo transmitidas nos intervalos de 10 e 20 milissegundos.

Tabela 4 Mensagens CAN Módulo de Transmissão do Motor.

| PGN | Descrição da Mensagem | Período (ms) |
|----------|---|--------------|
| PGN0 | Controle de Torque/Velocidade | 10 |
| PGN61442 | Controlador Eletrônico da Transmissão (1) | 10 |
| PGN61452 | Controlador Eletrônico da Transmissão (8) | 20 |
| PGN256 | Controle da Transmissão (1) | 50 |
| PGN61440 | Controle Eletrônico do Retarder (1) | 100 |
| PGN61441 | Controlador Eletrônico do Freio (1) | 100 |
| PGN61445 | Controlador Eletrônico da Transmissão (2) | 100 |
| PGN65215 | Proprietária "A" | 100 |
| PGN64932 | Acoplamento da Tomada de Força | 100 |
| PGN65098 | Controlador Eletrônico da Transmissão (7) | 100 |
| PGN64917 | Fluidos da Transmissão (2) | 1000 |
| PGN65226 | Diagnóstico de Falhas DM1 | 1000 |
| PGN65272 | Fluidos da Transmissão (1) | 1000 |
| PGN65275 | Fluidos do Retarder | 1000 |
| PGN65249 | Configuração do Retarder | 5000 |
| PGN65099 | Configuração da Transmissão (2) | 5000 |

Fonte: Alison Transmission

Na Tabela 5 pode-se visualizar as mensagens do módulo do freio ABS do veículo. Este módulo, assim como o módulo do motor, também possui mensagens sendo transmitidas em curtos intervalos de 10, 20 e 50 milissegundos.

Tabela 5 Mensagens CAN Módulo do Freio ABS

| PGN | Descrição da Mensagem | Período (ms) |
|----------|---|--------------|
| PGN0 | Controle de Torque/Velocidade | 10 |
| PGN65449 | Controle de Estabilidade Dinâmica do Veículo (2) | 10 |
| PGN65134 | Velocidade das Rodas em Alta Resolução | 20 |
| PGN41 | Controle de Torque/Velocidade, ABS -> <i>Retarder/Exhaust</i> | 50 |
| PGN15 | Controle de Torque/Velocidade, ABS -> <i>Retarder/Engine</i> | 50 |
| PGN16 | Controle de Torque/Velocidade, ABS -> <i>Retarder/Driveline</i> | 50 |
| PGN61441 | Controlador Eletrônico do Freio (1) | 100 |
| PGN64998 | Sistema do Freio Hidráulico | 100 |
| PGN65215 | Informação de Velocidade da Roda | 100 |
| PGN65265 | Controle de Cruzeiro/Velocidade do Veículo | 100 |
| PGN65103 | Controle de Estabilidade Dinâmica do Veículo (1) | 100 |
| PGN65280 | Indicação de Aspereza da Pista | 100 |
| PGN64964 | Controle Eletrônico do Freio (5) | 100 |
| PGN65268 | Condição do Pneu | 500 |
| PGN65274 | Freios (Pressão) | 1000 |
| PGN57344 | Mensagens Cabine (1) | 1000 |

Fonte: WABCO ABS/ACR

Na Tabela 6, Tabela 7 e Tabela 8 são apresentadas as mensagens transmitidas pelos módulos da cabine, carroceria e chave-seta. Verifica-se para estes módulos uma quantidade menor de mensagens transmitidas, bem como intervalos de transmissão maiores para as mesmas.

Tabela 6 Mensagens CAN Módulo da Cabine

| PGN | Descrição da Mensagem | Período (ms) |
|----------|-------------------------------------|--------------|
| PGN65294 | Posição do Pedal do Acelerador | 50 |
| PGN61440 | Controle Eletrônico do Retarder (1) | 100 |
| PGN65292 | Estado Cargas Cabine | 1000 |
| PGN65293 | Indicação de Falhas (1) | 1000 |
| PGN65269 | Condições Ambientais | 1000 |

Tabela 7 Mensagens CAN Módulo Carroceria

| PGN | Descrição da Mensagem | Período (ms) |
|----------|-----------------------------------|--------------|
| PGN65292 | Estado Cargas Carroceria | 1000 |
| PGN65276 | Informações do Painel de Comandos | 1000 |
| PGN64994 | Pressão do Ar Principal | 1000 |
| PGN53248 | Mensagem de Iluminação da Cabine | 5000 |
| PGN60159 | Requisição de Mensagem | 5000 |

Tabela 8 Mensagens CAN Dispositivo Chave-Seta

| PGN | Descrição da Mensagem | Período (ms) |
|----------|---|--------------|
| PGN64972 | Mensagem de Controle Luzes de Indicação Externa | 1000 |
| PGN65535 | Mensagem Proprietária | 1000 |
| PGN64973 | Controle de Operação do Limpador e Esguicho | 1000 |

6.2.1 Determinação Inicial da Taxa de Ocupação da Rede

Após realizar o levantamento das informações transmitidas por cada módulo do sistema e determinar os períodos de transmissão das mensagens, realizou-se então determinação da taxa de ocupação das duas linhas de comunicação (linha do chassi e a da carroceria).

Além da avaliação para a linha do chassi e da carroceria, também foi considerado um cenário que não utilizasse o *gateway*, ou seja, um arranjo com um único barramento conectando todos os módulos do veículo. Para este cenário também foi realizada avaliação teórica da taxa de ocupação da rede. Esta proposta surgiu como uma opção para os experimentos elaborados durante a realização do trabalho e será melhor explanada nas próximas seções.

A determinação prévia da taxa de ocupação das linhas de comunicação foi efetuada utilizando método distinto dos apresentados na seção 3.2. A equação 25 representa o modo de determinação das taxas de ocupação do sistema.

$$T_o = \frac{Dqtb}{M_i} = \frac{Qtb}{M_i} = \frac{(131+13) * \sum_{i=1} Qtp_i}{5000} \quad (25)$$

Para determinação da equação 25 utilizou-se a seguinte metodologia:

- a) determinação do maior intervalo de transmissão para o conjunto de mensagens do sistema. Neste caso verifica-se 5000 ms como sendo o maior período de transmissão (M_i);
- b) determinação da quantidade de pacotes transmitidos dentro do intervalo de 5000 ms. Esta avaliação é feita para cada uma das mensagens transmitidas, por exemplo, para uma mensagem com período de transmissão de 100 ms, serão realizadas 50 transmissões durante o intervalo de 5000 ms. Esta variável é especificada como Q_{tp} ;
- c) somatório da quantidade total de mensagens transmitidas dentro do intervalo de 5000 ms ($\sum Q_{tpi}$);
- d) quantidade total de *bits* transmitidos no intervalo de 5000 ms. Cada mensagem possui aproximadamente 131 *bits*, e se considerarmos o mesmo valor no campo de dados, será necessário adicionar mais 13 bits de *stuff*. Este dado é representado por Q_{tb} ;
- e) determinação do tempo de duração da quantidade total de *bits* transmitidos no referido intervalo. Este dado é obtido dividindo-se a quantidade total de *bits* pela taxa de transmissão (250 kbps). Este dado é representado por D_{qtb} ;
- f) determinação da taxa de ocupação da rede, determinada pelo tempo de duração dividido pelo maior intervalo de transmissão (T_o).

Utilizando-se então a metodologia acima apresentada, verificou-se os seguintes resultados para os barramentos de comunicação:

- a) taxa de ocupação de 48,21% para a linha de comunicação do chassi;
- b) taxa de ocupação de 27,92% para a linha de comunicação da carroceria;
- c) taxa de ocupação de 50,48% para o barramento unificado.

Cabe ressaltar que apesar da linha de comunicação da carroceria possuir poucas mensagens sendo transmitidas por seus módulos, deve-se considerar também as mensagens retransmitidas pelo *gateway* a partir da linha de comunicação do chassi. Estas mensagens retransmitidas são utilizadas pela IHM do sistema a qual utiliza a informação para visualização de dados do motor pelo operador do veículo.

6.3 DEFINIÇÃO DOS ENSAIOS

Após avaliação teórica da taxa de ocupação da rede CAN do chassi e da carroceria do caminhão, utiliza-se então o programa CANoe para simulação do sistema atual e também do sistema modificado com o padrão CAN FD. Nestas simulações será possível determinar, além da taxa de ocupação da rede, tempos de atraso das mensagens em relação ao intervalo ideal de transmissão das mesmas. Nas próximas seções serão descritas as características de cada experimento realizado utilizando o programa CANoe.

6.3.1 Ensaio 1: Simulação do Sistema Atual

Neste experimento implementou-se no programa CANoe um sistema similar ao atualmente utilizado no caminhão fora-de-estrada, ou seja, duas linhas de comunicação CAN no formato clássico (chassi e carroceria) interconectadas por um módulo *gateway* (Figura 30). A linha CAN do chassi possui os módulos do motor, transmissão e ABS e o módulo da carroceria possui os módulos da cabine, carroceria, chave-seta, além do teclado e IHM, estes dois últimos apenas como receptores na linha. O *gateway*, conforme anteriormente mencionado, retransmite algumas mensagens de uma linha à outra, sobretudo mensagens da linha do chassi à linha da carroceria para visualização de parâmetros na IHM.

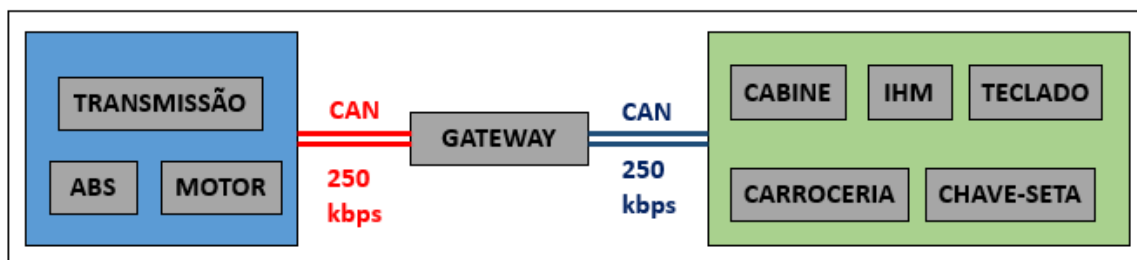


Figura 30 Estruturação da rede para ensaio 1

A simulação utilizou em ambas linhas de comunicação uma taxa de comunicação de 250 kbps e mensagens com identificador de 29 *bits*, conforme padrão estabelecido pela SAE J1939. Todas as mensagens transmitidas possuem 8 bytes no campo de dados.

6.3.2 Ensaio 2: Rede CAN Unificada

Este experimento foi realizado com o intuito de avaliar uma situação hipotética onde o *gateway* não tivesse sido considerado na constituição das linhas de comunicação CAN. O experimento foi realizado unificando-se as duas redes (chassi e carroceria), mantendo-se todos os módulos originais (exceto *gateway*) e mantendo-se taxa de comunicação em 250 kbps e mensagens com identificadores de 29 bits, além dos 8 bytes no campo de dados (Figura 31).

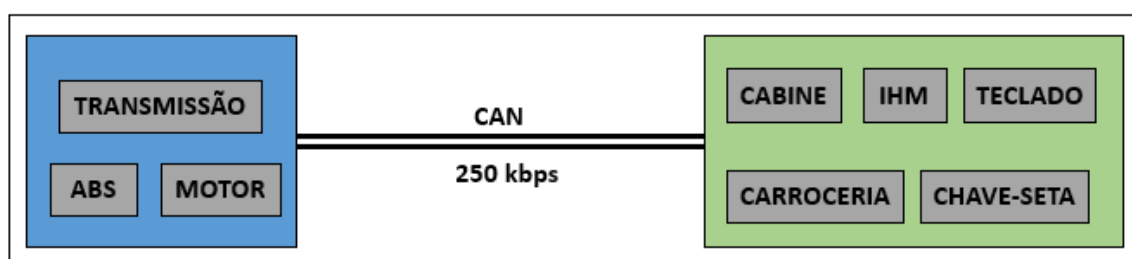


Figura 31 Estruturação da rede para ensaio 2

6.3.3 Ensaio 3: Rede CAN FD Unificada

A realização deste experimento segue o mesmo conceito da simulação com a rede CAN clássica em um único barramento (Figura 32). O objetivo neste caso é avaliar a taxa de ocupação

e os tempos de atraso das mensagens para efetuar comparativo com o experimento anterior e verificar se os resultados obtidos permitem a implementação segura deste padrão no caminhão fora-de-estrada.

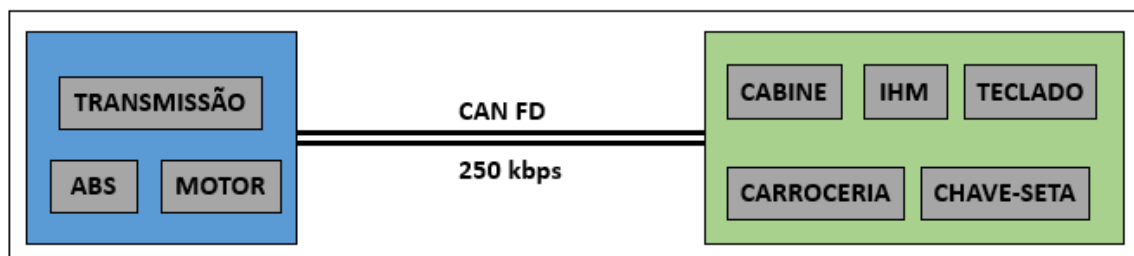


Figura 32 Estruturação da rede para ensaio 3

O experimento baseia-se na conversão das mensagens em formato CAN clássico para CAN FD, mantendo-se a mesma quantidade de dados transmitidos (8 bytes por mensagem), porém aumentando-se a taxa de comunicação de 250 kbps para 2 Mbps. O aumento da taxa de transmissão ocorre, neste caso, apenas no campo de dados, para os demais campos da mensagem a velocidade permanece em 250 kbps.

Com relação à taxa de comunicação do campo de dados, optou-se por uma velocidade de 2 Mbps em virtude de este ser o valor mais usual para os *transceivers* CAN FD atualmente disponíveis comercialmente (Tabela 2). Além disto, conforme indicado na seção 5.1, trata-se de uma velocidade que não representa aumento excessivo nos níveis de emissão eletromagnética.

6.3.4 Ensaio 4: Rede CAN clássica para Carroceria e CAN FD para Chassi

Esta simulação considera novamente a utilização de duas linhas de comunicação interconectadas pelo *gateway*, porém, utilizando padrões distintos (Figura 33). Esta simulação baseia-se na maior taxa de ocupação calculada anteriormente para a rede CAN do chassi, logo propõe-se para esta a alteração para o padrão CAN FD com taxa de comunicação de 2 Mbps (apenas dados). A rede da carroceria se mantém conforme o padrão original e o *gateway* assume

nova função de além de retransmitir mensagens, realizar a conversão dos padrões CAN para CAN FD e vice-versa.

