

HELIO TALIANI JUNIOR

**ESTUDO DOS PROTOCOLOS DE COMUNICAÇÃO DAS
ARQUITETURAS ELETROELETRÔNICAS AUTOMOTIVAS, COM
FOCO NAS SUAS CARACTERÍSTICAS E RESPECTIVAS
APLICAÇÕES, VISANDO O DIRECIONAMENTO PARA O USO
ADEQUADO E CUSTOMIZADO EM CADA CATEGORIA DE
VEÍCULO**

SÃO CAETANO DO SUL

2012

HELIO TALIANI JUNIOR

**ESTUDO DOS PROTOCOLOS DE COMUNICAÇÃO DAS
ARQUITETURAS ELETROELETRÔNICAS AUTOMOTIVAS, COM
FOCO NAS SUAS CARACTERÍSTICAS E RESPECTIVAS
APLICAÇÕES, VISANDO O DIRECIONAMENTO PARA O USO
ADEQUADO E CUSTOMIZADO EM CADA CATEGORIA DE
VEÍCULO**

Monografia apresentada ao curso de Pós Graduação em Engenharia de Processos Industriais – ênfase em Engenharia Automotiva, da Escola de Engenharia Mauá do Centro Universitário do Instituto Mauá de Tecnologia, para obtenção do título de Especialista.

Orientador: Prof. Dr. Marcos Costa Hunold

SÃO CAETANO DO SUL

2012

JUNIOR, Helio Taliani

Estudo dos protocolos de comunicação das arquiteturas eletroeletrônicas automotivas, com foco nas suas características e respectivas aplicações, visando o direcionamento para o uso adequado e customizado em cada categoria de veículo.

Helio Taliani Junior. – São Caetano do Sul, SP: CEUN-EEM, 2012.
97 p.

Monografia (Especialização) – Engenharia de Processos Industriais – ênfase em Engenharia Automotiva.

Escola de Engenharia Mauá do Centro Universitário do Instituto Mauá de Tecnologia, São Caetano do Sul, SP, 2012.

Orientador: Prof. Dr. Marcos Costa Hunold.

1. Protocolo de comunicação. 2. Arquitetura eletroeletrônica. I. Instituto Mauá de Tecnologia. Centro Universitário. Escola de Engenharia Mauá. II. Título.

AGRADECIMENTOS

Primeiramente a Deus, pelo maior presente que um ser humano pode receber que é a oportunidade da vida.

Ao incansável apoio e presença essencial da minha esposa Renata, sem a qual não conseguiria alcançar o êxito na conclusão deste trabalho.

Aos meus pais Helio e Neide pela educação e valores passados a mim.

Ao Professor Dr. Marcos Costa Hunold pelo suporte para a conclusão deste trabalho e obtenção deste título.

RESUMO

Este estudo contempla a análise das características dos protocolos de comunicação utilizados em arquiteturas eletroeletrônicas automotivas, bem como suas diversas aplicações. O aprofundamento desta análise é de extrema importância para a escolha do(s) protocolo(s) a ser(em) utilizado(s) na concepção de uma arquitetura eletroeletrônica, pois tal conhecimento permite que o projeto da arquitetura seja adequadamente desenvolvido. Os custos de desenvolvimento da arquitetura eletroeletrônica, bem como o custo final do veículo são sensivelmente afetados em uma escolha equivocada do(s) protocolo(s) de comunicação. Assim, o objetivo deste trabalho é trazer os detalhes dos diversos protocolos de comunicação automotivos para aplicações adequadas aos diferentes tipos de arquiteturas eletroeletrônicas automotivas.

Palavras-chave: Protocolo de comunicação. Arquitetura eletroeletrônica.

ABSTRACT

This study brings the features analysis of the communication protocols used in the automotive electrical-electronic architectures as well as their applications. The serious study of such analysis is extremely important for the selection of the communication protocols to be used in the concept of an electrical-electronic architecture, as this knowledge allows that the architecture project be correctly developed. The costs of the electrical-electronic architecture development, as well as the final cost of the vehicle, are sensibly affected if a wrong selection of the communication protocols is done. Therefore, the aim of this work is to bring the details of the various automotive communication protocols available for the suitable applications in the different types of automotive electrical-electronic architectures.

Key words: Communication protocol. Electrical-electronic architecture.

LISTA DE ILUSTRAÇÕES

FIGURA 1: Arquitetura eletrônica centralizada.....	12
FIGURA 2: Arquitetura eletrônica distribuída.....	13
FIGURA 3: Arquitetura eletrônica do modelo Fiat Palio com barramento CAN e interface de diagnóstico.....	15
FIGURA 4: Arquitetura eletrônica hierárquica para o modelo Fiat Stilo	16
FIGURA 5: Arquitetura eletroeletrônica de um modelo Volvo XC90	16
FIGURA 6: Arquitetura eletroeletrônica e características dos barramentos de comunicação de um modelo BMW 7	17
FIGURA 7: Arquitetura eletroeletrônica do modelo Citroën C5 com tecnologias CAN e VAN	18
FIGURA 8: Arquitetura eletroeletrônica do modelo VW Constellation para caminhões	19
FIGURA 9: Arquitetura eletroeletrônica de caminhões do modelo Volvo FH.....	20
FIGURA 10: Estrutura em camadas do modelo OSI	21
FIGURA 11: Estrutura em camadas de software para ECU automotiva baseada no modelo AUTOSAR	23
FIGURA 12: Arquitetura distribuída hierárquica para veículos de passeio da Volvo Cars.....	24
FIGURA 13: Classificação das redes automotivas quanto ao custo e taxas de transmissão....	25
FIGURA 14: Sistema de entretenimento em um automóvel.....	27
FIGURA 15: Tecnologias de redes sem fio (wireless) para sistemas de transporte.....	28
FIGURA 16: Dispositivos de CI interconectados via barramento I ² C.....	31
FIGURA 17: Interfaces de conexão com barramento I ² C entre dispositivos.....	32
FIGURA 18: Diagrama de tempo durante a transmissão de um dado válido	32
FIGURA 19: Diagrama de tempo para as condições de início e fim da transmissão de um frame	33
FIGURA 20: Diagrama de tempo para transmissão de dados.....	34
FIGURA 21: Sincronização de clock do SCL.....	35
FIGURA 22: Arbitragem de transmissão de dados pelo DAS	36
FIGURA 23: Sinais de interface entre master e slave no SPI	36
FIGURA 24: Sinais de interface entre dispositivos mestre e escravo com barramento SPI....	37
FIGURA 25: Fluxo de controle para comunicação entre dispositivos mestre e escravo com barramento SPI	38
FIGURA 26: Comunicação serial assíncrona IBM-PC via sistema de telefonia	39
FIGURA 27: Comunicação serial assíncrona mainframe-terminais	39
FIGURA 28: Controlador UART	40
FIGURA 29: Sequência de sinais de controle para transmissão no UART	40
FIGURA 30: Grupo de consórcio inicial do LIN	41
FIGURA 31: Arquitetura distribuída com tecnologia LIN para funções de controle na porta	42
FIGURA 32: Estrutura funcional do consórcio LIN	42
FIGURA 33: Arquitetura automotiva com tecnologias CAN e LIN.....	44
FIGURA 34: Detalhes do transceiver LIN 2.0	44
FIGURA 35: Determinação do nível lógico sob barramento LIN	45
FIGURA 36: Estrutura de um barramento LIN com componentes mestre e escravo	45
FIGURA 37: Sequência de execução de frame LIN entre tarefas para comunicação.....	46
FIGURA 38: Estrutura de um frame LIN.....	46
FIGURA 39: O round e quadros do protocolo TTP/A	49
FIGURA 40: Estrutura de uma célula de 13 bits do TTP/A no protocolo UART	49
FIGURA 41: Uma mensagem J1587 contém dois PIDs, 21 e 12	50
FIGURA 42: Comparativo da estrutura em camadas para os padrões CAN e OSI	53

FIGURA 43: Relação entre comprimento da rede e taxa de transmissão	54
FIGURA 44: Formato de frames CAN 2.0A e CAN 2.0B	55
FIGURA 45: Processo de arbitragem entre frames sob o barramento CAN	56
FIGURA 46: Estrutura de um frame de dados CAN	56
FIGURA 47: Estrutura de um frame remoto CAN	57
FIGURA 48: Estrutura de um frame de erro CAN	57
FIGURA 49: Diagrama de ligação de um barramento CAN com resistores de terminação	58
FIGURA 50: Especificações do padrão SAE J1939	59
FIGURA 51: Identificador CAN na mensagem J1939	61
FIGURA 52: As interfaces do ASAP	62
FIGURA 53: Conexão do barramento CCP	62
FIGURA 54: Organização da mensagem CRO	63
FIGURA 55: Organização da mensagem DTO	63
FIGURA 56: Instante de amostragem do sinal e tempo de bit para transmissão	64
FIGURA 57: Mecanismo de transmissão de informação através da transição de estado do sinal	64
FIGURA 58: Estrutura de um frame para tecnologia VAN	65
FIGURA 59: Processo de comunicação da rede VAN	66
FIGURA 60: Estrutura de mensagem KWP 2000 com o campo Fmt	67
FIGURA 61: Arquitetura de diagnose para uma rede KWP 2000	67
FIGURA 62: Primitivas de comunicação entre elementos no KWP 2000	68
FIGURA 63: Exemplo de fluxo de mensagens para diagnose	68
FIGURA 64: Mensagem SAE J1850 com IFR	70
FIGURA 65: Tempo de bit do SAE J1850 VWP	70
FIGURA 66: Arquitetura distribuída com backbone FlexRay	73
FIGURA 67: Topologias para rede FlexRay: a) barramento; b) estrela e c) híbrida	74
FIGURA 68: Ciclo de comunicação FlexRay	75
FIGURA 69: Arquitetura steer-by-wire com tecnologia TTP/C	76
FIGURA 70: Topologia em duplo barramento para o TTP/C	76
FIGURA 71: Topologia com um cluster TTP/C com unidades FTU e SRU	77
FIGURA 72: Tipos de frame do TTP/C	78
FIGURA 73: Esquema de acesso ao meio do TTP/C	78
FIGURA 74: Grupo de empresas responsáveis pelo ByteFlight	79
FIGURA 75: Diagrama temporal do protocolo FTDMA	80
FIGURA 76: Exemplo de uma rede Safe-by-Wire	81
FIGURA 77: Estados de tensão do barramento diferencial	82
FIGURA 78: D-frame com o endereçamento ponto a ponto	83
FIGURA 79: D-frame com endereçamento bitmap	83
FIGURA 80: S-frame com três sensores e 4 bits de dados	83
FIGURA 81: Estrutura de um frame D2B	85
FIGURA 82: Configuração em anel D2B (Class M)	86
FIGURA 83: Salto de frequência em uma rede Bluetooth	91
FIGURA 84: Fluxo de comandos para reconhecimento Bluetooth	92
FIGURA 85: Configurações de redes Bluetooth: piconet e scatternet	92
FIGURA 86: Topologia da aplicação de UWB para comunicação intra-veicular	94
FIGURA 87: Linha do tempo para a evolução dos sistemas eletrônicos automotivos	95

LISTA DE TABELAS

TABELA 1 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE A	25
TABELA 2 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE B	26
TABELA 3 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE C	27
TABELA 4 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE ENTRETENIMENTO	28
TABELA 5 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE WIRELESS.....	29
TABELA 6 – TIPOS DE REDE CAN	54