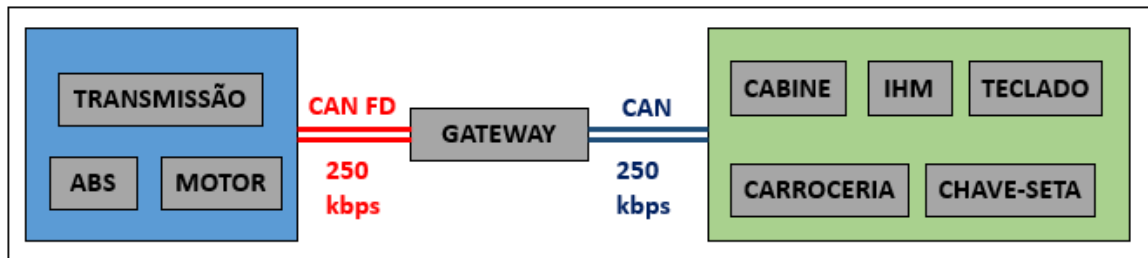


Figura 33 Estruturação da rede para ensaio 4

6.3.5 Ensaio 5: Rede CAN FD Unificada Utilizando Mensagens Agrupadas

O último experimento (Figura 34) retoma a proposta de unificação das linhas de comunicação utilizando o padrão CAN FD, porém desta vez aproveitando-se um dos recursos mais importantes desta nova versão do CAN: enviar maior quantidade de bytes no campo de dados a fim de otimizar a transmissão das mensagens. Para estruturar este procedimento torna-se necessário realizar o agrupamento de várias mensagens em um único pacote de dados. Na seção 7.2.5 este procedimento será melhor explicitado.

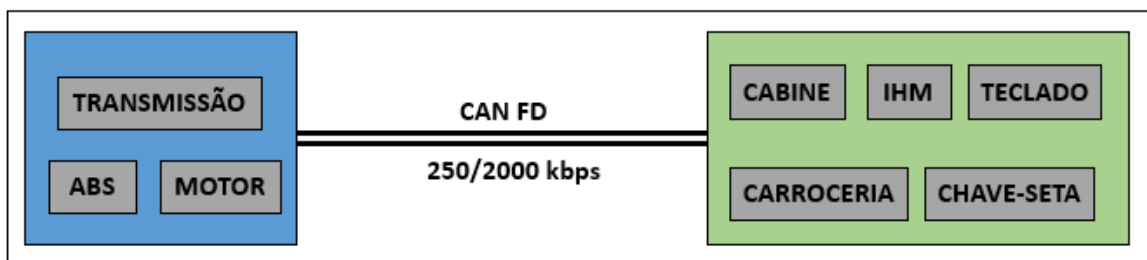


Figura 34 Estruturação da rede para ensaio 5

7 SIMULAÇÕES E RESULTADOS

Para a realização dos experimentos utilizou-se o programa Vector CANoe 8.5. Nas próximas seções serão abordados os procedimentos elaborados para realização das simulações das redes CAN e CAN FD utilizando este programa.

7.1 DESCRIÇÃO DO PROGRAMA DE SIMULAÇÃO

O programa CANoe é uma ferramenta de desenvolvimento, teste e análise para sistemas de comunicação automotivos, desenvolvido e distribuído sob licença proprietária pela empresa Vector Informatik GmbH (VECTOR, 2015).

O CANoe é comumente utilizado por fabricantes de automóveis e fornecedores de sistemas eletrônicos para indústria automotiva, pois proporciona uma série de recursos que permitem simular não só sistemas de comunicação CAN, como também sistemas LIN, *FlexRay*, *Ethernet* e MOST, além de protocolos de aplicação CAN como o SAE J1939, CANopen, ISOBUS, entre outros (VECTOR, 2015).

O desenvolvimento de sistemas distribuídos utilizando o CANoe requer a utilização de uma base de dados que contenha uma descrição das mensagens a serem utilizadas pelo sistema, parâmetros de configuração da rede, cadastro dos módulos eletrônicos (ECUs) e associação entre módulos e mensagens transmitidas. Também requer a estruturação da rede e programação do modo funcional das ECUs.

A criação da base de dados é realizada utilizando-se o *CANdb++ Editor*. Este módulo do CANoe permite, além de realizar o cadastro das mensagens, criar variáveis de ambiente para controlar os diferentes modos de simulação e permitir a integração dos módulos da rede com as interfaces visuais disponíveis.

O CANoe disponibiliza um editor, o CAPL (*Communication Access Programming Language*), o qual permite a elaboração dos arquivos de programação das ECUs utilizando

comandos e expressões similares à linguagem “C”. Cada arquivo de programação está associado a um módulo eletrônico do sistema e executa tarefas dedicadas ao mesmo.

7.2 IMPLEMENTAÇÃO DO SISTEMA

Para cada experimento realizado fez-se necessário configurar o sistema no programa CANoe seguindo os seguintes passos:

- a) estruturação das redes da carroceria e chassi;
- b) configuração das mensagens no banco de dados do CANoe;
- c) configuração funcional dos módulos do sistema;

Nas próximas seções serão abordados com maiores detalhes os procedimentos acima mencionados.

7.2.1 Configuração das Redes CAN e CAN FD

Nesta etapa realizou-se a configuração das redes do chassi e carroceria e efetuou-se a declaração e nomeação dos módulos eletrônicos utilizados. Além disso, é nesta parte do processo que se define o tipo de comunicação utilizado na linha de comunicação (CAN ou CAN FD) e também onde se estipula a velocidade de transmissão das mensagens.

Na Figura 35 e na Figura 36, é possível verificar os módulos eletrônicos interconectados pelo barramento CAN em suas respectivas linhas de comunicação (carroceria e chassi).

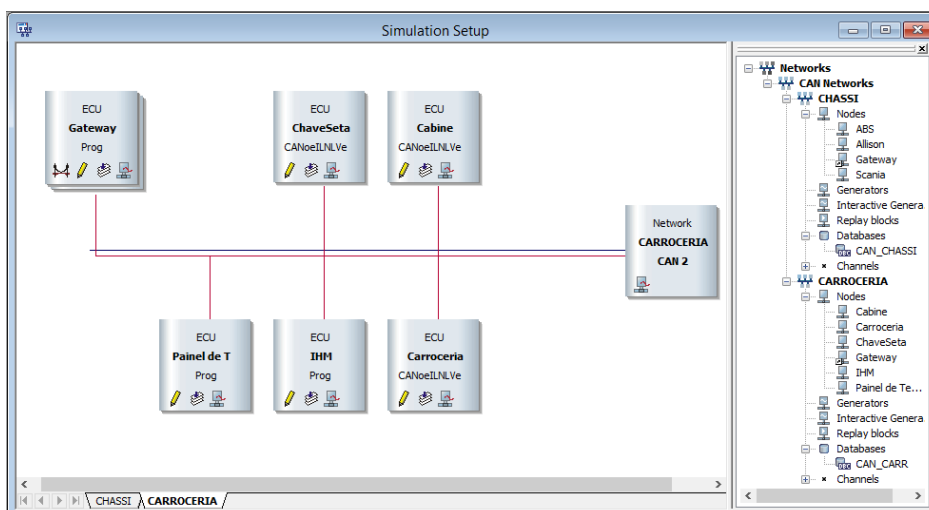


Figura 35 Rede CAN da Carroceria do Caminhão

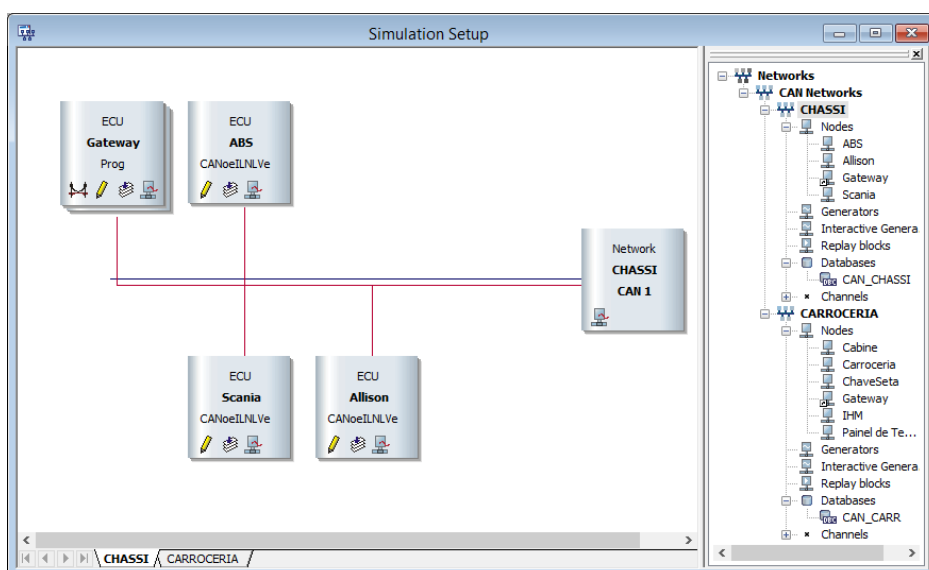


Figura 36 Rede CAN do Chassi do Caminhão

7.2.2 Cadastro das Mensagens

Após efetuar a estruturação das redes de comunicação e seus respectivos módulos, realizou-se então cadastro das mensagens no banco de dados do programa.

O processo de cadastro das mensagens CAN é realizado em um módulo adicional do CANoe, denominado *CANdb++ Editor*. As mensagens cadastradas no *CANdb++* recebem um

nome específico, a quantidade de bits do identificador (11 ou 29 bits), um valor numérico em hexadecimal para o campo do identificador da mensagem e a quantidade de bytes transmitidos no campo de dados. Para os primeiros ensaios as duas redes CAN (chassi e carroceria) seguem o padrão de mensagens do protocolo SAE J1939, ou seja, o identificador de todas as mensagens será do tipo “estendido” (29 bits) e a quantidade de dados transmitidos será sempre de 8 bytes.

Por fim, utiliza-se ainda o *CANdb++* para vincular as mensagens cadastradas aos módulos que efetuarão a transmissão das mesmas. Cada mensagem cadastrada possui um identificador único, o que significa que ela somente será transmitida por um único módulo.

7.2.3 Transmissão das Mensagens

Estruturas de comunicação baseadas no padrão SAE J1939 possuem como característica inerente a transmissão de mensagens de forma síncrona, com intervalos de transmissão especificados pela própria norma. Estes intervalos podem ser especificados para valores desde 10, 20, 50, 100 milissegundos até valores como 1 ou 5 segundos e variam de acordo com o tipo de mensagem a ser transmitida. Torna-se importante ressaltar também que uma mensagem CAN possui um tempo de transmissão aproximado de 500 microssegundos (para uma taxa de transmissão de 250 kbps e 8 bytes de dados), logo, quanto maior o intervalo de transmissão, menor a taxa de ocupação desta mensagem no barramento de comunicação.

Além da forma síncrona de transmissão, a norma SAE J1939 pode especificar também o envio instantâneo de mensagens do sistema antes que o intervalo especificado seja atingido. Trata-se, neste caso, de uma mescla de transmissão síncrona com transmissão assíncrona, ou seja, no momento em que o evento relacionado se mantém em um estado definido, a mensagem volta a ser transmitida em seu intervalo regular. Este tipo de transmissão é geralmente aplicado às mensagens que contenham informações importantes para o desempenho do sistema ou que

estão relacionadas à segurança veicular. Esta estratégia também permite aos sistemas de controle atuar em tempo real quando necessário.

Como primeira consideração para a simulação do sistema do caminhão fora-de-estrada, definem-se os intervalos de transmissão com base na especificação da SAE J1939. As tabelas 3 a 8, mencionadas na seção 6.2, demonstram os módulos e as mensagens com seus respectivos períodos de transmissão síncrona. Além da simulação e avaliação da transmissão síncrona, também será implementada uma análise do desempenho na transmissão de duas destas mensagens de forma assíncrona (transmissão por evento), com gerenciamento da transmissão conforme descrito no parágrafo anterior. A geração do evento para a transmissão da mensagem será condicionada ao pressionar de uma tecla do computador. Este procedimento permitirá a medição do tempo de atraso entre a geração do evento e o recebimento da mensagem (*jitter* da mensagem).

De posse dos intervalos de transmissão das mensagens, tornou-se necessário transportar esta informação para o programa CANoe para que o sistema a ser simulado represente de forma adequada as redes de comunicação do caminhão. Durante este processo, verificou-se que o banco de dados CANdb++ não possui recurso que permita a configuração do período de transmissão das mensagens na rede de comunicação CAN. Faz-se necessário então configurar a sincronização de mensagens de cada dispositivo CAN utilizando um segundo módulo do CANoe, o CAPL. O mesmo ocorreu para a geração de eventos para transmissão de mensagens assíncronas.

O CAPL é um ambiente de programação baseado em linguagem C, o qual foi projetado com o intuito de prover flexibilidade e customização na estruturação de sistemas de comunicação (VECTOR, 2004). Para utilização e correta configuração de transmissão do CAPL, fez-se necessário criar um arquivo de configuração para cada um dos módulos da rede. Neste arquivo deve haver uma referência para o banco de dados anteriormente criado no

CANdb++ e também devem ser criadas rotinas de sincronização do intervalo de tempo para as diferentes mensagens. A rotina de configuração consiste basicamente em iniciar uma variável com o tempo necessário e recarregar o temporizador a cada fechamento do ciclo. Neste momento também são transmitidas as mensagens que possuem o intervalo de tempo correspondente. Abaixo segue exemplo de configuração e transmissão de mensagens utilizando a estrutura de programação do CAPL. No apêndice é apresentado o código fonte completo de todos os módulos do sistema.

```

variables                // Variável de estouro de tempo de 50 ms
{
msTimer Timer_50ms;
}
on start                  // Iniciação do temporizador
{
setTimer(Timer_50ms,50);
}
on timer Timer_50ms      // Estouro de Temporizador
{
setTimer(Timer_50ms,50); // Recarrega para novo estouro
output(PGN65294_);      // Transmite mensagem CAN
}

```

7.2.4 Medição da Taxa de Ocupação da Rede e Intervalos de Transmissão

A elevada taxa de ocupação existente na linha de comunicação do chassi, anteriormente calculada e apresentada na seção 6.2.1, deve ser um dos parâmetros a serem avaliados nas simulações realizadas no CANoe. Para avaliação deste dado, utiliza-se recurso disponibilizado pela ferramenta de simulação e denominado de *CAN Statistics*. Esta ferramenta apresenta o valor média de ocupação da rede em um intervalo de 1 segundo. Nas próximas seções serão apresentados os dados de ocupação da rede relativos às simulações realizadas.

Além da taxa de ocupação, verifica-se também a necessidade de realizar a medição dos tempos de atraso das mensagens. Este dado torna-se importante de ser avaliado, pois está diretamente relacionado à taxa de ocupação da rede, além de possuir relevância na atuação de

sistemas de controle de tempo real. Para a avaliação do intervalo de transmissão das mensagens, verifica-se a necessidade de criar uma rotina de análise em um dos módulos da rede uma vez que este tipo de informação não é disponibilizada pelo simulador. Assim sendo, determina-se que o módulo responsável pela medição dos intervalos de transmissão das mensagens será o próprio *gateway*, já configurado na rede para retransmitir mensagens da linha de comunicação do chassi para a carroceria e vice-versa. Basicamente este dispositivo monitora o intervalo de recepção entre uma mensagem “*n*” e a mensagem “*n-1*”. Outro aspecto importante é que este intervalo é determinado individualmente para cada mensagem do barramento. Abaixo demonstra-se trecho do código fonte para determinar o tempo de atraso da mensagem.

```
Previus_GetTime = Now_GetTime;  
Now_GetTime = timeNow();  
MsgInterval = Now_GetTime-Previus_GetTime;
```

Utilizando-se então a lógica acima mencionada, define-se pela especificação do maior período de transmissão de uma mensagem específica dentre 100 valores adquiridos para esta mesma mensagem. A aquisição destes valores ocorre durante o funcionamento normal do sistema, ou seja, com todas as demais mensagens trafegando na linha de comunicação.