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Objetivo	11
1.2	Justificativa	11
2	ARQUITETURAS ELETROELETRÔNICAS AUTOMOTIVAS	12
2.1	Arquiteturas eletroeletrônicas de veículos de passeio	14
2.2	Arquiteturas eletroeletrônicas de caminhões e ônibus	18
3	REDES DE COMUNICAÇÃO AUTOMOTIVAS	21
3.1	Classificações das redes automotivas	22
3.2	Redes automotivas Classe A (funções de conforto e diagnóstico)	25
3.3	Redes automotivas Classe B (funções relacionadas à dinâmica do veículo)	26
3.4	Redes automotivas Classe C (funções de segurança crítica)	26
3.5	Redes automotivas Classe Entretenimento (funções de entretenimento)	27
3.6	Redes automotivas Classe Wireless (funções de logística e roteirização)	28
4	REDES AUTOMOTIVAS CLASSE A	30
4.1	I ² C (Inter-Integrated Circuit)	30
4.2	SPI (Serial Peripheral Interface)	36
4.3	UART (Universal Asynchronous Receiver/Trasnmitter)	38
4.4	LIN (Local Interconnect Network)	41
4.5	TTP/A (Time-Triggered Protocol/Class A)	47
4.6	J1587/J1708	50
5	REDES AUTOMOTIVAS CLASSE B	52
5.1	CAN (Controller Area Network)	52
5.2	SAE J1939	58
5.3	CCP (CAN Calibration Protocol)	61
5.4	VAN (Vehicle Area Network)	64
5.5	K-Line (ISO 9141)	66
5.6	KWP2000 (Keyword Protocol 2000)	66
5.7	SAE J1850	69
6	REDES AUTOMOTIVAS CLASSE C	72
6.1	FlexRay	72
6.2	TTP/C (Time-Triggered Protocol/Class C)	75
6.3	ByteFlight	79
6.4	Safe-by-Wire	81
7	REDES DE ENTRETENIMENTO	84
7.1	D2B (Domestic Digital Bus)	84
7.2	IDB-1394	86
7.3	MOST (Media Oriented System Transport)	88
8	REDES SEM FIO	90
8.1	Bluetooth	90
8.2	UWB (Ultra Wide Band)	93
8.3	ZigBee	94
9	O FUTURO DAS REDES DE COMUNICAÇÃO AUTOMOTIVAS	95
10	CONCLUSÃO	96
	REFERÊNCIAS	97

1 INTRODUÇÃO

O conteúdo elétrico e eletrônico dos veículos aumenta a cada dia, com a busca por satisfazer as exigências do mercado. Novos dispositivos utilizados para diversas funções, como por exemplo, sistemas de navegação, entretenimento, conforto, telemática etc. precisam na maioria das vezes apresentarem uma convergência entre si. Ou seja, há a necessidade que estes estejam conectados para propiciar um universo de funções que atenderão a demanda de mercado e até mesmo proporcionar um diferencial para o cliente.

Este trabalho é fundamentado através do detalhamento das características dos diferentes protocolos de comunicação automotivos e suas aplicações nas diversas topologias de arquitetura.

Primeiramente, são apresentadas as topologias de arquiteturas eletroeletrônicas automotivas e suas características. Em seguida, as redes de comunicação automotivas são descritas e classificadas. Por fim, os detalhes técnicos de cada protocolo de comunicação, dentro de suas classificações, são expostos. Tais detalhamentos técnicos dos protocolos de comunicação foram pesquisados no livro *Redes de Comunicação Automotiva – Características, Tecnologias e Aplicações*, também citado nas referências deste trabalho.

1.1 Objetivo

O objetivo deste estudo é a análise dos protocolos de comunicação automotivos, visando o suporte para o desenvolvimento de projetos de arquiteturas eletroeletrônicas automotivas.

1.2 Justificativa

O estudo das arquiteturas eletroeletrônicas automotivas e dos respectivos protocolos de comunicação é de extrema importância para a indústria automobilística. Uma compilação dos dados técnicos pertinentes a este tema nem sempre é facilmente encontrada, sendo um estudo deste tipo de grande valor como fonte de futuras pesquisas.

2 ARQUITETURAS ELETROELETRÔNICAS AUTOMOTIVAS

As arquiteturas eletroeletrônicas automotivas constituem os elementos responsáveis por todo o sistema elétrico do veículo: alternador, bateria, cabos de transmissão de sinais e energia (chicotes), centrais elétricas, módulos eletrônicos (ECUs), sensores e atuadores.

Uma das principais características de uma arquitetura eletroeletrônica automotiva é dada pela forma como os seus elementos estão interligados, em especial as ECUs. Isto direcionará quais funções estarão presentes no veículo, a maneira como estas funções se relacionarão, bem como suas distribuições pelas ECUs.

Basicamente, observando-se as conexões dos elementos de uma arquitetura eletroeletrônica, existem dois tipos de estrutura: a arquitetura centralizada e a arquitetura distribuída.

Em uma arquitetura centralizada, apenas uma ECU assume o papel de controlar uma gama de funções do veículo. Assim, esta ECU possui entradas responsáveis por receber os sinais provenientes de diversos sensores. Tais sinais são processados internamente e o resultado são ações que consistem no controle de atuadores realizado por saídas presentes nesta mesma ECU. A FIGURA 1 ilustra este tipo de arquitetura.

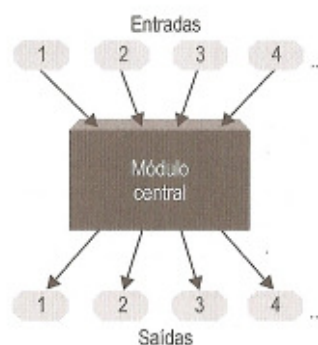


FIGURA 1: Arquitetura eletrônica centralizada
Fonte: SANTOS (2010)

A principal vantagem deste tipo de arquitetura é a simplicidade que ela traz, pois o hardware é composto somente de uma ECU, sensores, atuadores e cabos para interconexão destes elementos. A desvantagem fica por conta da grande quantidade de cabos necessária para conectar todos os sensores e atuadores à ECU, uma vez que independente da localização destes, os mesmos precisam estar ligados à ECU. Além disso, qualquer alteração necessária

(acréscimo de funções, por exemplo) exigiria modificações no software e até mesmo no hardware do único módulo eletrônico do veículo.

Já a arquitetura distribuída, como o próprio nome sugere, propicia uma distribuição das funções, uma vez que se vale de diversas ECUs conectadas entre si por um barramento de comunicação. Tal barramento utiliza um protocolo de comunicação e um veículo pode ter mais de um barramento de comunicação disponível. Os protocolos de comunicação são definidos por GUIMARÃES (2011) da seguinte forma:

Protocolos de comunicação são meios de transmissão e recepção de dados utilizados para intercomunicar módulos eletrônicos e/ou sensores e atuadores inteligentes equipados com microcontroladores e transceivers, por exemplo. Existem vários tipos de protocolos de comunicação, cada qual com suas características técnicas específicas e, portanto, com as suas aplicações mais apropriadas. (GUIMARÃES, 2011, p.209)

Como exemplo de arquitetura distribuída, utilizam-se algumas ECUs dedicadas à leitura de sinais (entradas) e outras ECUs responsáveis pelo controle de atuadores (saídas), conforme ilustrado na FIGURA 2.

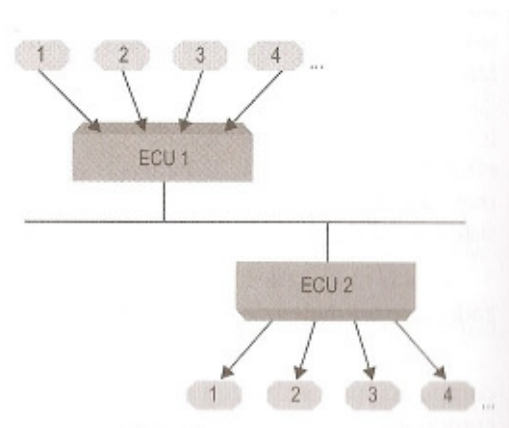


FIGURA 2: Arquitetura eletrônica distribuída
Fonte: SANTOS (2010)

Uma vantagem clara da arquitetura distribuída é a menor quantidade de cabos necessária para interligação de seus elementos, uma vez que as ECUs podem ser estrategicamente colocadas próximas aos elementos que serão conectados a elas. Esta vantagem propicia uma maior simplicidade, e conseqüentemente menor tempo, na montagem do veículo, bem como oferece uma maior flexibilidade para alterações, uma vez que as funções estão distribuídas entre diversas ECUs. Existe ainda a possibilidade de se agregar mais funções, pois há um maior número de módulos eletrônicos presentes no veículo.

A desvantagem de uma arquitetura deste tipo é a necessidade de utilização de um meio de comunicação entre as ECUs, o que, geralmente, encarece a arquitetura, pois há a necessidade de se aumentar a porção eletrônica presente nos módulos eletrônicos do veículo.

Para GUIMARÃES (2012), a escolha do melhor tipo de arquitetura depende de algumas características do projeto dos sistemas eletroeletrônicos do veículo. Quando se tem um conteúdo eletrônico reduzido, com menor número de variáveis de entrada e saída e recursos reduzidos para validação, o que caracteriza um projeto de custo mais baixo, então provavelmente a melhor escolha é por uma arquitetura centralizada.

Já quando existe um maior conteúdo tecnológico no veículo, onde naturalmente o número de entradas e saídas é maior, sugere-se a utilização de mais ECUs na configuração de uma arquitetura distribuída.

Conforme SANTOS (2010), não existe um padrão para a topologia das arquiteturas distribuídas utilizadas entre as montadoras:

As montadoras automotivas oferecem produtos veiculares com uma vasta gama de funções proprietárias ou não (disponibilizadas por fornecedores), e em função disso podem estruturar suas arquiteturas distribuídas de forma mais eficiente possível, objetivando reduzir custos de produção, diagnóstico, manutenção e melhoria constante na qualidade dos serviços oferecidos.
(SANTOS, 2010, p.45)

Segundo GUIMARÃES (2011, p.204), analisando cada tipo de arquitetura (centralizada e distribuída), pode-se ver que a arquitetura distribuída é mais interessante atualmente, pois proporciona maior flexibilidade na expansão de funções nos veículos. Assim, com o cenário atual (e futuro) de uma alta demanda por agregação de valores aos veículos, a arquitetura distribuída tem a tendência em ser mais utilizada.

2.1 Arquiteturas eletroeletrônicas de veículos de passeio

Para exemplificar o que foi apresentado sobre as arquiteturas eletroeletrônicas distribuídas, são dados a seguir alguns exemplos reais de arquiteturas utilizadas pelas montadoras de veículos. Muitas das tecnologias de redes que serão citadas nesta seção do trabalho serão detalhadas nos próximos capítulos.

Nas duas figuras a seguir são apresentadas arquiteturas dos veículos Fiat Palio e Fiat Stilo. Observa-se na FIGURA 3 que, para o Fiat Palio, a arquitetura apresentada é mais simples, composta de um barramento CAN (Controller Area Network) com três ECUs (a do painel de instrumentos, NQS, a do módulo de controle do motor, NCM e a do módulo de controle da carroceria, NBC) mais as linhas de diagnose dedicadas para cada função restante (ABS, airbag e imobilizador). O diagnóstico das ECUs presentes no barramento CAN é realizada através do módulo de controle do motor, o qual possui a interface de diagnóstico com o conector de diagnóstico do veículo, DLC.

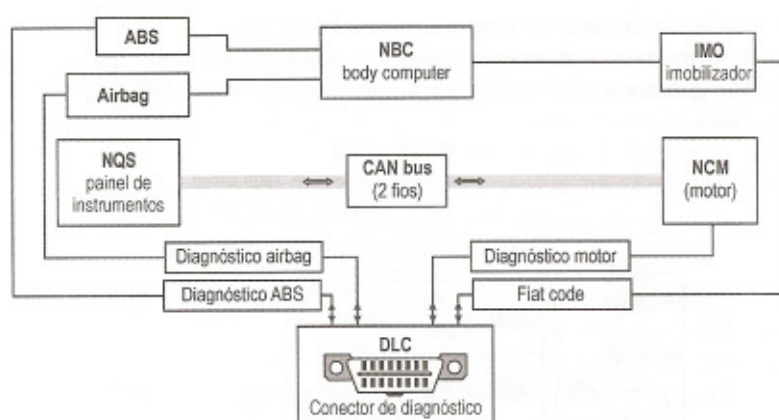


FIGURA 3: Arquitetura eletrônica do modelo Fiat Palio com barramento CAN e interface de diagnóstico
Fonte: SANTOS (2010)

Já a arquitetura do veículo Fiat Stilo é visivelmente mais complexa, pois se vale de 22 ECUs. Assim, a montadora optou por dividi-las em mais de uma rede, distribuindo-as em grupos de funções (powertrain e chassis, body electronics e conforto).