Para a medição do tempo de *jitter* das mensagens assíncronas utilizou-se procedimento similar, ou seja, no momento em que a tecla do computador era pressionada gerando o evento de disparo na transmissão, carregava-se o tempo de relógio para uma variável global. Posteriormente, na recepção da mensagem pelo módulo de monitoramento (*gateway*), carregava-se novamente o tempo de relógio e obtinha-se a diferença entre a geração do evento e recepção da mensagem. A lógica aqui descrita também pode ser verificada no apêndice deste trabalho.

7.2.5 Agrupamento de Mensagens para CAN FD

Os procedimentos de simulação executados no CANoe apresentam como característica comum a transmissão de 8 bytes no campo de dados, seja ela utilizando o CAN clássico ou o

CAN FD, conforme recomendação da norma SAE J1939. Porém, uma das características de maior relevância para o CAN FD é o fato de tornar possível a transmissão de até 64 bytes no campo de dados, aumentando assim a eficiência no envio de informação para a rede CAN. Então, para realizar proveito desta característica, torna-se necessário elaborar adaptações no sistema de forma que se transmita a maior quantidade possível de dados em uma única mensagem e de forma que os intervalos de transmissão mínimos de cada mensagem sejam atendidos.

Desta forma, define-se que uma das simulações utilizará mensagens agrupadas em um único pacote de dados, com mensagens que utilizam períodos de transmissão de mesmo valor. Por exemplo, um agrupamento de mensagens que são transmitidas a cada 10 milissegundos, outro agrupamento de mensagens transmitidas a cada 50 milissegundos, etc. Outra característica deste agrupamento é que as mensagens reunidas no mesmo pacote sejam transmitidas sempre pelo mesmo módulo.

Para realizar o agrupamento determina-se um identificador comum de 29 bits e, posteriormente, agrupa-se as mensagens ocupando 8 bytes consecutivos, sem estabelecer um delimitador para separar e identificar as mesmas (ver Figura 37). Este tipo de alocação de mensagens em um único pacote de dados pressupõe que todos os módulos do sistema tenham pré-conhecimento desta formatação. Também fica evidente que este tipo de metodologia não está em conformidade com o protocolo de aplicação SAE J1939, uma vez que o mesmo ainda não contempla o padrão CAN FD em suas especificações.

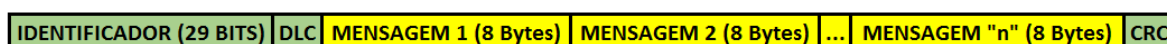


Figura 37 Formatação de Mensagens CANFD com mais de 8 bytes

Nas tabelas a seguir são demonstrados os agrupamentos de mensagens para o ensaio com o barramento CAN FD.

A Tabela 9 demonstra cinco grupos de mensagens relativas ao módulo do motor e que puderam ser alocadas em um único pacote, sendo elas mensagens com intervalos de 100, 500, 1000 e 5000 milissegundos. Para as mensagens com intervalo de 1000 milissegundos, verifica-se a necessidade de realizar dois agrupamentos independentes em função do número total de dados exceder 64 bytes (quantidade máxima suportada por uma mensagem CAN FD).

Tabela 9 Agrupamento de Mensagens do Motor

| PGN | Descrição da Mensagem | Período (ms) | Dados |
|----------|---|--------------|-------|
| PGN61444 | Controlador Eletrônico do Motor (1) | 20 | 8 |
| PGN61443 | Controlador Eletrônico do Motor (2) | 50 | 8 |
| PGN65265 | Proprietária (DLN2) Controle de Cruzeiro/Velocidade do Veículo Economia de Combustível (Líquido) | 100 | 24 |
| PGN65247 | Controlador Eletrônico do Motor (3) | 250 | 8 |
| PGN65263 | Nível/Pressão Fluído do Motor Condição de Admissão/Exaustão (1) | 500 | 16 |
| PGN65262 | Temperatura do Motor (1) Mensagem de anúncio de transmissão mensagens motor Mensagem de anúncio de transmissão mensagens exaustão Consumo de combustível Horímetro Potencial Elétrico do Veículo | 1000 | 48 |
| PGN65415 | Mensagem Proprietária (1) Mensagem Proprietária (2) Códigos de Erro (1) Códigos de Erro (2) | 1000 | 32 |
| PGN60415 | Mensagens de Configuração do Motor Configuração do Motor (Proprietário) Mensagens de Uso Geral (1) | 5000 | 24 |

A Tabela 10 demonstra as mensagens agrupadas para o módulo da transmissão do caminhão. Neste arranjo verifica-se que houve agrupamento de duas mensagens de 10 milissegundos de intervalo, estratégia esta que pode trazer benefícios significativos considerando que estas são algumas das mensagens que mais utilizam banda na linha de comunicação.

Tabela 10 Agrupamento de Mensagens Transmissão

| PGN | Descrição da Mensagem | Período (ms) | Dados |
|--------------------|---|--------------|-------|
| PGN61442 | Controle de Torque/Velocidade | 10 | 16 |
| | Controlador Eletrônico da Transmissão (1) | 10 | |
| PGN61452 PGN256 | Controlador Eletrônico da Transmissão (8) | 20 | 8 |
| | Controle da Transmissão (1) | 50 | 8 |
| PGN65215 | Controle Eletrônico do Retarder (1) | 100 | 48 |
| | Controlador Eletrônico do Freio (1) | | |
| | Controlador Eletrônico da Transmissão (2) | | |
| | Proprietária "A" Acoplamento da Tomada de Força Controlador Eletrônico da Transmissão (7) | | |
| PGN64917 | Fluídos da Transmissão (2) | 1000 | 32 |
| | Diagnóstico de Falhas DM1 | | |
| | Fluídos da Transmissão (1) | | |
| | Fluídos do Retarder | | |
| PGN65249 | Configuração do Retarder | 5000 | 16 |
| | Configuração da Transmissão (2) | | |

Na Tabela 11 verificam-se as mensagens agrupadas para o módulo do freio ABS. Neste arranjo também foi possível unir mensagens de período de transmissão de 10 milissegundos, além de um grande grupo de mensagens relativas ao período de 100 milissegundos.

Tabela 11 Agrupamento de Mensagens Freio ABS

| PGN | Descrição da Mensagem | Período (ms) | Dados |
|----------|---|--------------|-------|
| PGN0 | Controle de Torque/Velocidade | 10 | 16 |
| | Controle de Estabilidade Dinâmica do Veículo (2) | | |
| PGN65134 | Velocidade das Rodas em Alta Resolução | 20 | 8 |
| PGN41 | Controle de Torque/Velocidade, ABS -> <i>Retarder/Exhaust</i> | 50 | 24 |
| | Controle de Torque/Velocidade, ABS -> <i>Retarder/Engine</i> | | |
| | Controle de Torque/Velocidade, ABS -> <i>Retarder/Driveline</i> | | |
| PGN65215 | Controlador Eletrônico do Freio (1) | 100 | 64 |
| | Sistema do Freio Hidráulico | 100 | |
| | Informação de Velocidade da Roda | 100 | |
| | Controle de Cruzeiro/Velocidade do Veículo | 100 | |
| | Controle de Estabilidade Dinâmica do Veículo (1) | 100 | |
| | Indicação de Aspereza da Pista Controlador Eletrônico do Freio (5) | 100 | |
| PGN65268 | Condição do Pneu | 500 | 8 |
| PGN57344 | Freios (Pressão) | 1000 | 16 |
| | Mensagens Cabine (1) | | |

Na Tabela 12, Tabela 13 e Tabela 14 verificam-se os agrupamentos realizados para as mensagens dos módulos da carroceria. Nestes módulos a quantidade de mensagens agrupadas é proporcionalmente menor do que nos módulos do chassi do veículo.

Tabela 12 Agrupamento de Mensagens Módulo da Cabine

| PGN | Descrição da Mensagem | Período (ms) | Dados |
|----------|---|--------------|-------|
| PGN65294 | Posição do Pedal do Acelerador | 50 | 8 |
| PGN61440 | Controle Eletrônico do Retarder (1) | 100 | 8 |
| PGN65292 | Estado Cargas Cabine Indicação de Falhas (1) Condições Ambientais | 1000 | 24 |

Tabela 13 Agrupamento de Mensagens Módulo da Carroceria

| PGN | Descrição da Mensagem | Período (ms) | Dados |
|----------|--|--------------|-------|
| PGN65276 | Estado Cargas Carroceria Informações do Painel de Comandos Pressão do Ar Principal | 1000 | 24 |
| PGN53248 | Mensagem de Iluminação da Cabine | 5000 | 8 |
| PGN60159 | Requisição de Mensagem | 5000 | 8 |

Tabela 14 Agrupamento de Mensagens Módulo da Chave Seta

| PGN | Descrição da Mensagem | Período (ms) | Dados |
|----------|---|--------------|-------|
| PGN65535 | Mensagem de Controle Luzes de Indicação Externa Mensagem Proprietária Controle de Operação do Limpador e Esguicho | 1000 | 24 |

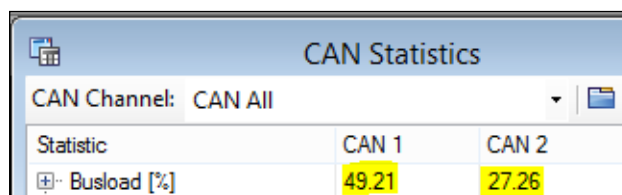
7.3 ENSAIOS E RESULTADOS OBTIDOS

Nas próximas seções serão apresentados com maiores detalhes os experimentos realizados, bem como os resultados obtidos para cada um dos ensaios anteriormente mencionados na seção 6.3. Dentre os resultados a serem demonstrados, destacam-se as taxas de ocupação das redes de comunicação e os períodos de transmissão das mensagens.

7.3.1 Ensaio 1

Após a correta configuração do sistema no CANoe (conforme indicado na Figura 30), configuração das mensagens via *CANdb++* e implementação das rotinas de transmissão de mensagens via CAPL, inicia-se então o processo de análise da taxa de ocupação das redes e intervalos de transmissão das mensagens.

Para a avaliação da taxa de ocupação, torna-se possível utilizar o módulo do CANoe denominado *CAN Statistics* o qual indica uma taxa de ocupação de 49,21% para a rede CAN do chassi (CAN 1) e 27,26% para a rede da carroceria (CAN 2), conforme indicado pela Figura 38. Cabe aqui salientar que, apesar da baixa quantidade de dados transmitidos pelos módulos da carroceria, verifica-se uma taxa de ocupação superior a 27%, valor este que resulta principalmente em função das mensagens retransmitidas pelo *gateway* a partir da linha do chassi.



| Statistic | CAN 1 | CAN 2 |
|-------------|-------|-------|
| Busload [%] | 49.21 | 27.26 |

Figura 38 Taxa de ocupação ensaio 1

Com relação aos valores de taxa de ocupação obtidos na simulação (Figura 38), pode-se verificar uma pequena diferença em relação aos valores anteriormente calculados (ver seção 6.2.1):

- a) cálculo da rede do chassi: 48.21% (1% menor do que o valor simulado);
- b) cálculo da rede da carroceria: 27.92% (0,66% maior do que o valor simulado);

Na seção 7.4 serão apresentadas possíveis causas para as diferenças encontradas entre os valores de taxa de ocupação calculados e os valores simulados.

Após verificação da taxa de ocupação da rede, realizou-se avaliação dos intervalos de transmissão das mensagens. As avaliações foram realizadas de forma individual para cada uma das 66 mensagens do sistema. Considerando-se o número elevado de mensagens transmitidas na rede, demonstram-se aqui (Tabela 15) os resultados relativos aos pacotes que apresentaram maiores valores de tempo de atraso em relação ao intervalo regular de transmissão síncrona.

Tabela 15 Intervalos de Transmissão Ensaio 1

| Módulo | PGN | Período (ms) | CAN (ms) | Atraso (%) |
|-------------|----------|--------------|----------|------------|
| ABS | PGN0 | 10 | 11,08 | 10,8 |
| | PGN65449 | 10 | 16,47 | 64,7 |
| | PGN65215 | 100 | 116,45 | 16,45 |
| | PGN65265 | 100 | 117,62 | 17,62 |
| Motor | PGN61444 | 20 | 22,94 | 14,7 |
| | PGN65265 | 100 | 117,04 | 17,04 |
| | PGN65247 | 250 | 269,94 | 7,976 |
| Transmissão | PGN61442 | 10 | 21,74 | 117,4 |
| | PGN65215 | 100 | 115,86 | 15,86 |
| | PGN65098 | 100 | 114,69 | 14,69 |
| Cabine | PGN65294 | 50 | 52,36 | 4,72 |
| | PGN65292 | 1000 | 1005,1 | 0,51 |
| Carroceria | PGN65276 | 1000 | 1008,81 | 0,881 |
| Chave Seta | PGN65535 | 1000 | 1009,4 | 0,94 |

Segundo dados apresentados na Tabela 15, verificam-se atrasos na transmissão das mensagens com valores superiores a 10%, especialmente para mensagens transmitidas na linha de comunicação do chassi. Entre estes, verifica-se mensagem com atraso superior a 100% (PGN61442, da transmissão do veículo), dado este que representa uma potencial perda de eficiência no processo de controle relativo a esta informação. Estes resultados devem-se principalmente à elevada taxa de ocupação da rede (49.21%, conforme Figura 38) e pela quantidade elevada de mensagens com intervalos de transmissão entre 10 e 100 milissegundos. Para uma taxa de comunicação de 250 kbps, uma mensagem CAN apresenta aproximadamente

0.5 milissegundos de comprimento, o que representa 5% do intervalo de transmissão para uma mensagem com um intervalo de transmissão de 10 milissegundos.

7.3.2 Ensaio 2

Da mesma forma que no ensaio 1, realiza-se a implementação da rede proposta (Figura 31) utilizando-se o programa CANoe. A Figura 39 demonstra a rede estabelecida para simulação das mensagens CAN.

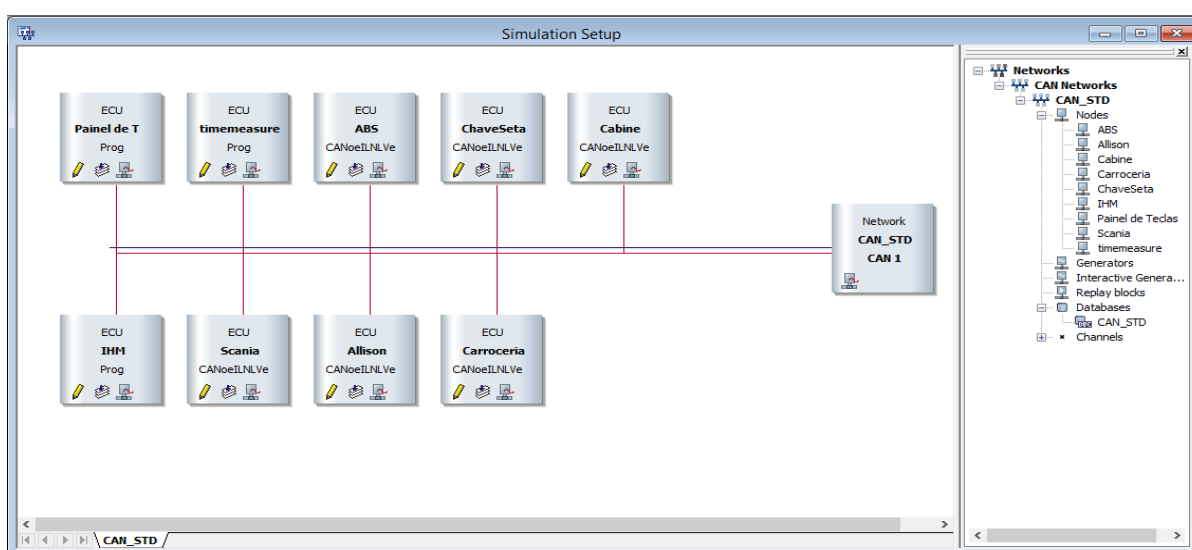
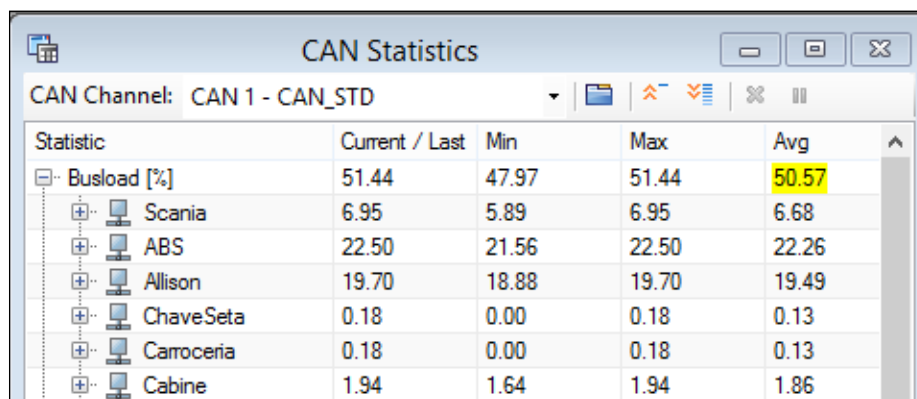


Figura 39: Rede CAN no CANoe para ensaio 2

Além de realizar a nova configuração da rede conectando todos os módulos do sistema, torna-se necessário realizar alteração no banco de dados CANdb++ a fim de unificar as mensagens em um único sistema de dados. Também se torna necessário inutilizar a função de retransmissão de mensagens por parte do *gateway* e mantê-lo na linha apenas como dispositivo de análise dos intervalos de transmissão das mensagens.

Após configuração do sistema para o novo conceito de rede, realiza-se então análise da taxa de ocupação do barramento CAN unificado e constata-se uma taxa de ocupação média de 50,57%, conforme Figura 40. Este valor, quando comparado à taxa de ocupação da linha CAN do chassi no experimento 1, representa aumento de apenas 1,36% devido ao acréscimo das

mensagens providas dos módulos da carroceria. Este dado demonstra a baixa interferência das mensagens providas dos módulos da carroceria, situação esta que pode ser justificada pelo baixo volume de mensagens destes módulos. Outro aspecto a ser analisado em relação ao valor obtido na simulação (Figura 40), refere-se à comparação com o valor calculado (ver seção 6.2.1) de uma taxa de ocupação de 50,48%, ou seja, o valor simulado foi 0,09% superior ao calculado.



| CAN Statistics | | | | |
|------------------------------|----------------|-------|-------|-------|
| CAN Channel: CAN 1 - CAN_STD | | | | |
| Statistic | Current / Last | Min | Max | Avg |
| Busload [%] | 51.44 | 47.97 | 51.44 | 50.57 |
| Scania | 6.95 | 5.89 | 6.95 | 6.68 |
| ABS | 22.50 | 21.56 | 22.50 | 22.26 |
| Allison | 19.70 | 18.88 | 19.70 | 19.49 |
| ChaveSeta | 0.18 | 0.00 | 0.18 | 0.13 |
| Carroceria | 0.18 | 0.00 | 0.18 | 0.13 |
| Cabine | 1.94 | 1.64 | 1.94 | 1.86 |

Figura 40 Taxa de ocupação ensaio 2

Para este ensaio, verificam-se então os seguintes resultados para os atrasos relativos aos intervalos de transmissão das mensagens CAN (Tabela 16):

Tabela 16 Intervalos de Transmissão Ensaio 2

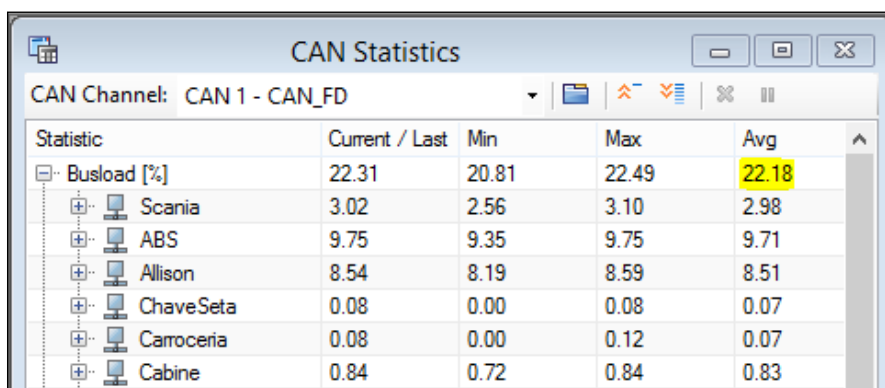
| Módulo | PGN | Período (ms) | CAN (ms) | Atraso (%) |
|-------------|----------|--------------|----------|------------|
| ABS | PGN0 | 10 | 11,18 | 11,8 |
| | PGN65449 | 10 | 18,24 | 82,4 |
| | PGN65215 | 100 | 117,62 | 17,62 |
| | PGN65265 | 100 | 119,79 | 19,79 |
| Motor | PGN61444 | 20 | 22,94 | 14,7 |
| | PGN65265 | 100 | 119,79 | 19,79 |
| | PGN65247 | 250 | 277,98 | 11,192 |
| Transmissão | PGN61442 | 10 | 24,09 | 140,9 |
| | PGN65215 | 100 | 117,04 | 17,04 |
| | PGN65098 | 100 | 115,86 | 15,86 |
| Cabine | PGN65294 | 50 | 54,72 | 9,44 |
| | PGN65292 | 1000 | 1007,06 | 0,706 |
| Carroceria | PGN65276 | 1000 | 1045,24 | 4,524 |
| Chave Seta | PGN65535 | 1000 | 1047,62 | 4,762 |

De acordo com os dados apresentados na Tabela 16, os atrasos nos intervalos de transmissão também apresentaram valores superiores a 10%, sobretudo nas mensagens relativas aos módulos do chassi. Os valores obtidos foram, em sua totalidade, superiores aos demonstrados no experimento 1, o que pode ser explicado pela maior taxa de ocupação no barramento de comunicação (50.57%, conforme Figura 40).

7.3.3 Ensaio 3

Da mesma forma que o ensaio 2, este ensaio consiste em um barramento de comunicação unificado, sem a utilização do *gateway* conectando os módulos do chassi e da carroceria, porém utilizando o padrão CAN FD, com velocidade de 250 kbps para arbitragem da mensagem e 2000 kbps para o pacote de dados da mensagem (Figura 32). Nesta simulação foram utilizados os mesmos identificadores das mensagens adotadas nos experimentos anteriores. A configuração dos códigos de transmissão das mensagens para cada módulo pôde ser mantida conforme experimento 2 (códigos do CAPL), porém as mensagens existentes no banco de dados tiveram de ser recadastradas como mensagens CAN FD com 8 bytes no pacote de dados.

O primeiro procedimento realizado foi a verificação da taxa de ocupação da rede, a qual pode ser visualizada na Figura 41. O valor a ser observado indica uma taxa de ocupação de 22.18%, o que representa uma redução de aproximadamente 56% em relação ao ensaio de número 2.



The screenshot shows a window titled "CAN Statistics" for "CAN Channel: CAN 1 - CAN_FD". It displays a table with columns for "Statistic", "Current / Last", "Min", "Max", and "Avg". The "Busload [%]" row is highlighted in yellow, showing a current value of 22.31 and an average of 22.18. Below it, several modules are listed with their respective busload statistics.

| Statistic | Current / Last | Min | Max | Avg |
|-------------|----------------|-------|-------|-------|
| Busload [%] | 22.31 | 20.81 | 22.49 | 22.18 |
| Scania | 3.02 | 2.56 | 3.10 | 2.98 |
| ABS | 9.75 | 9.35 | 9.75 | 9.71 |
| Allison | 8.54 | 8.19 | 8.59 | 8.51 |
| ChaveSeta | 0.08 | 0.00 | 0.08 | 0.07 |
| Carroceria | 0.08 | 0.00 | 0.12 | 0.07 |
| Cabine | 0.84 | 0.72 | 0.84 | 0.83 |

Figura 41 Taxa de Ocupação ensaio 3

Da mesma maneira, realiza-se também avaliação dos tempos de atrasos para as mesmas mensagens até agora demonstradas. O resultado pode ser observado na Tabela 17.

Tabela 17 Intervalos de transmissão ensaio 3

| Módulo | PGN | Período (ms) | CAN FD (ms) | Atraso (%) |
|-------------|----------|--------------|-------------|------------|
| ABS | PGN0 | 10 | 10,52 | 5,2 |
| | PGN65449 | 10 | 13,57 | 35,7 |
| | PGN65215 | 100 | 106,59 | 6,59 |
| | PGN65265 | 100 | 107,09 | 7,09 |
| Motor | PGN61444 | 20 | 21,26 | 6,3 |
| | PGN65265 | 100 | 106,85 | 6,85 |
| | PGN65247 | 250 | 259,33 | 3,732 |
| Transmissão | PGN61442 | 10 | 15,33 | 53,3 |
| | PGN65215 | 100 | 106,34 | 6,34 |
| | PGN65098 | 100 | 105,84 | 5,84 |
| Cabine | PGN65294 | 50 | 52,04 | 4,08 |
| | PGN65292 | 1000 | 1003,05 | 0,305 |
| Carroceria | PGN65276 | 1000 | 1014,96 | 1,496 |
| Chave Seta | PGN65535 | 1000 | 1016 | 1,6 |

Segundo dados obtidos da Tabela 17, verifica-se elevada redução nos tempos de atraso das mensagens avaliadas quando comparados aos dados do ensaio número 2 e, além disso, verifica-se que grande parte das mensagens, possui atrasos inferiores a 10% em relação ao valor nominal de transmissão. Na seção 7.3.6 a seguir será realizada comparação mais detalhada dos resultados dos ensaios.

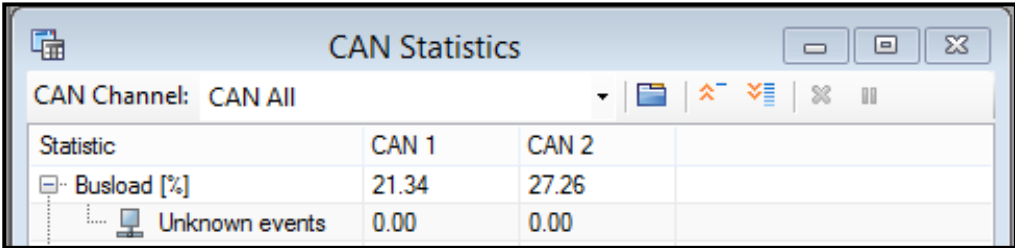
7.3.4 Ensaio 4

Este ensaio, conforme verificado anteriormente, volta a utilizar o dispositivo gateway separando uma rede CAN FD de uma rede CAN (ver Figura 33). Definiu-se por utilizar estes distintos sistemas em função dos resultados obtidos nos ensaios anteriores, os quais demonstraram que a linha CAN do chassi apresenta maior taxa de ocupação do que a linha

CAN da carroceria, logo supõe-se que as informações do chassi do veículo estariam melhor adequadas utilizando-se o CAN FD.

Para executar este ensaio, utilizou-se os recursos disponibilizados pelos experimentos anteriores. Basicamente aproveitou-se os mesmos dados disponíveis para o ensaio 1, porém realizando-se nova configuração da linha do chassi conforme características do ensaio 3, ou seja, uma rede CAN FD de 250 kbps na arbitragem da mensagem, 2000 kbps e 8 bytes para o campo de dados. A configuração estrutural desta rede é similar às apresentadas na Figura 35 e na Figura 36.

Na Figura 42 é possível visualizar os resultados obtidos na simulação do CANoe para o ensaio 4. A linha do chassi, a qual utilizou-se o padrão CAN FD e a qual é referenciada como CAN1 na Figura 42, obteve uma taxa de ocupação, de 21.34 %. Na configuração do ensaio 1 o valor apresentado foi de 49.21%, o que indica uma redução de 27.87%. A rede da carroceria, a qual utiliza o CAN clássico e que é referenciada como CAN2 manteve o valor 27.26%, conforme resultado já avaliado no ensaio de número.



| Statistic | CAN 1 | CAN 2 |
|----------------|-------|-------|
| Busload [%] | 21.34 | 27.26 |
| Unknown events | 0.00 | 0.00 |

Figura 42 Taxa de ocupação das redes CAN FD e CAN

Na Tabela 18 pode-se visualizar os intervalos de atraso nos dois barramentos de comunicação. Verifica-se nesta tabela valores de atraso inferiores aos obtidos no ensaio 3 para a rede CAN FD. Esta pequena redução deve-se ao fato da taxa de ocupação também ter diminuído em aproximadamente 1% do ensaio 3 para o ensaio 4. Os valores relativos à rede CAN clássica apresentaram resultados similares aos obtidos no ensaio 1.

Tabela 18 Intervalos de atraso para ensaio 4

| Módulo | Identificador | Período (ms) | CAN (ms) | CAN FD (ms) | Atraso (%) |
|-------------|---------------|--------------|----------|-------------|------------|
| ABS | PGN0 | 10 | - | 10,25 | 2,5 |
| | PGN65449 | 10 | - | 12,54 | 25,4 |
| | PGN65215 | 100 | - | 105,1 | 5,1 |
| | PGN65265 | 100 | - | 104,23 | 4,23 |
| Motor | PGN61444 | 20 | - | 21,26 | 6,3 |
| | PGN65265 | 100 | - | 105,3 | 5,3 |
| | PGN65247 | 250 | - | 255,81 | 2,324 |
| Transmissão | PGN61442 | 10 | - | 14,31 | 43,1 |
| | PGN65215 | 100 | - | 105,83 | 5,83 |
| | PGN65098 | 100 | - | 105,33 | 5,33 |
| Cabine | PGN65294 | 50 | 52,3 | - | 4,6 |
| | PGN65292 | 1000 | 1005,1 | - | 0,51 |
| Carroceria | PGN65276 | 1000 | 1008,8 | - | 0,88 |
| Chave Seta | PGN65535 | 1000 | 1009,5 | - | 0,95 |

7.3.5 Ensaio 5

No ensaio de número 5 realizou-se novamente a unificação das linhas de comunicação (sem utilização do *gateway*) e utilizou-se o padrão CAN FD (250 kbps e 2000 kbps) como protocolo de comunicação (Figura 34). Além disto, realizou-se a transmissão de mensagens de forma agrupada, conforme mencionado na seção 7.2.5.

O ensaio realizado utiliza então as mensagens apresentadas nas Tabelas 9 a 14 e na Figura 43 pode-se visualizar a taxa de ocupação deste barramento, com um valor médio de 13.85%. De posse deste resultado pode-se afirmar que este ensaio apresenta a menor taxa de ocupação dentro todos os experimentos realizados, mesmo considerando a utilização de barramento único com todos os módulos conectados.