Na FIGURA 4 é apresentado o barramento C-CAN, responsável por interligar as ECUs que suportam as funções mais críticas do veículo, aquelas que exigem uma maior taxa de transmissão de dados (500kbps). As ECUs com funções de carroçaria (body) estão conectadas pelo barramento B-CAN cuja taxa de transmissão é de 50kbps, suficiente para este tipo de função. Por fim, utiliza-se o barramento A-bus com uma taxa de transmissão de 4,8kbps, suficiente para suportar a conexão entre as ECUs com funções de conforto. O NBC (Node Body Computer) funciona como gateway dos três barramentos, interligando-os e permitindo a troca de informações entre todas as ECUs, independente do barramento em que estejam conectadas.

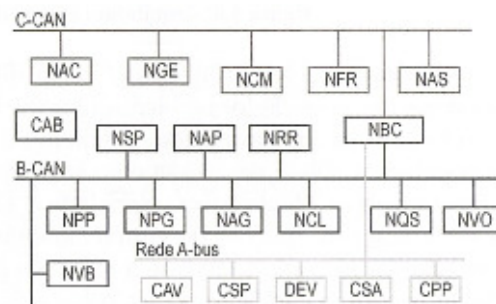


FIGURA 4: Arquitetura eletrônica hierárquica para o modelo Fiat Stilo
Fonte: SANTOS (2010)

Exemplos mais complexos podem ser vistos na indústria automotiva, como a arquitetura utilizada pela Volvo Cars no modelo XC90. Neste veículo, conforme se observa na FIGURA 5, existe uma grande quantidade de ECUs (40) conectadas entre si por diversas redes/barramentos de forma hierárquica. Os três grupos funcionais presentes neste veículo são: powertrain/chassis, entretenimento e body electronics. As ECUs do grupo de powertrain/chassis são conectadas por um barramento CAN de alta velocidade (500kbps). Já as ECUs do grupo body electronics utilizam uma rede CAN de baixa velocidade (125kbps) para se conectarem e, por fim, uma rede MOST (Media Oriented System Transport) interconecta as ECUs com a função de entretenimento. Existe ainda uma rede LIN (Local Interconnect Network) para conexão das ECUs que funcionam como escravas para as funções de conforto. O módulo eletrônico central (CEM) funciona como o gateway da arquitetura.

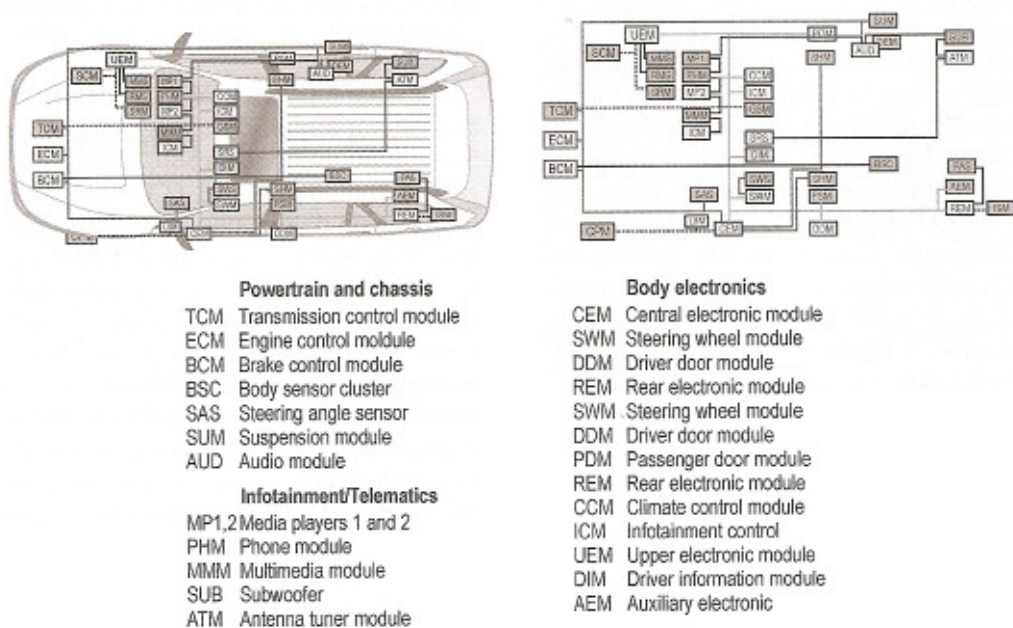


FIGURA 5: Arquitetura eletroeletrônica de um modelo Volvo XC90
Fonte: SANTOS (2010)

Aumentando a complexidade das arquiteturas dos veículos, pode-se analisar a topologia utilizada no modelo BMW 7. A arquitetura deste veículo utiliza 65 ECUs divididas em quatro grupos funcionais (powertrain/chassis, body electronics, entretenimento e segurança passiva). Para o primeiro grupo é utilizado um barramento CAN de alta velocidade com taxa de transmissão de 500kbps. O grupo body electronics utiliza também um barramento CAN, porém de baixa velocidade (100kbps). Já o grupo de entretenimento utiliza um barramento MOST com taxa de transmissão de 22 Mbps, enquanto o grupo de segurança passiva utiliza um barramento ByteFlight, cuja velocidade é de 10Mbps, além de ser uma tecnologia de propriedade da BMW.

A FIGURA 6 mostra a topologia e as características dos barramentos de comunicação utilizados no modelo BMW 7. Pode-se ver que existem diferentes combinações de arquiteturas de redes, como anel, estrela e barramento. Um gateway para a interligação das redes também está presente nesta arquitetura, possibilitando a comunicação entre os diversos sistemas do veículo.

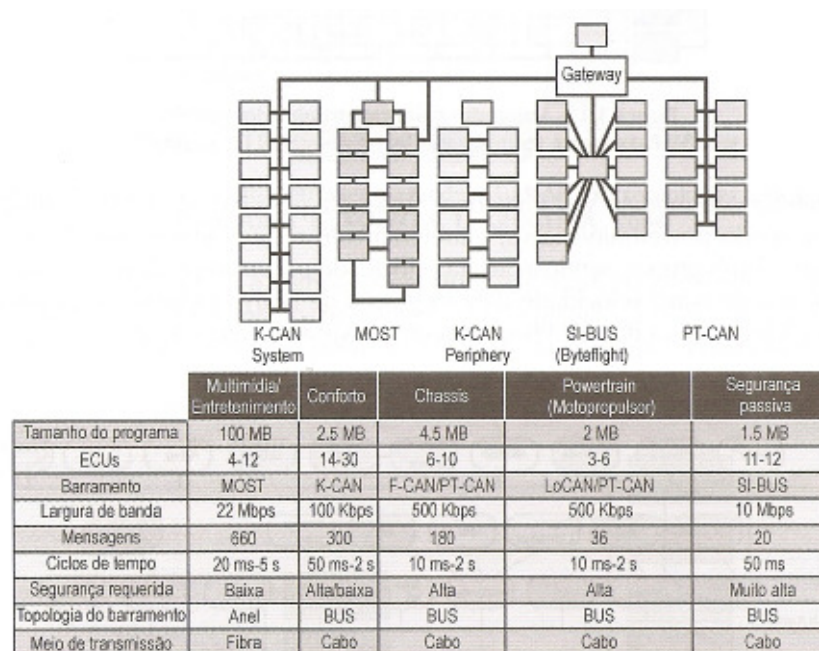


FIGURA 6: Arquitetura eletroeletrônica e características dos barramentos de comunicação de um modelo BMW

7

Fonte: SANTOS (2010)

Outro exemplo de arquitetura é a topologia do modelo Citroën C5. Como indicado na FIGURA 7, tal arquitetura é composta por 20 ECUs pertencentes a quatro grupos de funções: powertrain/chassis, entretenimento, body electronics e segurança crítica. Somente o grupo de

powertrain/chassis utiliza uma rede CAN. Os outros três grupos utilizam um barramento VAN (Vehicle Area Network).

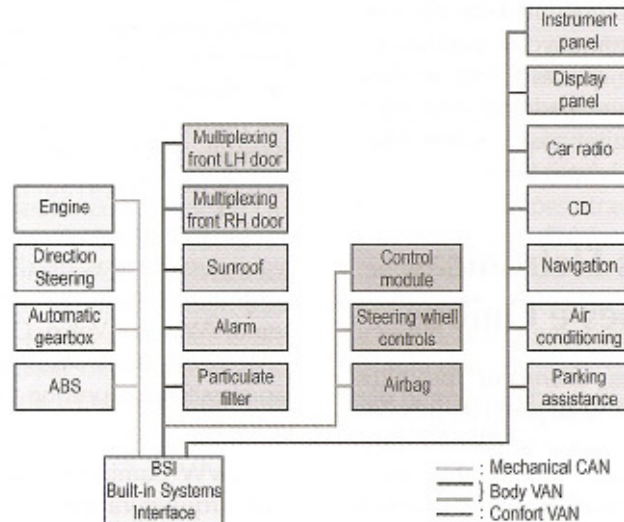


FIGURA 7: Arquitetura eletroeletrônica do modelo Citroën C5 com tecnologias CAN e VAN
Fonte: SANTOS (2010)

2.2 Arquiteturas eletroeletrônicas de caminhões e ônibus

Da mesma forma que os veículos de passeio, os veículos pesados também utilizam ECUs para as diversas funções presentes e, assim, também é necessária a definição de uma arquitetura eletroeletrônica para a interligação destes módulos eletrônicos.

Estes veículos também utilizam topologias de arquiteturas distribuídas, sendo em geral formadas por pelo menos uma rede de controle e uma rede de diagnóstico.

Somente para exemplificar, serão mostradas aqui duas arquiteturas de veículos pesados. A primeira delas, apresentada na FIGURA 8, é a da linha VW Constellation, cuja arquitetura é bem simples. Nela existem somente dois barramentos CAN, sendo um utilizado para as funções do veículo, o qual interliga todas as ECUs e o outro barramento utilizado exclusivamente para a conexão da ferramenta de diagnóstico, através do módulo de controle da carroçaria (Body Control Unit).

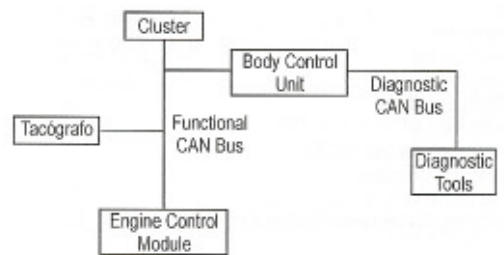


FIGURA 8: Arquitetura eletroeletrônica do modelo VW Constellation para caminhões
Fonte: SANTOS (2010)

Uma arquitetura mais complexa é a do caminhão modelo Volvo FH. Tal arquitetura interliga as ECUs com base em alguns padrões SAE, como o SAE J1939 (controle) e o SAE J1587/1708 (diagnóstico). O padrão SAE J1939 é feito a partir do protocolo CAN com uma taxa de transmissão de 250kbps. Já o padrão J1587 é utilizado em funções de diagnose e trabalha em conjunto com as interfaces físicas descritas pelo padrão J1708. Tal padrão (SAE J1587/1708) possui taxa de transmissão de 9,6kbps.

A arquitetura do Volvo FH, apresentada na FIGURA 9, é formada por três grupos funcionais: powertrain e veículo, segurança passiva, entretenimento. As ECUs do primeiro grupo são interligadas pelos barramentos de controle (J1939-1) e diagnose (J1708/J1587). Já as ECUs do segundo grupo (segurança passiva) utilizam um barramento J1587/J1708 para se conectarem. Por fim, as ECUs do grupo de entretenimento são interligadas por um barramento padrão J1939-3.