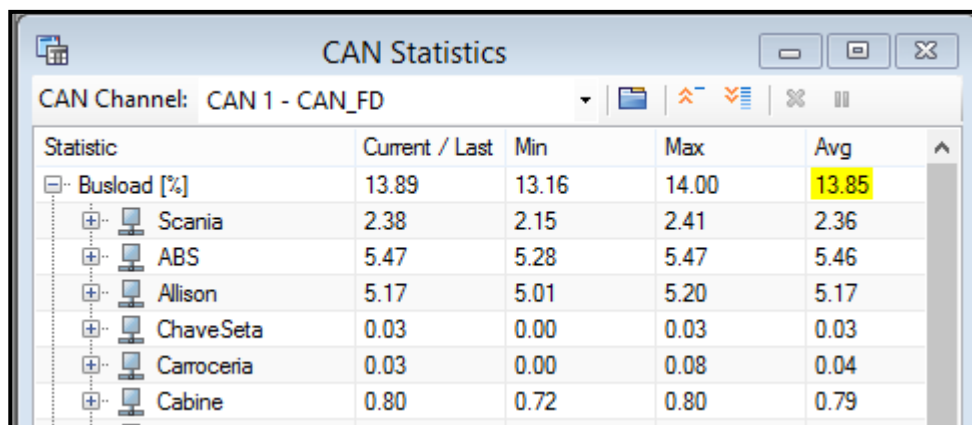


Figura 43 Taxa de ocupação ensaio 5 CAN FD

Para avaliação dos intervalos de transmissão, utilizaram-se mensagens com PGNs comuns aos ensaios anteriores. Na Tabela 19 pode-se visualizar os resultados dos intervalos de atraso das mensagens. Nesta tabela verifica-se que todas as mensagens apresentaram atrasos inferiores a 10%, exceto a mensagem PGN61442, com intervalo síncrono de 10 milissegundos. Apesar de ter apresentado tempo de atraso de 12,74 milissegundos, verifica-se que este valor também se encontra abaixo dos demais ensaios realizados, demonstrando assim que este experimento apresenta a melhor eficiência na transmissão das mensagens para o sistema proposto.

Tabela 19 Intervalo de atraso ensaio 5

| Módulo | Identificador | Período (ms) | Dados | CAN FD (ms) | Atraso (%) |
|-------------|---------------|--------------|-------|-------------|------------|
| ABS | PGN0 | 10 | 16 | 10,26 | 2,6 |
| | PGN65215 | 100 | 64 | 101,44 | 1,44 |
| Motor | PGN61444 | 20 | 8 | 20,84 | 4,2 |
| | PGN65265 | 100 | 24 | 103,1 | 3,1 |
| | PGN65247 | 250 | 8 | 253,79 | 1,516 |
| Transmissão | PGN61442 | 10 | 16 | 12,74 | 27,4 |
| | PGN65215 | 100 | 48 | 103,14 | 3,14 |
| Cabine | PGN65294 | 50 | 8 | 51,39 | 2,78 |
| | PGN65292 | 1000 | 24 | 1002,23 | 0,223 |
| Carroceria | PGN65276 | 1000 | 24 | 1006,58 | 0,658 |
| Chave Seta | PGN65535 | 1000 | 24 | 1007,29 | 0,729 |

7.3.6 Resumo dos resultados

Na Tabela 20 são novamente apresentados os resultados dos ensaios 1 a 5 de forma a permitir comparação mais eficiente das taxas de ocupação obtidas. Pela análise da mesma fica clara a maior taxa de ocupação da rede para os ensaios que utilizam o CAN clássico como padrão de comunicação.

Tabela 20 Taxa de ocupação da rede para diferentes ensaios

| Ensaio | Padrão (Chassi / Carroceria) | Taxa de Ocupação (%) |
|----------|---------------------------------|-------------------------|
| Ensaio 1 | CAN / CAN | 49,21 / 27,26 |
| Ensaio 2 | CAN | 50,57 |
| Ensaio 3 | CAN FD (8 bytes) | 22,18 |
| Ensaio 4 | CAN FD (8 bytes) / CAN | 21,34 / 27,26 |
| Ensaio 5 | CAN FD (> 8 bytes) | 13,85 |

Já na Tabela 21 pode-se verificar os diferentes intervalos de atraso de algumas mensagens nos ensaios realizados.

Tabela 21 Intervalos de atraso mensagens CAN ensaios 1 a 5

| ECU | PGN | Período (ms) | Ensaio 1 (ms) | Ensaio 2 (ms) | Ensaio 3 (ms) | Ensaio 4 (ms) | Ensaio 5 (ms) |
|-------------|----------|-----------------|------------------|------------------|------------------|------------------|------------------|
| ABS | PGN0 | 10 | 11,08 | 11,18 | 10,52 | 10,25 | 10,26 |
| | PGN65449 | 10 | 16,47 | 18,24 | 13,57 | 12,54 | - |
| | PGN65215 | 100 | 116,45 | 117,62 | 106,59 | 105,1 | 101,44 |
| | PGN65265 | 100 | 117,62 | 119,79 | 107,09 | 104,23 | - |
| Motor | PGN61444 | 20 | 22,94 | 22,94 | 21,26 | 21,26 | 20,84 |
| | PGN65265 | 100 | 117,04 | 119,79 | 106,85 | 105,3 | 103,1 |
| | PGN65247 | 250 | 269,94 | 277,98 | 259,33 | 255,81 | 253,79 |
| Transmissão | PGN61442 | 10 | 21,74 | 24,09 | 15,33 | 14,31 | 12,74 |
| | PGN65215 | 100 | 115,86 | 117,04 | 106,34 | 105,83 | 103,14 |
| | PGN65098 | 100 | 114,69 | 115,86 | 105,84 | 105,33 | - |
| Cabine | PGN65294 | 50 | 52,36 | 54,72 | 52,04 | 52,3 | 51,39 |
| | PGN65292 | 1000 | 1005,1 | 1007,06 | 1003,05 | 1005,1 | 1002,23 |
| Carroceria | PGN65276 | 1000 | 1008,81 | 1045,24 | 1014,96 | 1008,8 | 1006,58 |
| Chave Seta | PGN65535 | 1000 | 1009,4 | 1047,62 | 1016 | 1009,5 | 1007,29 |

Na Figura 44 são representados os mesmos dados da Tabela 21, porém agora em forma de gráfico, com representação dos atrasos em valores percentuais relativos ao intervalo original de transmissão da mensagem.

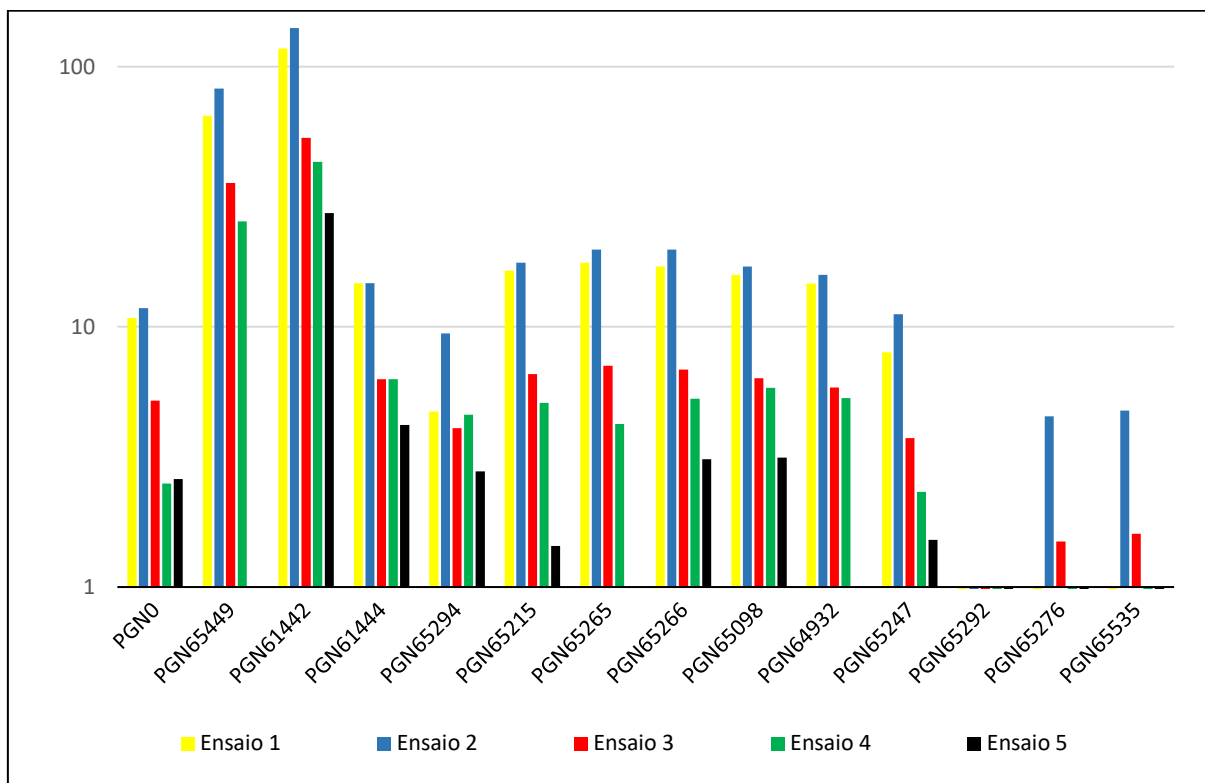


Figura 44 Variação percentual intervalos de transmissão das mensagens CAN/CAN FD

Considerando-se que os ensaios realizados utilizaram sempre o pior caso no período de transmissão, demonstra-se na Figura 45 a distribuição dos tempos de transmissão para todos os ensaios realizados (Ensaio 1 a 5) referentes ao PGN61442. Estas distribuições são referentes a uma quantidade de 100 transmissões consecutivas. A Tabela 22 seguinte demonstra os valores médios e desvios padrão dos tempos de transmissão para os valores apresentados na Figura 45.

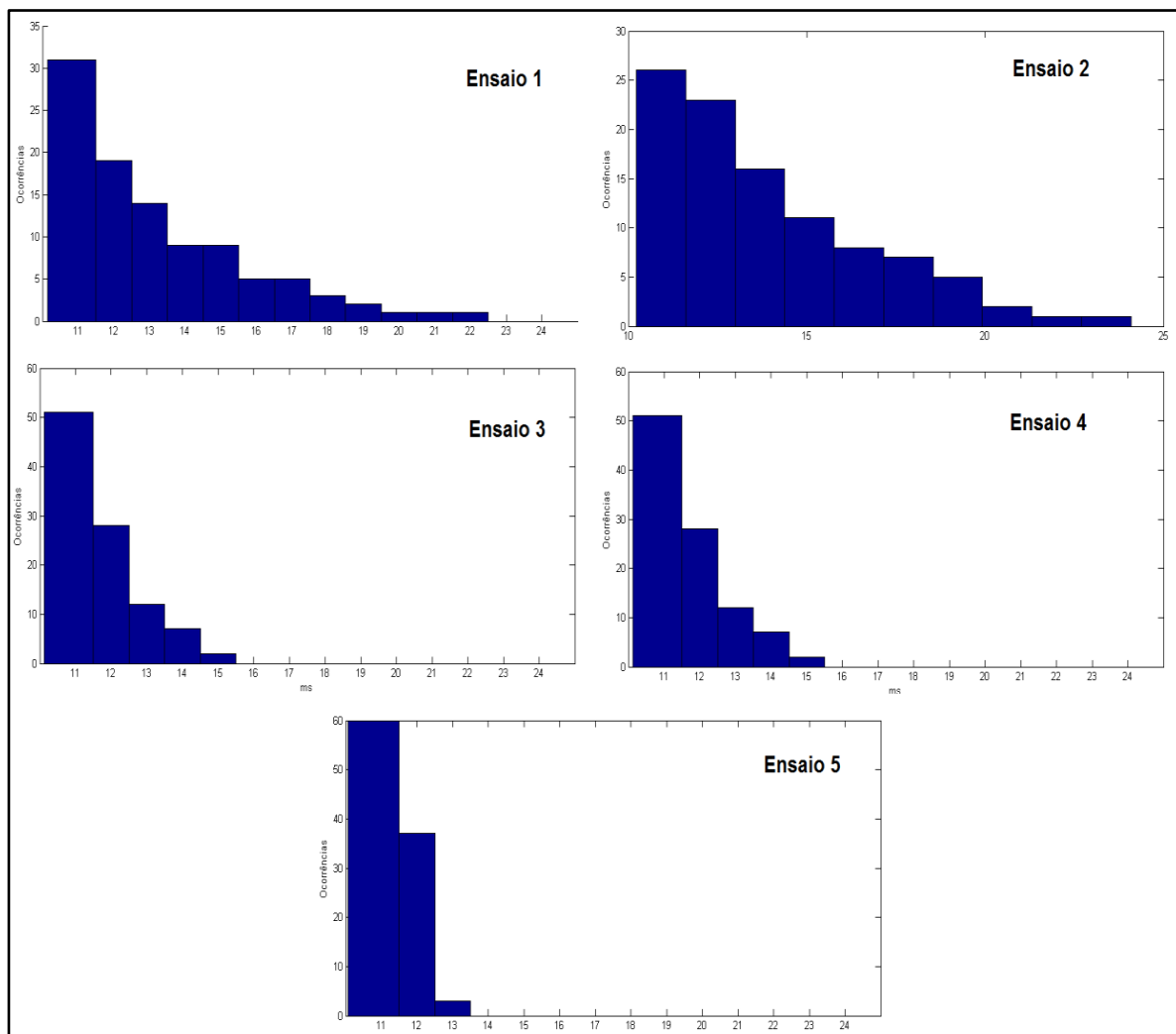


Figura 45 Distribuição períodos de transmissão PGN61442 (Ensaio 1 a 5)

Tabela 22 Distribuição períodos PGN61442 (Ensaio 1 a 5)

| Ensaio | Período Médio (ms) | Maior Período (ms) | Desvio Padrão (ms) |
|----------|--------------------|--------------------|--------------------|
| Ensaio 1 | 13,09 | 21,74 | 2,67 |
| Ensaio 2 | 13,77 | 24,09 | 3,06 |
| Ensaio 3 | 11,61 | 15,33 | 1,21 |
| Ensaio 4 | 11,53 | 14,76 | 1,10 |
| Ensaio 5 | 11,18 | 12,74 | 0,72 |

Da mesma forma que apresentado anteriormente, verifica-se na Figura 46 a distribuição dos períodos de transmissão para o PGN65265, o qual possui intervalo de transmissão definido para 100 milissegundos. Na

Tabela 23 pode ser verificado os valores médios e desvios padrão para os dados apresentados na Figura 46.

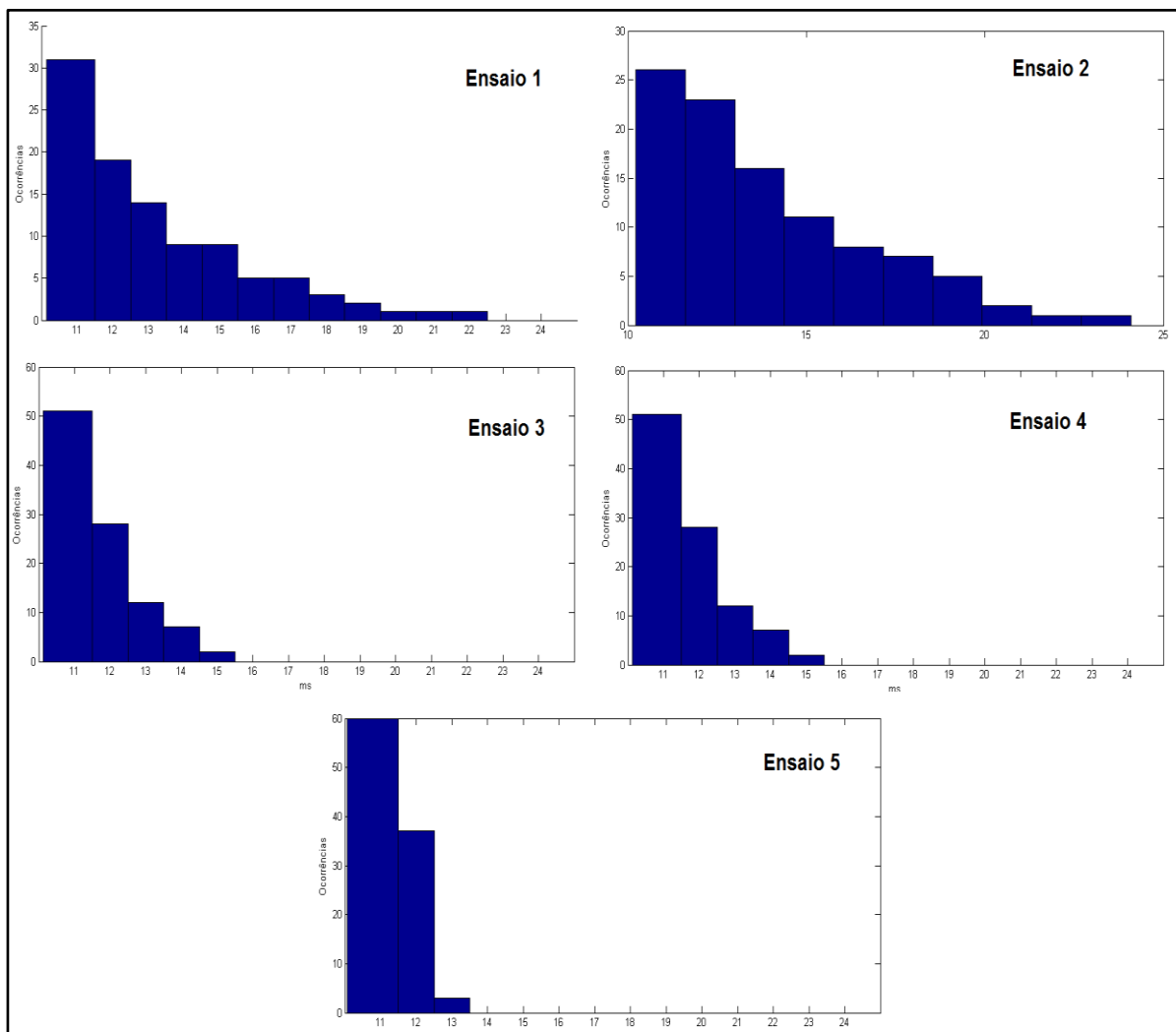


Figura 46 Distribuição períodos de transmissão PGN65265 (Ensaio 1 a 5)

Tabela 23 Distribuição períodos PGN61442 (Ensaio 1 a 5)

| Ensaio | Período Médio (ms) | Maior Período (ms) | Desvio Padrão (ms) |
|----------|--------------------|--------------------|--------------------|
| Ensaio 1 | 103,58 | 117,04 | 4,54 |
| Ensaio 2 | 104,11 | 119,79 | 5,36 |
| Ensaio 3 | 101,32 | 106,85 | 1,87 |
| Ensaio 4 | 101,04 | 105,3 | 1,33 |
| Ensaio 5 | 100,81 | 103,1 | 0,89 |

Com relação aos valores de atraso das mensagens de transmissão assíncrona (*jitter*), apresenta-se a Tabela 24 com os valores obtidos em todos os ensaios realizados.

Tabela 24 *Jitter* na transmissão de mensagens assíncronas

| Módulo | PGN | Ensaio 1 (ms) | Ensaio 2 (ms) | Ensaio 3 (ms) | Ensaio 4 (ms) | Ensaio 5 (ms) |
|--------|----------|---------------|---------------|---------------|---------------|---------------|
| ABS | PGN65215 | 1,42 | 2,54 | 0,608 | 0,606 | 0,34 |
| Cabine | PGN65292 | 0,215 | 0,684 | 0,278 | 0,215 | 0,193 |

No gráfico da Figura 47 podem ser comparados os valores da Tabela 24.

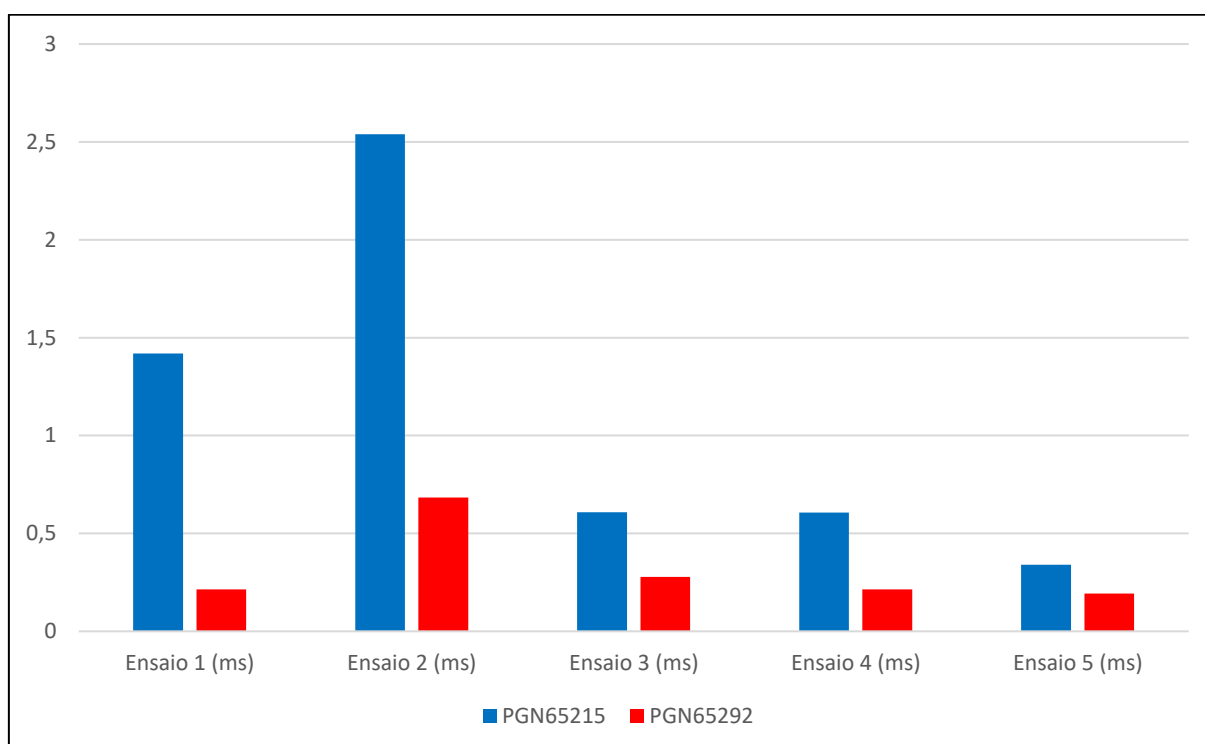


Figura 47 Representação do *Jitter* na transmissão de mensagens assíncronas

7.4 ANÁLISE DE RESULTADOS

Inicialmente, o primeiro ponto a ser analisado refere-se às diferenças encontradas entre as taxas de ocupação calculadas (seção 6.2.1) e os valores simulados (seção 7.3). Apesar destas diferenças terem sido inferiores a 1%, pode-se verificar alguns pontos que permitem justifica-

las. Primeiramente, verificou-se que a base de tempo utilizada no cálculo da taxa de ocupação foi dada em função do maior intervalo de transmissão, 5 segundos neste caso, enquanto que no CANoe, a taxa de ocupação é calculada dentro de um intervalo de 1 segundo. Outro ponto que não foi levado em consideração na realização do cálculo é a variação do comprimento das mensagens em função dos *stuff bits* gerados fora do campo de dados (identificador, DLC e *checksum*).

Outro ponto a ser considerado diz respeito aos resultados gerais das taxas de ocupação obtidos nos ensaios 1 a 5. Segundo (NAVET e PERRAULT, 2012), sistemas de controle tradicionais que buscam desempenho em tempo real deveriam trabalhar com taxas de ocupação inferiores a 35%. Por outro lado, (BJÖRN E TOVAR, 2009) afirmam que, para uma rede CAN 2.0B (com identificador de 29 bits), a taxa de utilização máxima deveria ser de 29% para atender os requisitos de tempo de um sistema padrão. Como pode ser verificado nos resultados apresentados nos ensaios 1 e 2, o uso do protocolo CAN tradicional resulta em uma taxa de utilização da rede superior a estes valores (35% e 29%). Por outro lado, o uso do protocolo CAN FD permite uma redução desta taxa para um valor abaixo deste limite. Além do mais, a proposta de agrupamento de mensagens similares em um mesmo pacote (ensaio 5) permite uma redução ainda mais significativa da taxa de ocupação, ou seja, obtém-se uma diminuição de quase 4 vezes do valor atual do sistema, de 49,21% para 13,85% (Tabela 20).

Com relação à proposta de unificação das redes CAN do chassi e da carroceria (ensaio 2), verifica-se elevação da ocupação da rede (1,36%) e elevação dos intervalos de transmissão das mensagens. Estes resultados justificam a utilização do dispositivo *gateway* para otimização de rede utilizando o padrão CAN clássico. Verifica-se que mensagens de maior velocidade (10, 20, 50 e 100 milissegundos) são as que apresentam maior variação nos tempos de resposta. Conforme dados da Figura 44, verifica-se por exemplo, que a mensagem PGN65449 apresenta atrasos de 64.7% e 82.4% para os ensaios 1 e 2 respectivamente. Destacam-se também os

valores apresentados para a mensagem PGN61442, que indicaram tempos de atraso superiores a 100% (117.4% para o ensaio 1 e 140.9% para o ensaio 2). Esta mensagem demonstra tempos de atraso superiores aos de outras mensagens pois o valor do seu identificador CAN é maior do que o das demais mensagens, o que indica que ela possui menor prioridade no processo de arbitragem.

O ensaio 3, que também propõe uma rede de comunicação unificada, porém utilizando o CAN FD ao invés do CAN clássico, demonstra resultados variados. Além da taxa de ocupação inferior aos ensaios 1 e 2, verifica-se também melhora no desempenho de mensagens de alta velocidade (10, 20, 50, 100 e 250 milissegundos). Porém, quando avaliados os resultados para mensagens de baixa velocidade (1 segundo), verifica-se que o desempenho foi inferior ao sistema original do caminhão (ensaio 1).

Para o ensaio 4, verifica-se pequena redução na taxa de ocupação da rede do chassi em relação aos resultados obtidos no ensaio 3. A mesma característica pode ser observada em relação aos intervalos de atraso das mensagens, ou seja, houve uma pequena redução nos tempos medidos. Estas características devem-se principalmente ao fato de que neste ensaio a quantidade de mensagens transmitidas via padrão CAN FD é menor do que no ensaio 3. Apesar de apresentar resultados próximos ao ensaio 3, o qual utilizava toda a rede configurada como CAN FD, pode-se afirmar que este tipo de arranjo pode ser vantajoso pois exige atualização de apenas uma das atuais redes de comunicação do sistema (rede do chassi).

O ensaio 5, que propõe a utilização de uma rede unificada utilizando mensagens CAN FD em formato agrupado, apresenta os melhores resultados até então observados. Além da baixa taxa de ocupação (13,5%), verifica-se redução significativa nos intervalos de transmissão dos pacotes de mensagens que utilizam os mesmos identificadores. Por exemplo, para a mensagem PGN61442 com intervalo de transmissão esperado de 10 milissegundos, obteve-se como maior intervalo o tempo de 12,14 milissegundos, enquanto que em ensaios anteriores (1

e 2) este valor ultrapassa a barreira dos 20 milissegundos. Esta melhora significativa é decorrente da grande redução na taxa de ocupação da rede uma vez que o valor do identificador se manteve com o mesmo valor das configurações anteriores.

Por fim, com relação aos valores de *jitter* observados para a transmissão de duas mensagens por geração de evento assíncrono (Tabela 24 e Figura 47), pode-se afirmar que os tempos observados possuem relação direta com a taxa de ocupação da rede. Por exemplo, para o ensaio 2, que propôs a unificação das duas redes utilizando o CAN clássico, pode-se verificar o maior tempo de atraso para ambas as mensagens assíncronas e para o ensaio 5, que propôs a unificação das duas redes utilizando mensagens agrupadas via CAN FD, pode-se verificar o menor tempo de atraso. Para estes resultados também é importante mencionar que a diferença verificada nos valores entre o PGN65215 e o PGN65292 deve-se principalmente à prioridade dos identificadores das mensagens CAN e CAN FD. Neste caso, o PGN65292, com valor de identificador igual a 0x0CFF0CB3 (em hexadecimal), possui maior prioridade no envio de mensagens que o PGN65215, o qual possui identificador de maior valor numérico (0x18FEBF0B, em hexadecimal).

8 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho realizou-se análise específica a respeito da utilização do novo protocolo de comunicação CAN FD em sistemas de controle para a linha automotiva pesada. Como estudo de caso, aplicou-se o novo padrão a um sistema de comunicação real de um caminhão fora-de-estrada, o qual possui duas redes de comunicação CAN tradicionais, interconectadas por um dispositivo *gateway*. Uma das redes de comunicação destina-se ao controle do chassi do veículo enquanto outra destina-se ao controle da carroceria.

Com o intuito de verificar distintas possibilidades para utilização do CAN FD, especificou-se cinco distintos arranjos da rede de comunicação do veículo, os quais foram simulados e avaliados utilizando-se o programa de simulação CANoe, desenvolvido pela empresa *Vector Informatik GmbH* (VECTOR, 2015). Para a elaboração dos ensaios, utilizaram-se mensagens CAN reais, padronizadas pelo protocolo de aplicação SAE J1939, o qual determinada a formatação das mensagens e seus respectivos intervalos de transmissão.

Durante simulações do sistema, efetuaram-se basicamente três distintas avaliações:

- a) taxa de ocupação das redes de comunicação;
- b) intervalo de atraso para as mensagens com transmissão síncrona (*time-triggered*);
- c) tempo de atraso para transmissão de mensagens assíncronas (*event-triggered*);

Como resultado registrou-se redução significativa das taxas de ocupação para as redes de comunicação que utilizaram o padrão CAN FD, sobretudo para a rede com maior volume de mensagens (rede do chassi). Também se constatou redução nos intervalos de atraso das mensagens, principalmente mensagens rápidas, com intervalo de transmissão entre 10 milissegundos e 100 milissegundos.

De posse destes resultados e considerando-se a constante evolução dos sistemas de controle veiculares e o aumento da quantidade de dados a serem transmitidos, afirma-se que a utilização do CAN FD é uma alternativa eficaz para prover maior eficiência dos sistemas de comunicação CAN tradicionais. Quanto à possível substituição de sistemas de maior velocidade, como FlexRay por exemplo, pode-se afirmar que o CAN FD ainda possui limitações que não o tornam recomendável para este tipo de implementação.

A realização deste trabalho demonstra que a implementação do CAN FD em sistemas automotivos embarcados possui melhor desempenho quando utiliza-se uma maior quantidade de dados na mensagem (de 8 a 64 bytes), o que pode ser comprovado pelos resultados obtidos no ensaio 5. Para evoluir nesta proposta, considera-se para os trabalhos futuros nova forma de escalonamento das mensagens a fim de utilizar da melhor maneira possível os novos recursos oferecidos pelo CAN FD. Como ideia inicial, considera-se alocar os PGNs especificados pela SAE J1939 dentro do campo de dados do CAN FD, de forma que seja possível transmitir dinamicamente distintas mensagens em um mesmo pacote de dados (Figura 48).