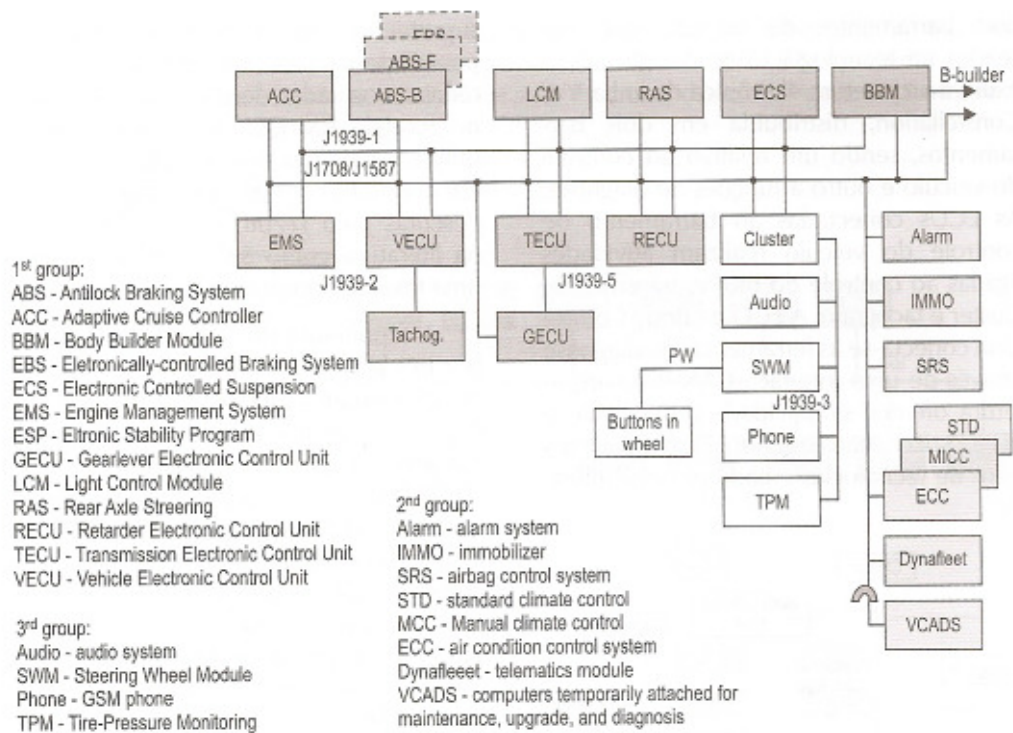


FIGURA 9: Arquitetura eletroeletrônica de caminhões do modelo Volvo FH
Fonte: SANTOS (2010)

3 REDES DE COMUNICAÇÃO AUTOMOTIVAS

Conforme apresentado na introdução, as redes/barramentos de comunicação são de extrema importância para as arquiteturas eletroeletrônicas automotivas. Basicamente, tais redes proporcionam a comunicação entre as ECUs, a fim de tornar possível o funcionamento correto de todos os sistemas eletroeletrônicos do veículo, bem como permitir a conexão com dispositivos de diagnóstico externos.

A princípio, olhando somente para os diagramas em blocos das arquiteturas, a comunicação entre as ECUs parece simples. Porém, para que tal comunicação ocorra de forma robusta é necessário que se tenha de forma clara a definição dos níveis envolvidos em um processo de comunicação. Para isso, a ISO (International Standard Organization) criou o modelo de camadas OSI (Open System Interconnection), o qual define claramente os níveis presentes em um processo de comunicação. A FIGURA 10 mostra o modelo de camadas OSI para a comunicação entre dispositivos.



7	Aplicação
6	Apresentação
5	Sessão
4	Transporte
3	Rede
2	Enlace
1	Física

FIGURA 10: Estrutura em camadas do modelo OSI
Fonte: SANTOS (2010)

A camada de aplicação é a camada superior responsável pelos serviços destinados às aplicações superiores, de forma a abstrair a existência de comunicação entre processos entre diferentes ECUs.

Já a camada de apresentação assegura a forma como a informação será apresentada, ou seja, ela assegura que o formato da informação será entendido pelo receptor. Diferentemente das camadas mais inferiores, esta camada não lida com os dados quanto a bits, mas sim com sua representação.

A camada de sessão controla os diálogos entre dois processos de aplicação. Nesta camada acontece a identificação dos blocos de comunicação, para que não ocorra uma recarga de dados. Por exemplo, para evitar a perda de dados em uma comunicação, é utilizado o conceito de sincronização, onde o receptor deve enviar uma resposta confirmando o recebimento da mensagem sempre que receber um ponto de sincronização. Assim, caso alguma interrupção ocorra na transmissão, a mesma é iniciada a partir do último ponto de sincronização.

A camada de transporte é responsável por dividir os dados recebidos da camada de sessão em pacotes para enviá-los à camada de rede. No receptor, tal camada é responsável por recolher os pacotes recebidos da camada de rede e remontar o dado original para ser enviado à camada de sessão.

O controle da operação de rede através de endereçamento de ECUs e roteamento é realizado pela camada de rede. Esta camada torna transparente para a camada de transporte a forma como os recursos dos níveis inferiores são utilizados para implementar as conexões de rede.

A conexão confiável a um meio físico é papel da camada de enlace de dados, a qual recebe e transmite uma sequência de bits para o nível físico, garantindo que esta sequência esteja livre de erros. Controla ainda a taxa de transmissão de dados, a fim de evitar que o transmissor envie as mensagens a uma taxa maior que o receptor possa gerenciar.

A camada física é responsável pela codificação/decodificação das mensagens em sinais elétricos que serão passados para o meio físico. Sua função é transmitir uma sequência de bits por um canal de comunicação, sendo o protocolo deste nível responsável por fazer que um bit “1” transmitido pelo transmissor seja corretamente interpretado como bit “1” pelo receptor e não como bit “0”. Assim, esta camada trabalha com as características do meio físico, como por exemplo, os níveis de tensão elétrica que devem representar os sinais lógicos “0” e “1”.

3.1 Classificações das redes automotivas

Conforme citado anteriormente, as ECUs de um veículo devem estar conectadas através de uma interface e barramento de comunicação para possibilitar à arquitetura a correta execução das diversas funções do veículo.

A FIGURA 11 apresenta a estrutura de uma ECU (Electronic Control Unit) automotiva, constituída por um sistema computacional, cuja arquitetura é formada por parte de hardware, sistema operacional de tempo real (RTOS), unidades de entrada e saída (E/S), rede de comunicação e aplicação. Tal estrutura pode variar conforme as especificações de hardware e software de cada montadora.

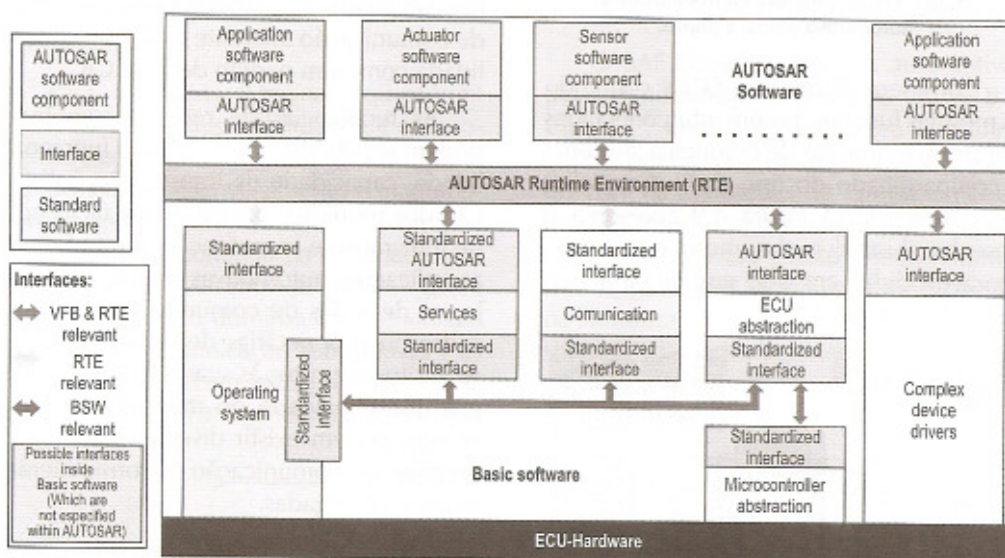


FIGURA 11: Estrutura em camadas de software para ECU automotiva baseada no modelo AUTOSAR
Fonte: SANTOS (2010)

Para algumas ECUs, não se faz necessária a utilização de sistema operacional para controle das atividades do hardware, sendo estas tarefas controladas a partir de eventos ou ativações de um relógio (timer).

A classificação das redes de comunicação automotivas podem levar em consideração diversos fatores, tais como funcionalidade, capacidade de transmissão, satisfação dos requisitos operacionais, facilidade de integração e manutenção. Dentro de um veículo podem existir várias tecnologias de redes de comunicação automotivas interligadas.

Um exemplo disto é ilustrado na FIGURA 12, que mostra uma arquitetura eletrônica utilizada pela Volvo Cars, onde se observa a presença de diversas tecnologias de redes de comunicação automotivas. Tal modelo carrega alguns benefícios e vantagens como, por exemplo, a proximidade das ECUs dos dispositivos com os quais as mesmas têm interface, com consequente redução do número de cabos, que por sua vez traz redução no peso do veículo.

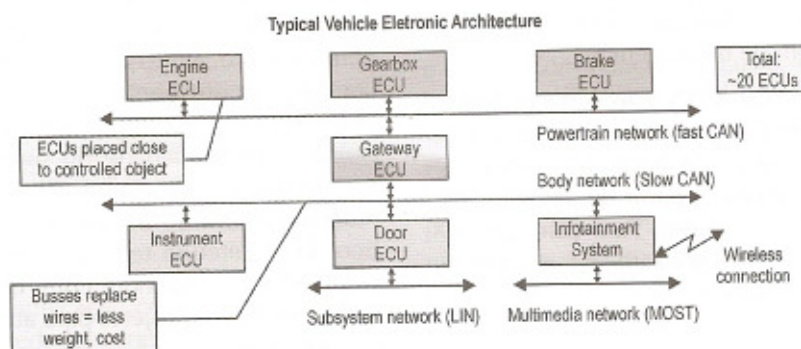


FIGURA 12: Arquitetura distribuída hierárquica para veículos de passeio da Volvo Cars
Fonte: SANTOS (2010)

O controle do powertrain e das funções críticas para a dinâmica do veículo é realizado pelas ECUs presentes na rede Fast CAN, com alta taxa de transmissão de dados. Já as ECUs que controlam as funções de conforto e entretenimento utilizam as tecnologias LIN (conforto) e MOST (entretenimento). Existe ainda a rede de comunicação Slow CAN, com taxa de transmissão mais baixa e que controla as funções de chassi do veículo.

De acordo com as tecnologias disponíveis e com os componentes e equipamentos padronizados para as redes de comunicação automotivas, a SAE (Society of Automotive Engineers) apresentou uma classificação para as tecnologias de redes automotivas. Tal classificação é determinada em função dos requisitos de comunicação das aplicações como taxa de transmissão, largura de banda, atraso, jitter, deadline, entre outros. Nesta classificação, as redes recebem as denominações Classe A, Classe B e Classe C. A seguir, apresenta-se uma breve definição de cada uma destas classes de rede, bem como algumas classificações extras (redes de entretenimento e redes sem fio). A FIGURA 13 mostra as diversas tecnologias de redes de comunicação automotivas em função do custo e da taxa de transmissão de dados.

Segundo MAHMUD (2005, p.3), a tendência atual para um veículo é possuir todos os tipos de rede que serão apresentadas a seguir, uma vez que a demanda por requisitos específicos para cada sistema aumenta cada vez mais. Assim, serão necessárias as redes de comunicação com tolerância a falhas e alta velocidade para as funções de segurança, encontradas nas redes automotivas Classe C. Também será praticamente mandatória a presença de tecnologias de rede com alta largura de banda para transmissões de conteúdo multimídia, o que será detalhado nas redes automotivas Classe Entretenimento. Ainda, as tecnologias de rede com menor taxa de transmissão continuarão presentes para manter um nível aceitável de custo para as funções de diagnóstico e conforto, conforme se verifica nas redes automotivas Classe A.

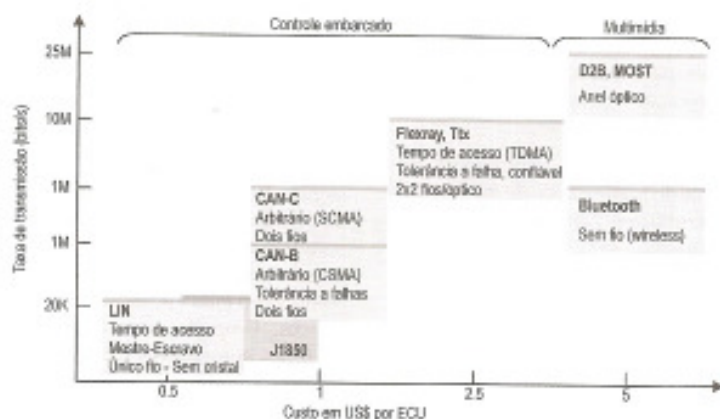


FIGURA 13: Classificação das redes automotivas quanto ao custo e taxas de transmissão
Fonte: SANTOS (2010)

3.2 Redes automotivas Classe A (funções de conforto e diagnóstico)

Em SANTOS (2010), as redes automotivas Classe A são definidas da seguinte forma:

São redes de comunicação com baixa largura de banda utilizadas em funções de conforto e diagnóstico (geralmente para essas funções a tendência é ter como acessar uma das redes padrão no veículo), como vidro elétrico, retrovisor, controle de bancos, lâmpadas etc. Alguns exemplos de redes automotivas Classe A são UART, I²C, BEAN, A-BUS, LIN, TTP/A e outras. (SANTOS, 2010, p.72)

A Tabela 1 mostra as principais características técnicas das redes automotivas Classe A, bem como a empresa ou conjunto de empresas responsáveis por cada tecnologia.

TABELA 1 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE A

Empresa	UART (ALDL)	SINEBUS	I ² C	SAE J1708	CCD	ACP	BEAN	LIN	TTP/A	A-BUS
	GM	DELCO	PHILIPS (NXP)	TMC-ATA	Chrysler	FORD	TOYOTA	Consórcio	TTTech	VW
Aplicação	Geral Diagnóstico	Áudio	Controle Diagnóstico	Controle Diagnóstico	Geral Diagnóstico	Áudio Controle	Carroceria Controle Diagnóstico	Carroceria Controle Diagnóstico	Carroceria Controle Diagnóstico	Carroceria Diagnóstico
Meio Físico	Simples Fio	Simples Fio	Par Trançado	Par Trançado	Simples Fio	Par Trançado	Simples Fio	Simples Fio	Simples Fio	-
Código Bit	NRZ	SAM	AM	NRZ	NRZ	NRZ	NRZ	NRZ	-	-
Controle Acesso Meio	Mestre / Escravo	Mestre / Escravo	Mestre / Escravo	Mestre / Escravo	Mestre / Escravo	Mestre / Escravo	Contenção	Mestre / Escravo	Mestre / Escravo	-
Controle Erro	8-bit CS	Nenhum	bit ACK	8-bit CS	8-bit CS	8-bit CS	8-bit CRC	8-bit CS	-	-
Cabeçalho	16 Bits	2 Bits	-	16 Bits	8 Bits	12-24 Bits	25 Bits	2 Bits/Byte	16 Bits	-
Dados	0-85 Bytes	10-18 Bits	-	-	5 Bytes	6-12 Bytes	1-11 Bytes	8 Bytes	16 Bytes	-
Overhead	Variável	75%	45%	Variável	16.7%	25%	28%	2 Bytes	39%	-
Taxa Transmissão	8192 bps	66.6KHz - 200KHz	1-100 kbps	9600 bps	7812.5 bps	9600 bps	10 kbps	20 kbps	50 kbps	-
Max. Compr. Barramento	Não Especificado	10 m	Não Especificado	Não Especificado	Não Especificado	40 m	Não Especificado	40 m	-	-
Max. Nº ECU	10	-	-	-	6	20	20	16	-	-

FONTE: Santos, 2010.

3.3 Redes automotivas Classe B (funções relacionadas à dinâmica do veículo)

Assim são definidas as redes automotivas Classe B em SANTOS (2010):

São redes utilizadas para aplicações importantes para a operação do automóvel e não demandam elevados requisitos de comunicação de dados. Geralmente são utilizadas para interconectar ECUs que gerenciam unidades como motor, transmissão, retarder, embreagem etc. Alguns exemplos de rede automotiva Classe B são VAN, J1850, J1939 e CAN. (SANTOS, 2010, p.74)

As principais características das redes automotivas Classe B estão descritas na Tabela 2.

TABELA 2 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE B

Empresa	SINGLE-WIRE CAN (SWC)	CAN 2.0 ISO 11898 ISO 11519-2 ISO 11992 J2284	J1850 ISO 11519-4			SAE J1939	VAN
	SAE/ISO	Bosch/SAE/ISO	GM	FORD	Chrysler	TMC-ATA	PSA/Renault
Aplicação	Diagnóstico	Controle Diagnóstico	Geral Diagnóstico	Geral Diagnóstico	Geral Diagnóstico	Diagnóstico	Controle Diagnóstico
Meio Físico	Simple Fio	Par Trançado	Simple Fio	Par Trançado	Simple Fio	Simple Fio	Par Trançado
Código Bit	NRZ-5 MSB first	NRZ-5 MSB first	VPW MSB first	PWM MSB first	VPW MSB first	NRZ-5 MSB first	NRZ-5 MSB first
Controle Acesso Meio	Contenção	Contenção	Contenção	Contenção	Contenção	Contenção	-
Controle Erro	CRC	CRC	CRC	CRC	CRC	CRC	-
Cabeçalho	11 Bits	11 ou 29 Bits	32 Bits	32 Bits	8 Bits	11 Bits	-
Dados	0-8 Bytes	0-8 Bytes	0-8 Bytes	0-8 Bytes	0-10 Bytes	0-8 Bytes	-
Overhead	9.9%	9.9% - 22%	33.3%	33.3%	8.3%	9.9%	-
Taxa Transmissão	33.3 kbps 83.33 kbps	10 kbps to 1 Mbps	10.4 kbps	41.6 kbps	10.4 kbps	33.33 kbps 83.33 kbps	-
Max. Compr. Barramento	30 m	Não Especificado 40 m (típico)	35 m (5m para scan tool)	35 m (5m para scan tool)	35 m (5m para scan tool)	30 m	-
Max. Nº ECU	16	Não Especificado 32 (típico)	32	32	32	16	-

FONTE: Santos, 2010.

3.4 Redes automotivas Classe C (funções de segurança crítica)

A definição das redes automotivas Classe C é assim por SANTOS (2010) colocada:

São redes utilizadas em aplicações de segurança crítica com requisitos de tempo real e tolerância a falhas, que estejam diretamente ligadas à dinâmica do automóvel e à segurança ativa. Aplicações baseadas na tecnologia x-by-wire requerem que as redes ofereçam transmissão de dados com baixo atraso de transmissão, alta frequência, tolerância à faltas e outros mecanismos essenciais para segurança crítica. (SANTOS, 2010, p.75)

A Tabela 3 descreve as principais características técnicas das redes automotivas Classe C.

TABELA 3 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE C

Empresa	TTP/C	FlexRay	TT-CAN	ByteFlight	BST	Safe-by-Wire	DSI
	TTTech	Consórcio	CiA	BMW	Siemens Bosch/Temic	Delphi/Philips TRW/Autoliv SDI	Motorola
Aplicação	Airbag	Segurança Crítica	Segurança Crítica	Segurança Crítica	Segurança Crítica	Segurança Crítica	Segurança Crítica
Meio Físico	Par Trançado	Par Trançado Fibra Óptica	Par Trançado	Fibra Óptica	Dois Fios	Dois Fios	Dois Fios
Código Bit	-	NRZ	NRZ	-	Manchester Bifásico	3 Níveis Tensão	3 Níveis Tensão
Controle Acesso Meio	TDMA	F-TDMA	TDMA	F-TDMA	-	-	-
Controle Erro	CRC 16 Bits	CRC 24 Bits	CRC 15 Bits	CRC 16 Bits	Paridade CRC	CRC 8 Bits	CRC 4 Bits
Dados	16 Bytes	0-246 Bytes	0-8 Bytes	-	1 Byte	1 Byte	1-2 Bytes
Overhead	-	-	32	-	-	64	16
Taxa Transmissão	5-25 Mbps	10 Mbps	1-2 Mbps	10 Mbps	31.25 kbps 125 kbps ou 250 kbps	150 kbps	5 kbps 150 kbps
Max. Compr. Barramento	-	25 m	40 m	-	-	25-40 m	-
Max. Nº ECU	64/256	-	32	-	12 squibs + 62 slaves	64	16

FONTE: Santos, 2010.

3.5 Redes automotivas Classe Entretenimento (funções de entretenimento)

Conforme descreve SANTOS (2010), segue a definição das redes Classe Entretenimento:

São redes utilizadas em aplicações de entretenimento automotivo como tecnologias de multimídia, telemetria, navegação, vídeo, e-books e outras. Alguns exemplos de tecnologias de redes automotivas para entretenimento são MOST, D2B e FireWire.
(SANTOS, 2010, p.76)

A FIGURA 14 mostra algumas tecnologias de sistemas de entretenimento automotivo.

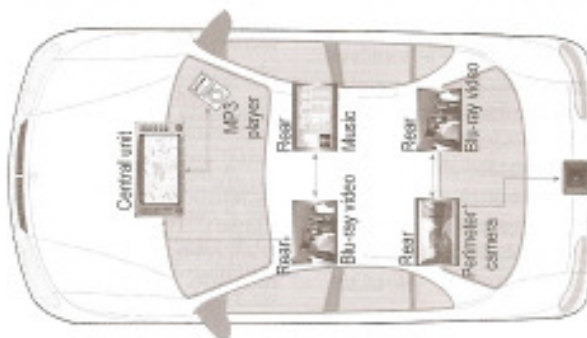


FIGURA 14: Sistema de entretenimento em um automóvel
Fonte: SANTOS (2010)

Na Tabela 4 são descritos alguns detalhes técnicos das tecnologias de redes automotivas Classe Entretenimento.

TABELA 4 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE ENTRETENIMENTO

Empresa	IDB-C	MOST	MML	USB	IEEE 1394	SYNC
	SAE	OASIS	DELCO	COMMERCIAL	IEEE	FORD Microsoft
Aplicação	Entretenimento	Entretenimento	Entretenimento	Periféricos	Periféricos	Entretenimento
Meio Físico	Dois Fios	Fibra Óptica	Fibra Óptica	Par Trançado Blindado	Par Trançado Blindado	-
Código Bit	NRZ	Bifásico	NRZ	NRZ	NRZ	-
Controle Erro	CRC 15 Bits	CRC	CRC	CRC	CRC	-
Dados	8 Bytes	-	1-200 Bytes	-	-	-
Taxa Transmissão	250 kbps	25 kbps	110 kbps	12 kbps	98-393 kbps	-
Max. Compr. Barramento	-	-	10 m	-	72 m	-
Max. Nº ECU	16	24	16	127	16	-

FONTE: Santos, 2010.

3.6 Redes automotivas Classe Wireless (funções de logística e roteirização)

As redes automotivas Classe Wireless são assim definidas por SANTOS (2010):

As funções de logística e roteirização estão sendo amplamente utilizadas e motivam a utilização de tecnologias de comunicação sem fio. Além destas há as tecnologias do tipo VANET (Vehicle Ad-Hoc Networks) que demandam comunicação veículo a veículo e veículo a estrada. Entre as tecnologias de comunicação sem fio, aplicadas ao domínio automotivo, têm-se o Bluetooth, Zigbee, UWB, Wi-Fi e outras.
(SANTOS, 2010, p.77)

A FIGURA 15 ilustra algumas tecnologias de rede sem fio, utilizadas no meio automotivo.

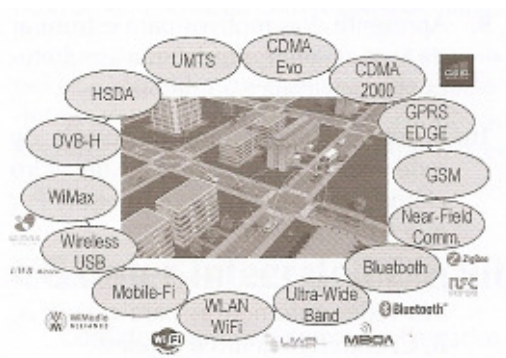


FIGURA 15: Tecnologias de redes sem fio (wireless) para sistemas de transporte
Fonte: SANTOS (2010)

Várias aplicações automotivas têm demandado a utilização de tecnologias de rede sem fio, levando à comunicação sem fio entre dispositivos dentro do veículo, bem como à troca de dados entre o veículo e seus arredores: outros veículos, estradas etc.