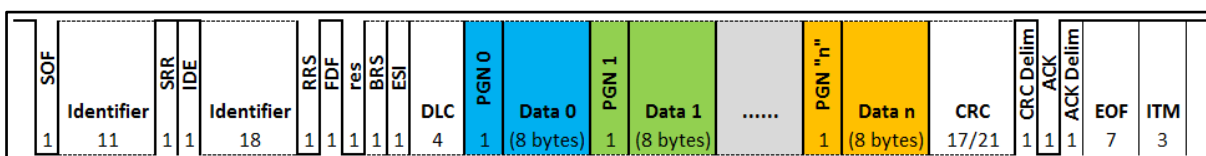


Figura 48 Proposta de distribuição de PGNs em mensagem CAN FD

Outra possibilidade para aumentar a eficiência de transmissão das mensagens utilizando o CAN FD, seria agrupar mensagens conforme indicado na Figura 48, porém reunindo não só mensagens com o mesmo período de transmissão, mas sim várias mensagens com períodos distintos. Por exemplo, seria possível um módulo transmitir mensagens de 100, 250 e 500 milissegundos todas em um único pacote com período de 100 milissegundos. Neste caso fica claro que haveria uma atualização constante de mensagens de menor prioridade, seria então necessário que os sistemas de controle se adaptassem a esta nova taxa de atualização.

Por se tratar de um protocolo novo, ainda existem pontos que devem ser melhor estudados pelos fabricantes, como por exemplo, a compatibilização com sistemas veiculares que utilizam o CAN clássico e também quais os limites de velocidade do campo de dados em função do meio físico e das implicações quanto à emissão de ruído em altas frequências. Também deve ser considerado que muitas das normas de especificação como a própria SAE J1939 ainda não contemplam a utilização do CAN FD e dentro em breve devem ser revisadas para abranger este novo padrão.

REFERÊNCIAS

- ANDERSSON, B.; TOVAR E. The Utilization Bound of Non-Preemptive Rate-Monotonic Scheduling in Controller Area Networks is 25 percent. In: IEEE SYMPOSIUM ON INDUSTRIAL EMBEDDED SYSTEMS, 4., 2009, Lausanne. **Proceedings...** [S.l.: s.n.], 2009. 8 p.
- ADAMSON, T. **Hybrid CAN and CAN FD networks**. 2015. Disponível em: <[http://can-newsletter.org/uploads/media/raw/35a48f1d88ddf2a198638c0a5be51761 .pdf](http://can-newsletter.org/uploads/media/raw/35a48f1d88ddf2a198638c0a5be51761.pdf)>. Acesso em: 10 maio 2016.
- BOSCH GmbH. **CAN Specification Version 2.0**. 1991. Disponível em: <<http://www.semiconductors.bosch.de/pdf/can2spec.pdf>>. Acesso em: 31 mar. 2016.
- BOSCH GmbH. **CAN with Flexible Data-Rate**. 2011. Disponível em: <http://www.bosch-semiconductors.de/media/pdf_1/canliteratur/can_fd.pdf>. Acesso em: 20 mar. 2016.
- CAN IN AUTOMATION. **History of CAN technology**. 2013. Disponível em: <<http://www.can-cia.org/can-knowledge/can/can-history/>>. Acesso em: 30 maio 2016.
- CHEON, B.; JEON, J. W. The CAN FD network performance analysis using the CANoe. In: INTERNATIONAL SYMPOSIUM ON ROBOTICS, 44., 2013, Seoul. **Proceedings...** [S.l.: s.n.], 2013. p. 1-5.
- CORRIGAN, S. **Introduction to the Controller Area Network (CAN)**. 2008. Disponível em: <<http://www.ti.com/lit/an/sloa101a/sloa101a.pdf>>. Acesso em: 30 mar. 2016.
- DAVIS, R. I. et al. Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised. **Journal Real-Time Systems**, New York, v. 35, n. 3, 2007. p. 239 – 272.
- DECKER, P. **CAN Gets Even Better** – Ways to transition from classic CAN to the improved CAN FD. 2013. 4 p. Disponível em: <http://vector.com/portal/medien/cmc/press/Vector/CAN_FD_ElektronikAutomotive_201304_PressArticle_EN.pdf>. Acesso em: 02 out. 2015.
- ELEND, B. **CAN FD: Impact on system design**. CAN in Automation. Kontumazgarten, 2014. 7 p. Disponível em: <http://www.can-cia.org/fileadmin/cia/files/Newsletter_02-14/2-14_p44_can-fd-impact-on-system-design.pdf>. Acesso em: 25 fev. 2016.
- FLEXRAY CONSORTIUM. **FlexRay Communications System Protocol Specification Version 3.0.1**. 2012. Disponível em: <<https://svn.ipd.kit.edu/nlrp/public/FlexRay/FlexRay%E2%84%A2%20Protocol%20Specification%20Version%203.0.1.pdf>>. Acesso em: 17 maio 2016.
- HARTWICH, F. **CAN with Flexible Data-Rate**. Postfach. 2012. 8 p. Disponível em: <http://www.bosch-semiconductors.de/media/ubk_semiconductors/pdf_1/ipmodules_1/can_fd/icc13_2012_paper_Hartwich.pdf>. Acesso em: 5 maio 2016.

HARTWICH, F.; MUTTER, A. **Advantages of CAN FD Error detection mechanisms compared to Classical CAN**. 2015. 8p. Disponível em: <http://www.boschsemiconductors.de/media/ubk_semiconductors/pdf_1/canliteratur/icc_2015_mutter.pdf>. Acesso em: 3 maio 2016.

HELL, M. **The Physical Layer in the CAN FD World**. 2013. 5 p. Disponível em: <http://www.can-cia.org/fileadmin/resources/documents/proceedings/2013_hell.pdf>. Acesso em: 4 maio 2016.

JANG S. J.; JEON J. W. Software reprogramming Performance Analysis of CAN FD and FLEXRAY protocols. In: IEEE INTERNATIONAL CONFERENCE ON INFORMATION AND AUTOMATION, 2015. Lijiang. **Proceedings...** [S.l.: s.n.], 2015. p. 1-6.

JOHANSSON, R.; TORIN, J. **On calculating guaranteed message response times on the SAE J1939 bus**. Göteborg: Chalmers Lindholmen University College, 2002. Disponível em: <<http://www.cse.chalmers.se/~roger/research/papers/chl-techrep-10.pdf>>. Acesso em: 06 maio 2016.

JUNGER, M. **Introduction to J1939** – Version 1.1. Vector Application Note AN-ION-1-3100. 2010. 11 p. Disponível em: <http://vector.com/portal/medien/cmc/application_notes/AN-ION-1-3100_Introduction_to_J1939.pdf>. Acesso em: 21 abr. 2016.

JURGEN, R. K. Multiplex Wiring Systems. In: JURGEN, R. K. (Org.). **Automotive Electronics Handbook**. New York: McGraw-Hill, 1999. p. 26.1-26.76.

LATTICE. **CAN Controller** - Reference Design RD1170. 2013. 11 p. Disponível em: <http://www.latticesemi.com/~media/LatticeSemi/Documents/ReferenceDesigns/1D/CANController-Documentation.PDF?document_id=49554>. Acesso em: 19 abr. 2016.

LENNARTSSON, K. **CAN FD** – Protocol in Detail. 2015. Disponível em: <<http://www.can-cia.org/services/seminars/can-fd-special-seminars/>>. Acesso em: 22 mar. 2016.

LIN CONSORTIUM. **LIN Specification Package** – Revision 2.2A. 2010. 194 p. Disponível em: <www.lin-subbus.org>. Acesso em: 29 set. 2015.

LOPEZ, D.; MOUNIER, P. **Convergence of Bandwidth, Robustness and Energy Saving Challenges on CAN Physical Layer**. 2015. 11 p. Disponível em: <http://cache.nxp.com/files/analog/doc/white_paper/CANWP.pdf>. Acesso em: 12 maio 2016.

LOTOCZKY, R. **CAN-FD Flexible Data Rate CAN** – An Abbreviated Primer. 2012. Disponível em: <www.vector.com>. Acesso em: 20 dez. 2016.

LUPINI, C. A. **In-Vehicle Networking Technology for 2010 and Beyond**. SAE International. 2010. 13 p. Disponível em: <<http://www.skyfi3.delphi.com/pdf/techpapers/2010-01-0687.pdf>>. Acesso em: 12 set. 2015.

NATIONAL INSTRUMENTS. **FlexRay Automotive Communication Bus Overview**. 2009. 6 p. Disponível em: <<http://www.ni.com/white-paper/3352/en/>>. Acesso em: 30 nov. 2015.

NAVET, N.; PERRAULT, H. **Higher busloads for automotive CAN network clusters**. 2012. 4 p. Disponível em: <<http://v2.can-newsletter.org/uploads/media/raw/ac2c68f444cbccff5d6960cb7590204a.pdf>>. Acesso em: 3 maio 2016.

NAVET, N.; SIMONOT-LION, F. A Review of Embedded Automotive Protocols. In: NAVET, N.; SIMONOT-LION, F. (Eds). **Automotive Embedded Systems Handbook, Industrial Information Technology Series**. Boca Raton: CRC Press, Taylor & Francis Group, 2009. cap. 4, p. 97-111.

NGUYEN, T.; CHEON, B. M.; JEON, J. W. CAN FD Performance Analysis for ECU Re-programming using the CANoe. In: IEEE INTERNATIONAL SYMPOSIUM ON CONSUMER ELECTRONICS, 18., 2014, JeJu Island. **Proceedings...** [S.l.: s.n.], 2014. p. 1-4.

NOLTE, T.; HANSSON, H.; NORSTRÖM, C. Probabilistic Worst-Case Response-Time Analysis for the Controller Area Network. In: IEEE REAL-TIME AND EMBEDDED TECHNOLOGY AND APPLICATIONS SYMPOSIUM, 9., 2003, Toronto. **Proceedings...** [S.l.: s.n.], 2003. p. 200-207

PARET, D. **Multiplexed Networks for Embedded Systems: CAN, LIN, FlexRay, Safe-by-Wire**. Chichester: John Wiley and Sons, 2007.

PÉREZ P. et al. Communication Jitter Influence on Control Loops Using Protocols for Distributed Real-Time Systems on CAN Bus. In: INTERNATIONAL SYMPOSIUM ON INTELLIGENT COMPONENTS AND INSTRUMENTS FOR CONTROL APPLICATIONS, 5., 2003, Porto. **Proceedings...** [S.l.: s.n.], 2003. p. 1-7.

QUIGLEY, C.; WILLIAMS, A.; MCLAUGHLIN R. **The Potential of CAN FD Technology to Impact Upon**. 2013. Disponível em: <http://www.can-cia.org/fileadmin/resources/documents/proceedings/2013_quigley.pdf>. Acesso em: 9 maio 2016.

RENESAS. **Introduction to CAN – Application Note**. 2006. 46 p. Disponível em: <http://documentation.renesas.com/doc/products/mpumcu/apn/rej05b0804_m16cap.pdf>. Acesso em: 15 abr. 2016.

RICHARDS, P. **A CAN Physical Layer Discussion**. 2002. 12 p. Disponível em: <<http://ww1.microchip.com/downloads/en/AppNotes/00228a.pdf>> Acesso em: 4 abr. 2016.

SAE INTERNATIONAL. **SAE J1939/71: Vehicle Application Layer**. [S.l.], 2006. 686 p.

SCHREINER, M. et al. **Safe-guarding CAN-FD for applications in trucks**. 2013. 8 p. Disponível em: <http://can-newsletter.org/engineering/standardization/nr_stand_can-fd_detroit_program_121022>. Acesso em: 20 fev. 2016.

TINDELL, K; BURNS, A. Guaranteeing message latencies on Controller Area Network (CAN). In: INTERNATIONAL CAN CONFERENCE, 1., 1994, Mainz. **Proceedings...** [S.l.: s.n.], 1994. 11p.

TINDELL, K.; BURNS, A.; WELLINGS, A. Calculating Controller Area Network (CAN) Message Response Time. **Control Engineering Practice**, Amsterdam, v. 3, n. 8, p. 1163-1169, 1995.

VECTOR. **Product Information CANoe**. 2015. Disponível em: <http://vector.com/portal/medien/cmc/info/CANoe_ProductInformation_EN.pdf>. Acesso em: 5 abr. 2016.

VECTOR. **Programming with CAPL**. 2004. Disponível em: <http://vector.com/portal/medien/vector_cantech/faq/ProgrammingWithCAPL.pdf>. Acesso em: 24 fev. 2016.

APÊNDICE: ALGORITMOS CAPL (CANOE)

Neste apêndice encontram-se os códigos de programa para o módulo CAPL do simulador CANoe. Os códigos apresentados dizem respeito apenas ao funcionamento do ensaio 1 uma vez que todos os demais ensaios possuem modo de programação similar.

A.1 MENSAGENS ECU MOTOR

```

/* Declaração das mensagens */
variables
{
  message PGN65262 PGN65262_;
  message PGN65263 PGN65263_;
  message PGN65270 PGN65270_;
  message PGN65409 PGN65409_;
  message PGN65415 PGN65415_;
  message PGN65416 PGN65416_;
  message PGN61444 PGN61444_;
  message PGN61443 PGN61443_;
  message PGN65247 PGN65247_;
  message PGN60415 PGN60415_;
  message PGN60671_I PGN60671_I_;
  message PGN60671_II PGN60671_II_;
  message PGN65414 PGN65414_;
  message PGN65265_I PGN65265_I_;
  message PGN65266 PGN65266_;
  message PGN65257 PGN65257_;
  message PGN65253 PGN65253_;
  message PGN65226_I PGN65226_I_;
  message PGN65226_II PGN65226_II_;
  message PGN65376 PGN65376_;
  message PGN65271 PGN65271_;

  msTimer Timer_250ms; // 250ms
  msTimer Timer_20ms; // 20ms
  msTimer Timer_50ms; // 50ms
  msTimer Timer_100ms; // 100ms
  msTimer Timer_1000ms; // 1000ms
  msTimer Timer_500ms; // 500ms
  msTimer Timer_5000ms; // 5000ms
}
/* Inicialização dos temporizadores */
on start
{
  setTimer(Timer_250ms,250);
  setTimer(Timer_20ms,20);
  setTimer(Timer_50ms,50);
  setTimer(Timer_100ms,100);
  setTimer(Timer_1000ms,1000);
  setTimer(Timer_500ms,500);
  setTimer(Timer_5000ms,5000);
}

// Estouro de Timer de 250ms

```

```
on timer Timer_250ms
{
  setTimer(Timer_250ms,250);
  output(PGN65247_);
}

// Estouro de Timer de 20ms
on timer Timer_20ms
{
  setTimer(Timer_20ms,20);
  output(PGN61444_);
}

// Estouro de Timer de 50ms
on timer Timer_50ms
{
  setTimer(Timer_50ms,50);
  output(PGN61443_);
}

// Estouro de Timer de 100ms
on timer Timer_100ms
{
  setTimer(Timer_100ms,100);
  output(PGN65409_);
  output(PGN65265_I_);
  output(PGN65266_);
}

// Estouro de Timer de 1000ms
on timer Timer_1000ms
{
  setTimer(Timer_1000ms,1000);
  output(PGN65262_);
  output(PGN65415_);
  output(PGN65416_);
  output(PGN60671_I_);
  output(PGN60671_II_);
  output(PGN65257_);
  output(PGN65253_);
  output(PGN65226_I_);
  output(PGN65226_II_);
  output(PGN65271_);
}

// Estouro de Timer de 500ms
on timer Timer_500ms
{
  setTimer(Timer_500ms,500);
  output(PGN65263_);
  output(PGN65270_);
}

// Estouro de Timer de 5000ms
on timer Timer_5000ms
{
  setTimer(Timer_5000ms,5000);
  output(PGN60415_);
  output(PGN65414_);
  output(PGN65376_);
}
```