As tecnologias de rede sem fio podem ser classificadas da mesma forma que as redes cabeadas, no que diz respeito à sua topologia: PAN (Personal Area Network), WLAN (Wireless Local Area Network), WMAN (Wireless Metropolitan Area Network) e WWAN (Wireless Wide Area Network).

A Tabela 5 descreve as características das principais tecnologias de redes automotivas classe Wireless.

TABELA 5 – TECNOLOGIAS DE REDES AUTOMOTIVAS CLASSE WIRELESS

Padrão	Bluetooth	ZigBee	UWB	Wi-Fi
	IEEE 802.15.1	IEEE 802.15.4	IEEE 802.12.3a	IEEE 802.11a IEEE 802.11b IEEE 802.11g
Aplicação	Voz e Dados	Sensor Controle	Multimídia Biomédica	Escritório, Casa WLAN, substituindo LAN
Banda Frequência	2.4 GHz 2.5 GHz	2.4 GHz	3.1-10.6 GHz	2.4 GHz (b/g) 5 GHz (a)
Barramento	P2P	Mesh	P2P	P2P
Técnica Modulação	FHSS	DSSS	OFDM DS-UWB	OFDM DSSS
Taxa Transmissão	1 Mbps (versão 1.0) 2 Mbps (versão 2.0)	250 kbps	50-100 Mbps	54 Mbps (802.11a) 11 Mbps (802.11b) 54 Mbps (802.11g)
Max. Compr. Barramento	100 m	70 m	20 m	100 m

FONTE: Santos, 2010.

4 REDES AUTOMOTIVAS CLASSE A

Neste e nos próximos capítulos serão detalhadas as tecnologias de redes de comunicação automotivas para melhor entendimento do funcionamento e aplicação de cada uma delas.

As redes automotivas Classe A são utilizadas para funções que não demandam grande velocidade e quantidade de dados. Assim, vale-se de tecnologias com baixa taxa de transmissão (10kbps a 50kbps), baixo custo, frame de dados pequenos, interligação de dispositivos a curta distância com barramentos de um único fio, com requisitos simples de comunicação.

Em geral, tais redes são utilizadas para as funções de conforto e diagnóstico, tais como controle de iluminação, vidros, travas, espelhos etc. A seguir serão detalhadas as principais características de algumas redes de comunicação Classe A: I²C (Inter-Integrated Circuit), SPI (Serial Peripheral Interface), UART (Universal Asynchronous Receiver/Transmitter), TTP/A (Time-Trigger Protocol/Class A), LIN (Local InterConnect Network) e J1587/1708.

4.1 I²C (Inter-Integrated Circuit)

Considerando a presença de circuitos integrados (CI) nos diversos dispositivos de um sistema embarcado é, por vezes, necessário realizar a comunicação entre os diversos CIs presentes em uma mesma PCB (Printed Circuit Board).

Tais circuitos integrados não exigem protocolos de comunicação complexos para que possam se comunicar entre si, sendo um protocolo simples suficiente para tal comunicação.

O barramento I²C reflete-se como um bom exemplo deste tipo de tecnologia, com requisitos reduzidos de largura de banda, custo, energia e espaço. Assim, foi introduzido na comunicação entre dispositivos com maior poder de processamento e adotado em aplicações automotivas.

Desenvolvido pela Philips (atual NXP), tal protocolo foi idealizado para a comunicação entre CIs em que os dados são transmitidos serialmente, orientados a 8 bits nas duas direções. Existem três opções de velocidade de operação:

- 100 kbps: modo padrão (Standard Mode);
- 400 kbps: modo rápido (Fast Mode);
- 3.4 Mbps: modo de alta velocidade (High-speed Mode).

O modelo de comunicação é o mestre/escravo, com opção para multimestre. O mestre ainda pode operar como transmissor e receptor, tendo um endereçamento entre dispositivos de 7 a 10 bits únicos. O I²C possui duas linhas de transmissão, sendo uma para transmissão de dados e outra para transmissão de sinais de sincronização (clock).

Para facilitar a construção da PCB, o protocolo I²C requer uma quantidade mínima de terminais do micro controlador e, ainda, possui um controle de sincronização de dados que é composto pelo conceito de endereçamento único para cada CI e um protocolo para o barramento de controle. Tal protocolo determina a prioridade de acesso ao barramento, sincronizando a comunicação através da definição da fonte de clock controladora entre os CIs presentes no barramento.

A FIGURA 16 mostra os tipos de CIs que podem ser interconectados pela tecnologia de rede I²C.

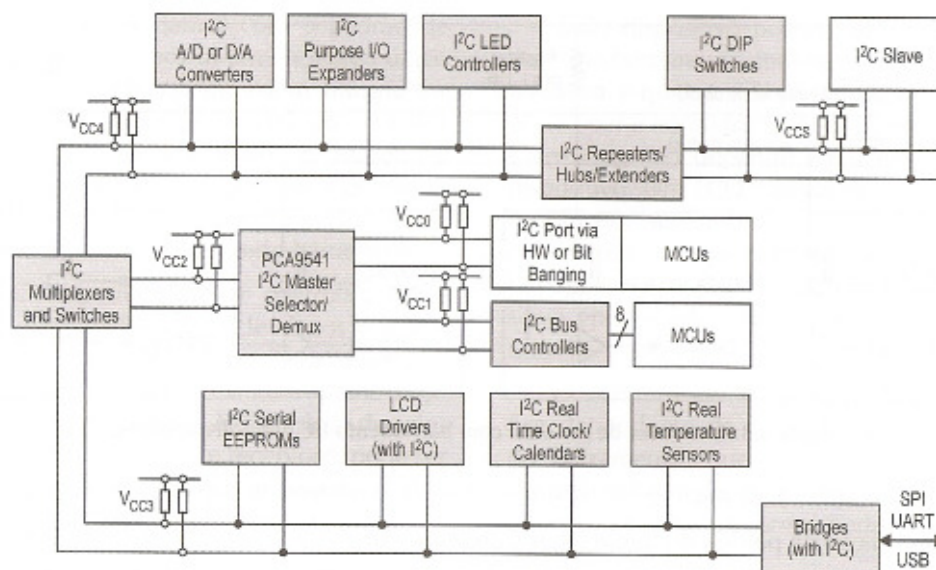


FIGURA 16: Dispositivos de CI interconectados via barramento I²C

Fonte: SANTOS (2010)

Conforme já mencionado, o protocolo I²C possui duas vias de comunicação: uma de dados (SDA – Serial Data Line) e outra de controle (SCL – Serial Clock Line). Ambas as linhas são bidirecionais e são conectadas à alimentação via um resistor pull-up, conforme pode ser visto na FIGURA 17.

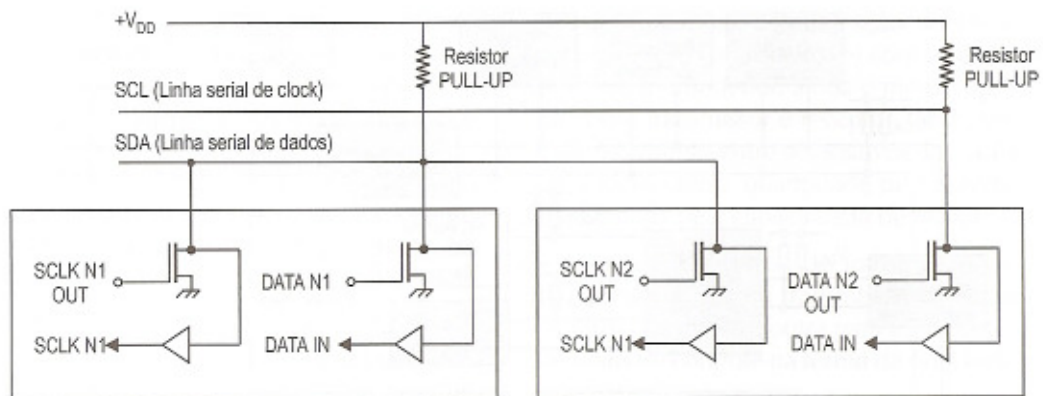


FIGURA 17: Interfaces de conexão com barramento I²C entre dispositivos
Fonte: SANTOS (2010)

A linha de dados SDA deve permanecer estável durante o período high (H ou 1) da linha de controle SCL. A mudança de nível lógico de high para low (L ou 0) ou de low para high na linha de dados só pode ser feita quando a linha de clock (SCL) estiver em low, conforme ilustra a FIGURA 18.

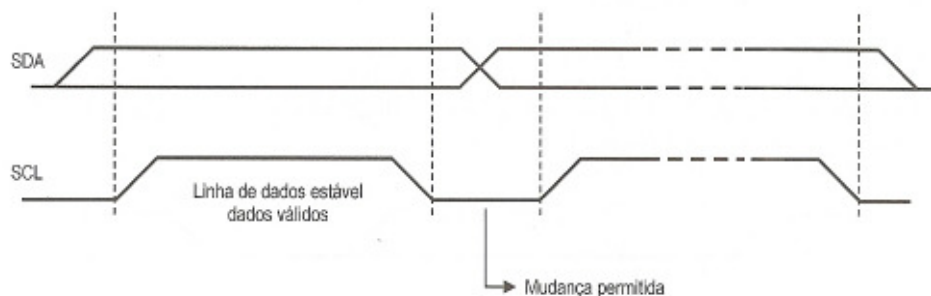


FIGURA 18: Diagrama de tempo durante a transmissão de um dado válido
Fonte: SANTOS (2010)

As condições para início e fim da transmissão de um frame de dados no protocolo I²C são as seguintes:

- Condição de partida do frame (Start Condition – S)
SDA de H para L: transição quando SCL = 1;

- Condição de parada do frame (Stop Condition – P)
SDA de L para H: transição quando SCL = 1.

Assim, os estados do barramento podem ser assim definidos:

- Ocupado (Busy): depois de S e antes de P;
- Livre (Free): depois de P e antes de S.

A FIGURA 19 ilustra tais condições de partida e parada da transmissão de um frame de dados. Tais condições são sempre determinadas pelo mestre e, após uma condição de partida, o barramento é considerado ocupado. Certo período de tempo após a condição de parada o mesmo é considerado livre.

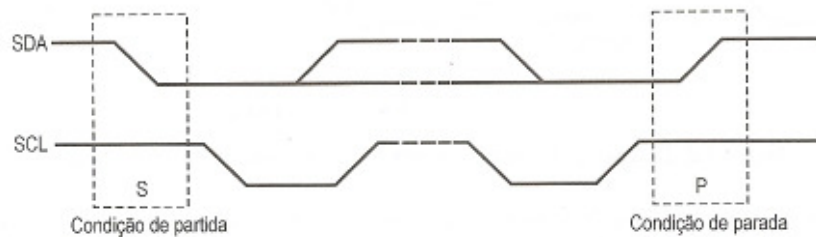


FIGURA 19: Diagrama de tempo para as condições de início e fim da transmissão de um frame
Fonte: SANTOS (2010)

Um CI dentro de um barramento I²C possui um endereçamento de 7 bits e pode operar como transmissor ou receptor, bem como mestre ou escravo. O mestre controla a linha de clock SCL para a transmissão de dados e também os endereços dos escravos. O escravo por sua vez obtém o endereço definido pelo mestre e a disposição dos transmissores e receptores pode-se dar da seguinte forma:

- Mestre ou escravo;
- Mestre-transmissor envia dados para o escravo-receptor;
- Mestre-receptor solicita dados do escravo-transmissor.

Por se tratar de um protocolo padronizado, a adição de um dispositivo escravo é simples. Qualquer mestre pode operar o barramento a uma velocidade máxima de 100 kbps, porém, se mais de um mestre tentar simultaneamente obter o controle do barramento, prevalecerá o clock do mestre que já estava em atividade.

Cada dispositivo possui um endereço único no barramento e todos eles podem operar como transmissor e receptor. É natural que tal característica também está ligada à natureza do dispositivo (por exemplo, um LCD geralmente será um receptor, enquanto um teclado geralmente será um transmissor).

A FIGURA 20 mostra a transferência de dados no protocolo I²C e SANTOS (2010) coloca a explicação da mesma da seguinte forma:

A transferência de dados realiza-se com o mestre definindo $SCL = 1$ e gerando um pulso a cada bit de dados. A cada 8 bits (1byte) de dados tem-se um pulso de reconhecimento (ACK) em que o número de bytes que podem ser transferidos é ilimitado. Se o CI receptor (escravo) não for capaz de receber outro byte completo de dados até que alguma função seja executada, como, por exemplo, uma interrupção interna, o escravo deve levar a linha de clock ao nível L, forçando o mestre a entrar em um modo de espera. Os dados serão lidos novamente, quando o escravo liberar a linha de clock. A transferência de dados com o reconhecimento é obrigatória e está relacionada ao pulso de clock gerado pelo mestre. O mestre libera a linha DAS (nível H) durante a ocorrência dos pulsos de clock e o dispositivo CI receptor (escravo) leva a linha DAS ao nível L durante o período H do pulso de reconhecimento.

(SANTOS, 2010, p.84)

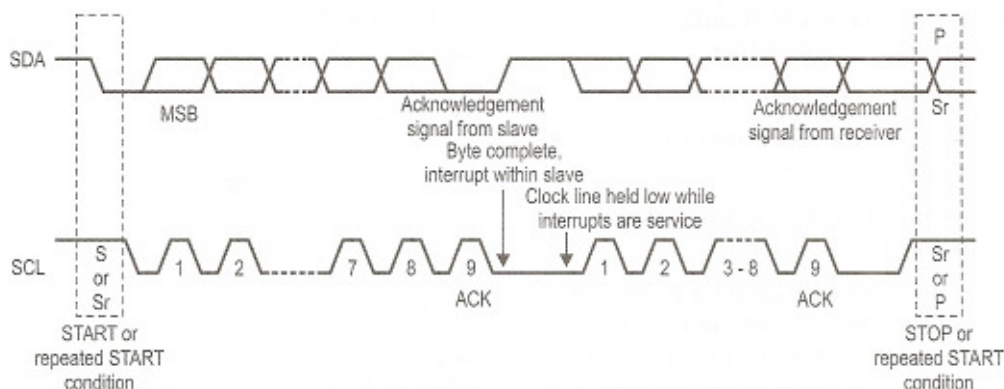


FIGURA 20: Diagrama de tempo para transmissão de dados
Fonte: SANTOS (2010)

O escravo endereçado deve sempre gerar um reconhecimento após cada byte recebido. Quando o escravo não reconhece a recepção, como por exemplo, devido a uma impossibilidade de processamento da recepção gerada pela execução de alguma função em tempo real, o mesmo deve deixar a linha de dados em nível H. Assim, o mestre gerará uma condição de parada e abortará a transferência.

Se o escravo reconhecer a recepção dos dados, mas depois de algum tempo não receber mais nenhum dado, o mestre deve abortar a transferência. Tal condição é indicada pelo escravo com a não geração do reconhecimento após a recepção do primeiro byte de dados. Neste caso, o escravo deixa a linha de dados em H e o mestre gera a condição de parada.

Outro ponto importante do protocolo I²C é a sincronização da linha SCL. Cada mestre gera seu próprio clock na linha SCL. Assim, é necessário definir um clock para que se tenha um procedimento de decisão de bit a bit na linha SCL. Tal sincronismo se dá por uma conexão AND entre os clocks de todos os CIs e a linha SCL.

A FIGURA 21 ilustra este processo. Uma transição de H para L na linha SCL força o CI a iniciar a contagem de seu período em L de clock. A linha permanece neste estado até que o CI passe o clock novamente para H. Porém, se outro CI ainda estiver no período L, não haverá mudança de estado na linha SCL. Assim, conclui-se que a linha SCL será mantida em L pelo CI cujo clock possuir o maior período em L. Para os outros CIs cujos períodos em L são menores, haverá um estado de espera H durante este tempo.

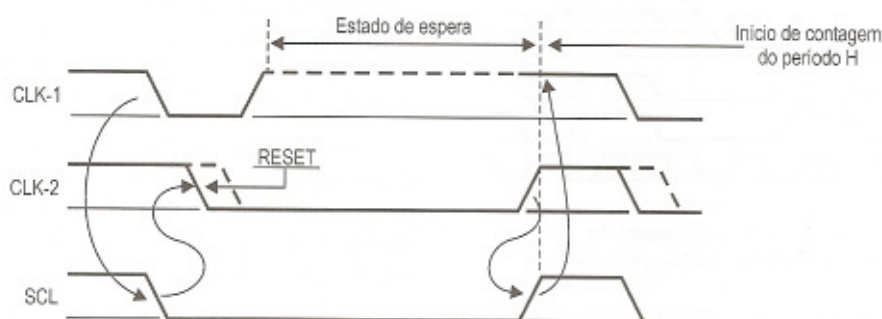


FIGURA 21: Sincronização de clock do SCL
Fonte: SANTOS (2010)

Quando todos os CIs concluírem a contagem do período em L de seus respectivos clocks, a linha SCL será levada para H e, assim, liberada. Neste momento todos os clocks dos CIs, assim como a linha SCL, estarão em H, não havendo diferença entre eles. Então, é neste instante que os CIs iniciam a contagem do período H de seus clocks e o primeiro que completar esta contagem, levará a linha SCL para L, e todos os CIs terão seus clocks sincronizados.

Conforme indicado na FIGURA 22, o formato de frames no protocolo I²C é constituído da parte de endereçamento, reconhecimento, operação de leitura/escrita e dados. Após a condição de partida, o endereço do escravo é fornecido (7 bits), o oitavo bit é o bit de reconhecimento e o nono bit constitui um bit de direção (W/R – WRITE/READ), onde o nível L indica uma transmissão (WRITE) e o nível H indica uma requisição de dados (READ). Após estes bits, os bits de dados completam o frame, até que uma condição de parada seja gerada pelo mestre.

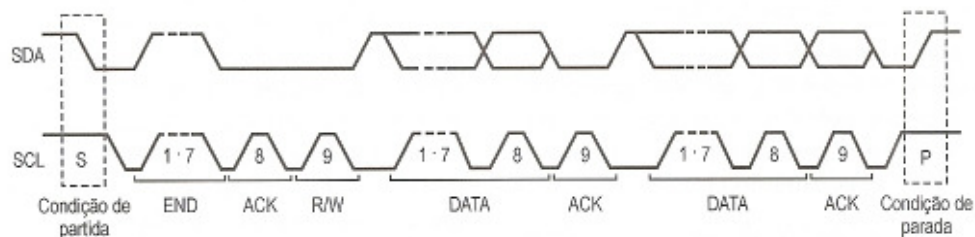


FIGURA 22: Arbitragem de transmissão de dados pelo DAS
Fonte: SANTOS (2010)

Os períodos mínimos de clock L e H no barramento I²C são, respectivamente, 4,7 ms e 4 ms. Os mestres podem gerar um clock de até 100 kHz e os dispositivos do barramento devem ser capazes de transmitir/receber nesta frequência ou responder ao procedimento de sincronização do clock, forçando o mestre a entrar em um estado de espera e aumentar o período L do clock. Com isso, é natural que a frequência seja reduzida.

4.2 SPI (Serial Peripheral Interface)

A tecnologia SPI (Serial Peripheral Interface) foi desenvolvida pela Motorola (Freescale) e consiste em um barramento serial, síncrono, full-duplex e composto pelos sinais de dados MOSI (Master data Output, Slave data Input) e MISO (Master data Input, Slave data Output) e pelos sinais de controle SCLK (Serial Clock) e SS (Slave Select). A FIGURA 23 ilustra os sinais de interface entre mestre e escravo no protocolo SPI.

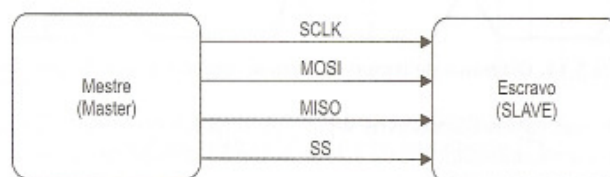


FIGURA 23: Sinais de interface entre master e slave no SPI
Fonte: SANTOS (2010)

A comunicação entre mestre e vários escravos na tecnologia SPI é dada por um dispositivo mestre capaz de selecionar diversos escravos através de várias linhas SS, onde é possível a geração de diversos sinais separados de Slave Select, conforme mostra a FIGURA 24.

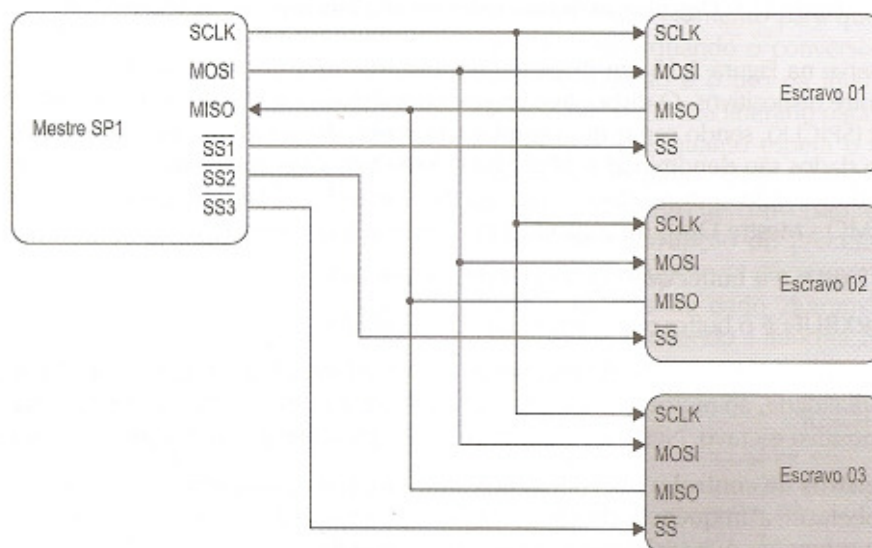


FIGURA 24: Sinais de interface entre dispositivos mestre e escravo com barramento SPI
Fonte: SANTOS (2010)

Este protocolo é indicado para aplicações que transferem um fluxo contínuo de dados, uma vez que não possui um mecanismo de confirmação de dados recebidos. Sua taxa de transmissão pode chegar a até 1 Mbps.

O fluxo de sinais para a troca de dados entre dispositivos SPI é mostrado na FIGURA 25. Nela, observam-se os seguintes dispositivos:

- SPITXBU: é o buffer que contém o dado transmitido;
- SPIRXBUF: é o buffer que contém o dado recebido;
- SPIDATA: é o registro de dados a serem transmitidos e transferidos para este registro. Na realidade, ao mesmo tempo em que o mestre transmite dados, também recebe informação do escravo;
- Registros de controle: são os registros utilizados para configuração do barramento SPI, como determinar a frequência do clock, definir se o dispositivo será mestre ou escravo etc.

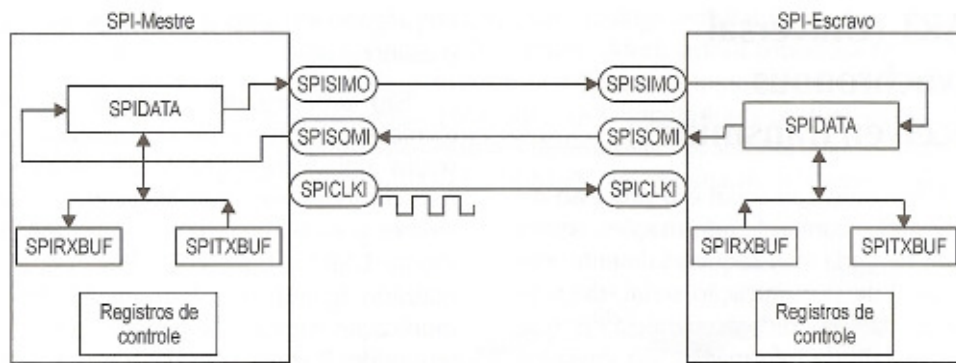


FIGURA 25: Fluxo de controle para comunicação entre dispositivos mestre e escravo com barramento SPI
Fonte: SANTOS (2010)

O mestre inicia uma transmissão enviando o sinal SPICLK. Os dados seriais são enviados pelo mestre através do pino SPISIMO e pelo escravo através do pino SPISOMI. Na borda de subida do clock os dados são armazenados nestes pinos e na descida subsequente do clock os mesmos são armazenados no registro SPIDATA. Assim, a cada descida do clock, o mestre captura um bit vindo do escravo e o escravo captura um bit vindo do mestre, caracterizando uma transmissão/recepção de dados simultânea entre mestre e escravo.