A.2 MENSAGENS ECU TRANSMISSÃO

```

/* Declaração das mensagens */
variables
{
  message PGN0_I PGN0_I_;
  message PGN256_I PGN256_I_;
  message PGN61440b PGN61440b_;
  message PGN61441_I PGN61441_I_;
  message PGN61442_ PGN61442_;
  message PGN61445 PGN61445_;
  message PGN65249 PGN65249_;
  message PGN65099 PGN65099_;
  message PGN61452 PGN61452_;
  message PGN61184 PGN61184_;
  message PGN64917 PGN64917_;
  message PGN64932 PGN64932_;
  message PGN65098 PGN65098_;
  message PGN65226 PGN65226_;
  message PGN65272 PGN65272_;
  message PGN65275 PGN65275_;

  msTimer Timer_10ms; // 10ms
  msTimer Timer_20ms; // 20ms
  msTimer Timer_50ms; // 50ms
  msTimer Timer_100ms; // 100ms
  msTimer Timer_1000ms; // 1000ms
  msTimer Timer_5000ms; // 5000ms
}

on start
{
  setTimer(Timer_10ms,10);
  setTimer(Timer_20ms,20);
  setTimer(Timer_50ms,50);
  setTimer(Timer_100ms,100);
  setTimer(Timer_1000ms,1000);
  setTimer(Timer_5000ms,5000);
}

// Estouro de Timer de 10ms
on timer Timer_10ms
{
  setTimer(Timer_10ms,10);
  output(PGN0_I_);
  output(PGN61442_);
}

// Estouro de Timer de 20ms
on timer Timer_20ms
{
  setTimer(Timer_20ms,20);
  output(PGN61452_);
}

// Estouro de Timer de 50ms
on timer Timer_50ms

```



```

{
  setTimer(Timer_50ms,50);
  output(PGN256_I_);
}

// Estouro de Timer de 100ms
on timer Timer_100ms
{
  setTimer(Timer_100ms,100);
  output(PGN61440b_);
  output(PGN61441_I_);
  output(PGN61445_);
  output(PGN61184_);
  output(PGN64932_);
  output(PGN65098_);
}

// Estouro de Timer de 1000ms
on timer Timer_1000ms
{
  setTimer(Timer_1000ms,1000);
  output(PGN64917_);
  output(PGN65226_);
  output(PGN65272_);
  output(PGN65275_);
}

// Estouro de Timer de 5000ms
on timer Timer_5000ms
{
  setTimer(Timer_5000ms,5000);
  output(PGN65249_);
  output(PGN65099_);
}

```

A.3 MENSAGENS ECU ABS

```

variables
{
  message PGN61441_II PGN61441_II_;
  message PGN64998 PGN64998_;
  message PGN65215 PGN65215_;
  message PGN0_II PGN0_II_;
  message PGN255 PGN255_;
  message PGN256_II PGN256_;
  message PGN41 PGN41_;
  message PGN65265_II PGN65265_II_;
  message PGN65274 PGN65274_;
  message PGN65134 PGN65134_;
  message PGN65268 PGN65268_;
  message PGN65103 PGN65103_;
  message PGN65449 PGN65449_;
  message PGN65280 PGN65280_;
  message PGN64964 PGN64964_;
  message PGN57344 PGN57344_;

  msTimer Timer_10ms; // 10ms
  msTimer Timer_20ms; // 20ms
}

```

```

msTimer Timer_50ms; // 50ms
msTimer Timer_100ms; // 100ms
msTimer Timer_1000ms; // 1000ms
msTimer Timer_500ms; // 500ms
}

on start
{
  setTimer(Timer_10ms,10);
  setTimer(Timer_20ms,20);
  setTimer(Timer_50ms,50);
  setTimer(Timer_100ms,100);
  setTimer(Timer_1000ms,1000);
  setTimer(Timer_500ms,500);
  count=0;
}

// Estouro de Timer de 10ms
on timer Timer_10ms
{
  setTimer(Timer_10ms,10);
  output(PGN0_II_);
  output(PGN65449_);
}

// Estouro de Timer de 20ms
on timer Timer_20ms
{
  setTimer(Timer_20ms,20);
  output(PGN65134_);
}

// Estouro de Timer de 50ms
on timer Timer_50ms
{
  setTimer(Timer_50ms,50);
  output(PGN255_);
  output(PGN256_);
  output(PGN41_);
}

// Estouro de Timer de 100ms
on timer Timer_100ms
{
  setTimer(Timer_100ms,100);
  output(PGN61441_II_);
  output(PGN64998_);
  output(PGN65215_);
  output(PGN65265_II_);
  output(PGN65103_);
  output(PGN65280_);
  output(PGN64964_);
}

// Estouro de Timer de 1000ms
on timer Timer_1000ms
{
  setTimer(Timer_1000ms,1000);
  output(PGN65274_);
  output(PGN57344_);
}

```

```

// Estouro de Timer de 500ms
on timer Timer_500ms
{
    setTimer(Timer_500ms,500);
    output(PGN65268_);
}

// Geração de evento para transmissão de mensagem
on key 'p'
{
    @JitterMeasure=timeNow();
    output(PGN65215_);
}

```

A.4 MENSAGENS ECU CABINE

```

variables
{
    message PGN61440 PGN61440_;
    message PGN65294 PGN65294_;
    message PGN65292 PGN65292_;
    message PGN65293 PGN65293_;
    message PGN65269 PGN65269_;

    msTimer Timer_50ms; // 50ms
    msTimer Timer_100ms; // 100ms
    msTimer Timer_1000ms; // 1000ms
}

on start
{
    setTimer(Timer_50ms,50);
    setTimer(Timer_100ms,100);
    setTimer(Timer_1000ms,1000);
}

// Estouro de Timer de 50ms
on timer Timer_50ms
{
    setTimer(Timer_50ms,50);
    output(PGN65294_);
}

// Estouro de Timer de 100ms
on timer Timer_100ms
{
    setTimer(Timer_100ms,100);
    output(PGN61440_);
}

// Estouro de Timer de 1000ms
on timer Timer_1000ms
{
    setTimer(Timer_1000ms,1000);
    output(PGN65292_);
    output(PGN65293_);
}

```

```

    output (PGN65269_);
}
// Evento para transmissão de message
on key 'o'
{
    @ JitterMeasure2=timeNow();
    output (PGN65292_);
}

```

A.5 MENSAGENS ECU CARROCERIA

```

variables
{
    message PGN65292b PGN65292b_;
    message PGN65276 PGN65276_;
    message PGN65198 PGN65198_;
    message PGN53248 PGN53248_;
    message PGN59904 PGN59904_;

    msTimer Timer_1000ms; // 1000ms
    msTimer Timer_5000ms; // 5000ms
}

on start
{
    setTimer(Timer_5000ms,5000);
    setTimer(Timer_1000ms,1000);
}

// Estouro de Timer de 5000ms
on timer Timer_5000ms
{
    setTimer(Timer_5000ms,5000);
    output (PGN53248_);
    output (PGN59904_);
}

// Estouro de Timer de 1000ms
on timer Timer_1000ms
{
    setTimer(Timer_1000ms,1000);
    output (PGN65292b_);
    output (PGN65276_);
    output (PGN65198_);
}

```

A.6 MENSAGENS ECU CHAVE-SETA

```

variables
{
    message PGN64972 PGN64972_;
    message PGN65535 PGN65535_;
    message PGN64973 PGN64973_;

    // 1000ms
    msTimer Timer_1000ms;
}

```

```

on start
{
  setTimer(Timer_1000ms,1000);
}

// Estouro de Timer de 1000ms
on timer Timer_1000ms
{
  setTimer(Timer_1000ms,1000);
  output(PGN64972_);
  output(PGN65535_);
  output(PGN64973_);
}

```

A.7 MENSAGENS GATEWAY

```

variables
{
  float Now_GetTime, Previus_GetTime;
  long count;

  // Mensagens da carroceria
  message PGN65292CHS PGN65292CHSgtw;
  message PGN59904CHS PGN59904CHSgtw;

  // Mensagens do Chassis
  message PGN65262CAR PGN65262CARgtw;
  message PGN65263CAR PGN65263CARgtw;
  message PGN65270CAR PGN65270CARgtw;
  message PGN61444CAR PGN61444CARgtw;
  message PGN61443CAR PGN61443CARgtw;
  message PGN65247CAR PGN65247CARgtw;
  message PGN65265_ICAR PGN65265_ICARgtw;
  message PGN65265_IICAR PGN65265_IICARgtw;
  message PGN65266CAR PGN65266CARgtw;
  message PGN65257CAR PGN65257CARgtw;
  message PGN65253CAR PGN65253CARgtw;
  message PGN65271CAR PGN65271CARgtw;
  message PGN61442CAR PGN61442CARgtw;
  message PGN61445CAR PGN61445CARgtw;
  message PGN61440bCAR PGN61440bCARgtw;
  message PGN0_IICAR PGN0_IICARgtw;
  message PGN256_IICAR PGN256_IICARgtw;
  message PGN61452CAR PGN61452CARgtw;
  message PGN64917CAR PGN64917CARgtw;
  message PGN64932CAR PGN64932CARgtw;
  message PGN65098CAR PGN65098CARgtw;
  message PGN65099CAR PGN65099CARgtw;
  message PGN65249CAR PGN65249CARgtw;
  message PGN65272CAR PGN65272CARgtw;
  message PGN65275CAR PGN65275CARgtw;
}

on start
{
  Now_GetTime=0;
  count=0;
}

```

```
// Mensagens da ECU da Cabine
////////////////////////////////////
//on message PGN65292 // A cada 1000mS
//on message PGN65293 // A cada 1000mS
//on message PGN61440 // a cada 100ms
//on message PGN65294 // a cada 50ms
//on message PGN65269 // a cada 1000ms

// Mensagens da ECU da Carroceria
////////////////////////////////////
//on message PGN65292b // A cada 1000ms
//on message PGN65276 // a cada 1000ms
//on message PGN65198 // a cada 1000ms
//on message PGN53248 // a cada 5000ms
//on message PGN59904 // a cada 5000ms

// Mensagens da chave seta
//on message PGN64972 // a cada 1000ms
//on message PGN65535 // a cada 1000ms
//on message PGN64973 // a cada 1000ms

// Mensagens da Transmissão Allison
//on message PGN0_I // a cada 10 ms
//on message PGN0_61442 // a cada 10ms
//on message PGN256_I // a cada 50ms
//on message PGN61440b // a cada 100ms
//on message PGN61441_I // a cada 100ms
//on message PGN61442 // a cada 10ms
//on message PGN61445 // a cada 100ms
//on message PGN65249 // a cada 5000ms
//on message PGN65099 // a cada 5000ms
//on message PGN61452 // a cada 20ms
//on message PGN61184 // a cada 100ms
//on message PGN64917 // a cada 1000ms
//on message PGN64932 // a cada 100ms
//on message PGN65098 // a cada 100ms
//on message PGN65226 // a cada 1000ms
//on message PGN65272 // a cada 1000ms
//on message PGN65275 // a cada 1000ms

// Mensagens do Freio ABS
//on message PGN61441_II // a cada 100ms
//on message PGN64998 // a cada 100ms
//on message PGN65215 // a cada 100ms
//on message PGN0_II // a cada 10ms
//on message PGN255 // a cada 50ms
//on message PGN256_II // a cada 50ms
//on message PGN41 // a cada 50ms
//on message PGN65265_II // a cada 100ms
//on message PGN65274 // a cada 1000ms
//on message PGN65134 // a cada 20ms
//on message PGN65268 // a cada 500ms
//on message PGN65103 // a cada 100ms
//on message PGN65449 // a cada 10ms
//on message PGN65280 // a cada 100ms
//on message PGN64964 // a cada 100ms
//on message PGN57344 // a cada 1000ms

// Mensagens Módulo Scania
//on message PGN65262 // a cada 1000ms
//on message PGN65263 // a cada 500ms
```

```

//on message PGN65270 // a cada 500ms
//on message PGN65409 // a cada 100ms
//on message PGN65415 // a cada 1000ms
//on message PGN65416 // a cada 1000ms
//on message PGN61444 // a cada 20ms
//on message PGN61443 // a cada 50ms
//on message PGN65247 // a cada 250ms
//on message PGN60415 // a cada 5000ms
//on message PGN60671_I // a cada 1000ms
//on message PGN60671_II // a cada 1000ms
//on message PGN65414 // a cada 5000ms
//on message PGN65265_I // a cada 100ms
//on message PGN65266 // a cada 100ms
//on message PGN65257 // a cada 1000ms
//on message PGN65253 // a cada 1000ms
//on message PGN65226_I // a cada 1000ms
//on message PGN65226_II // a cada 1000ms
//on message PGN65376 // a cada 5000ms
//on message PGN65271 // a cada 1000ms
on message PGN61440 // a cada 100ms
{
    if(count<100)
    {
        count++;
        Previous_GetTime=Now_GetTime;
        Now_GetTime= timeNow();
        @Global_Jitter_Measure=(Now_GetTime-Previous_GetTime)/100;
        write("Message:%d; Time to Transmit:%f",count,@Global_Jitter_Measure);
    }
}

// Mensagens Transmitidas da Carroceria para o Chassis
///////////////////////////////////////////////////////////////////
//
// Posição do pedal do acelerador, retransmite para barramento do chassis
on message PGN65292
{
    output(PGN65292CHSgtw);

    if(count<100)
    {
        count++;
        Previous_GetTime=@JitterMeasure2; // new
        Now_GetTime= timeNow();
        @Global_Jitter_Measure=(Now_GetTime-Previous_GetTime)/1000;
        write("Message:%d; Time to Transmit:%f",count,@Global_Jitter_Measure);
    }
}

// Mensagem de request para PGN 64906
on message PGN59904
{
    output(PGN59904CHSgtw);
}

// Mensagens Transmitidas do Chassis para a Carroceria
///////////////////////////////////////////////////////////////////
on message PGN65262
{
    output(PGN65262CARgtw);
}

```

```
on message PGN65263
{
output (PGN65263CARgtw);
}

on message PGN65270
{
output (PGN65270CARgtw);
}

on message PGN61444
{
output (PGN61444CARgtw);
}

on message PGN61443
{
output (PGN61443CARgtw);
}

on message PGN65247
{
output (PGN65247CARgtw);
}

on message PGN65265_I
{
output (PGN65265_ICARgtw);
}

on message PGN65265_II
{
output (PGN65265_IICARgtw);
}

on message PGN65266
{
output (PGN65266CARgtw);
}

on message PGN65257
{
output (PGN65257CARgtw);
}

on message PGN65253
{
output (PGN65253CARgtw);
}

on message PGN65271
{
output (PGN65271CARgtw);
}

on message PGN61442
{
output (PGN61442CARgtw);
}
}
```



```
on message PGN61445
{
  output (PGN61445CARgtw);
}

on message PGN61440b
{
  output (PGN61440bCARgtw);
}

on message PGN0_II
{
  output (PGN0_IICARgtw);
}

on message PGN256_II
{
  output (PGN256_IICARgtw);
}

on message PGN61452
{
  output (PGN61452CARgtw);
}

on message PGN64917
{
  output (PGN64917CARgtw);
}

on message PGN64932
{
  output (PGN64932CARgtw);
}

on message PGN65098
{
  output (PGN65098CARgtw);
}

on message PGN65099
{
  output (PGN65099CARgtw);
}

on message PGN65249
{
  output (PGN65249CARgtw);
}

on message PGN65272
{
  output (PGN65272CARgtw);
}

on message PGN65275
{
  output (PGN65275CARgtw);
}
```