4.3 UART (Universal Asynchronous Receiver/Trasnmitter)

O UART é caracterizado por uma comunicação serial assíncrona do ponto de vista de hardware e é utilizado para comunicação entre dispositivos com distâncias maiores do que aquelas onde uma comunicação síncrona (USRT) é utilizada. O UART é utilizado quando se deseja um barramento de comunicação barato e de baixa velocidade, pois seu hardware é relativamente simples, necessitando apenas de um único fio para cada direção.

Alguns exemplos de comunicação serial assíncrona podem ser vistos nas comunicações entre PCs, periféricos e mainframes. Por exemplo, a comunicação entre dois PCs distantes utiliza-se do modem (Modulador/Demodulador) para se beneficiar da estrutura de comunicação analógica/digital da linha telefônica. Assim, o modem é utilizado para a transmissão de dados através da conversão analógica (linha telefônica) em digital (computador) e vice-versa. A FIGURA 26 ilustra esta aplicação.

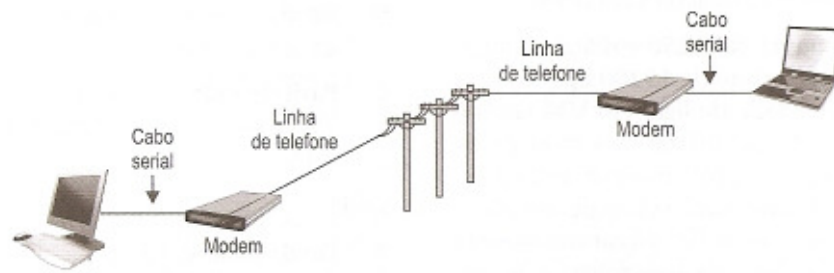


FIGURA 26: Comunicação serial assíncrona IBM-PC via sistema de telefonia
Fonte: SANTOS (2010)

Outro exemplo é a comunicação entre um mainframe e vários terminais em uma arquitetura estrela, onde os terminais utilizam a tecnologia de comunicação UART, conforme ilustra a FIGURA 27.

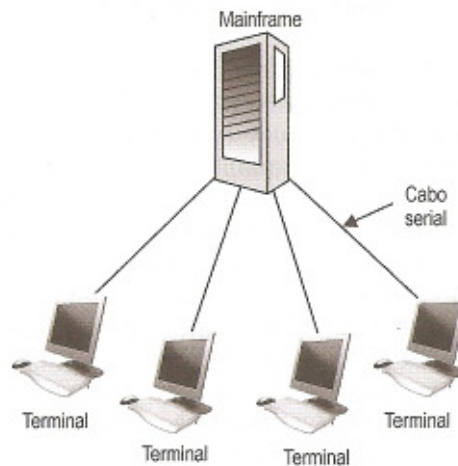


FIGURA 27: Comunicação serial assíncrona mainframe-terminais
Fonte: SANTOS (2010)

O UART deve ser configurado antes que os dados sejam transmitidos ou recebidos, sendo a limitação da velocidade de transmissão dada pela capacidade de recepção do dispositivo com o qual se deseja comunicar. O UART pode chegar a uma velocidade de transmissão/recepção de até 115,2 kbps. A FIGURA 28 mostra as características de um controlador UART, com as interfaces definidas da seguinte forma:

- Din [6:0]: 7 bits de dados de entrada paralelos;
- Send: transmite instruções para começar a transmissão;
- ParitySelect: seleciona paridade par (ParitySelect = 0) ou ímpar (ParitySelect = 1);
- Dout: saída serial;
- Busy: diz que o host está ocupado, enviando um caractere.

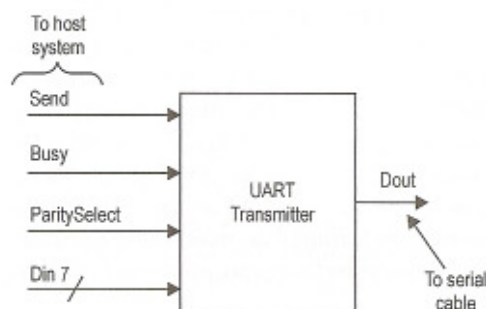


FIGURA 28: Controlador UART
Fonte: SANTOS (2010)

Quando o sistema deseja transmitir informações, ele habilita o sinal Send indicando para o transmissor UART que deseja enviar dados pelo barramento serial. Em seguida, o controlador UART habilita o sinal Busy mostrando ao sistema que é possível enviar o byte desejado e, assim, a transmissão se inicia. Quando a transmissão termina, então, o controlador UART desabilita o sinal Busy e o sistema desabilita o sinal Send. Esta sequência dos sinais de controle pode ser vista na FIGURA 29.

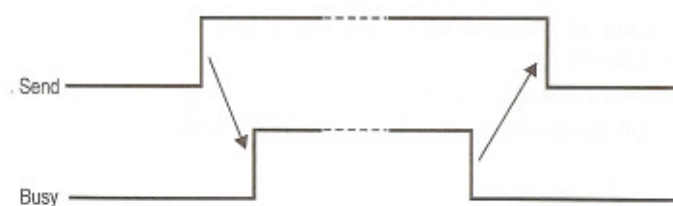


FIGURA 29: Sequência de sinais de controle para transmissão no UART
Fonte: SANTOS (2010)

O tempo de duração de cada bit é fixo, determinado pela taxa de transmissão. Se tomarmos como exemplo uma taxa de transmissão de 1200 bps, então o tempo de cada bit será de 833,3 μ s ($1/1200$ s).

A transmissão de dados através do UART inicia-se com o start bit que marca o início de um frame com oito bits. Assim, o receptor sincroniza o seu controlador de forma que seja capaz de receber a sequência de dados do transmissor.

Se forem transmitidos 7 bits de dados, o oitavo bit pode ser utilizado para indicar paridade (ímpar ou par). No caso de se utilizar paridade par, os bits de dados mais o bit de paridade formarão um número par. No caso de paridade ímpar formarão um número ímpar. Tal bit de

paridade é utilizado como detecção de erro, pois caso um dos bits de dados seja perdido, o bit de paridade não indicará a paridade correta.

O princípio do UART revela sua simplicidade operacional, onde pode ser aplicado quando se necessita de uma comunicação entre dois dispositivos ponto a ponto.

4.4 LIN (Local Interconnect Network)

O protocolo LIN surgiu da necessidade de se ter uma solução de comunicação de dados a baixo custo para algumas aplicações automotivas, tais como o controle de faróis, vidros elétricos, porta, ar-condicionado, entre outros.

Assim, inicialmente foi formado um consórcio entre cinco montadoras europeias (Audi, BMW, Daimler-Chrysler, Volkswagen e Volvo Corporation), um fornecedor de tecnologia de semicondutores (Motorola) e a empresa de desenvolvimento de software Volcano Communications Technologies. A FIGURA 30 mostra o consórcio entre as empresas responsáveis pela criação do protocolo LIN.



FIGURA 30: Grupo de consórcio inicial do LIN
Fonte: SANTOS (2010)

De acordo com a especificação do LIN e padronização definida pela SAE, ele é considerado uma sub-rede Classe A com comunicação de baixa largura de banda utilizada em aplicações não críticas em automóveis.

O conceito da tecnologia LIN baseia-se no baixo custo, escalabilidade, atuação como extensão lógica de uma rede CAN, baixo tempo de resposta (máximo de 100 ms) e previsibilidade temporal no pior caso.

Um exemplo da utilização do LIN pode ser visto na FIGURA 31, onde as funções de controle de porta são colocadas na forma de uma arquitetura eletroeletrônica distribuída com uma rede LIN interconectada.

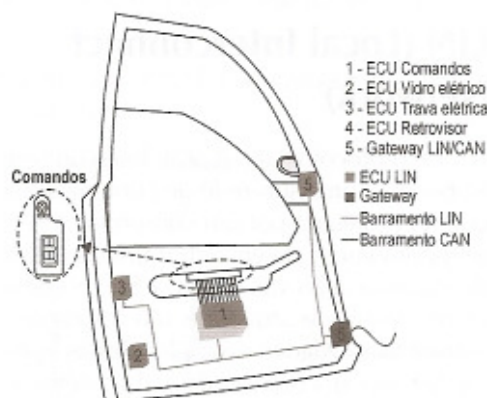


FIGURA 31: Arquitetura distribuída com tecnologia LIN para funções de controle na porta
Fonte: SANTOS (2010)

Outro exemplo, mostrado na FIGURA 32, é a mudança de um único controlador concentrando várias funções para vários dispositivos microprocessados que permitem uma interconexão através da rede LIN.

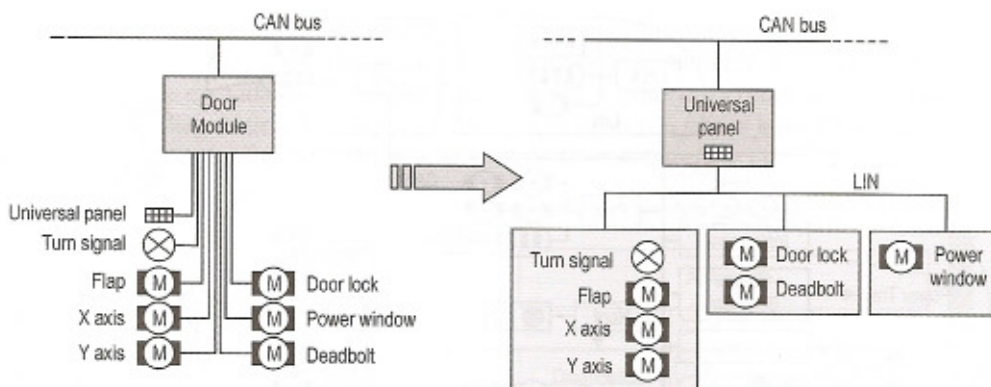


FIGURA 32: Estrutura funcional do consórcio LIN
Fonte: SANTOS (2010)

As principais características técnicas da rede LIN são:

- Comunicação mestre/escravo, no qual para um único nó mestre é possível ter 15 nós escravos.
- Utilização de interface UART/SCI presente na maioria dos microcontroladores.
- Controlador LIN implementado em CI que interconecta com as UART/SCI e executa uma máquina de estados simples.
- Autossincronização sem cristal nos nós escravos, sendo apenas o mestre a possuir clock de sincronização.
- Sinal de transmissão determinístico, com sinalização do tempo de propagação calculado antecipadamente, ou seja, com tempo de atraso garantido para a transmissão (previsibilidade).
- Implementação com apenas um fio (baixo custo).
- Sinal baseado em interação de aplicações.
- Máxima velocidade de transmissão de 20 kbps.
- Mensagens e broadcast e multicast.
- Barramento limitado a 40 metros.
- Mensagens de 2, 4 ou 8 bytes de dados, com 3 bytes de controle.
- Detecção de erros com 8 bits de checksum e 2 bits de paridade no identificador.
- Camada física com padrão ISO 9141.

Assim, é possível identificar algumas vantagens da rede LIN, tais como baixo custo, redução de cabos, possibilidade de expansão do número de nós etc.

A FIGURA 33 mostra uma arquitetura que utiliza os protocolos CAN e LIN, com as funções comunicando-se entre si de forma clara e através de dispositivos gateway para garantir a comunicação de dados entre todas as ECUs do veículo.

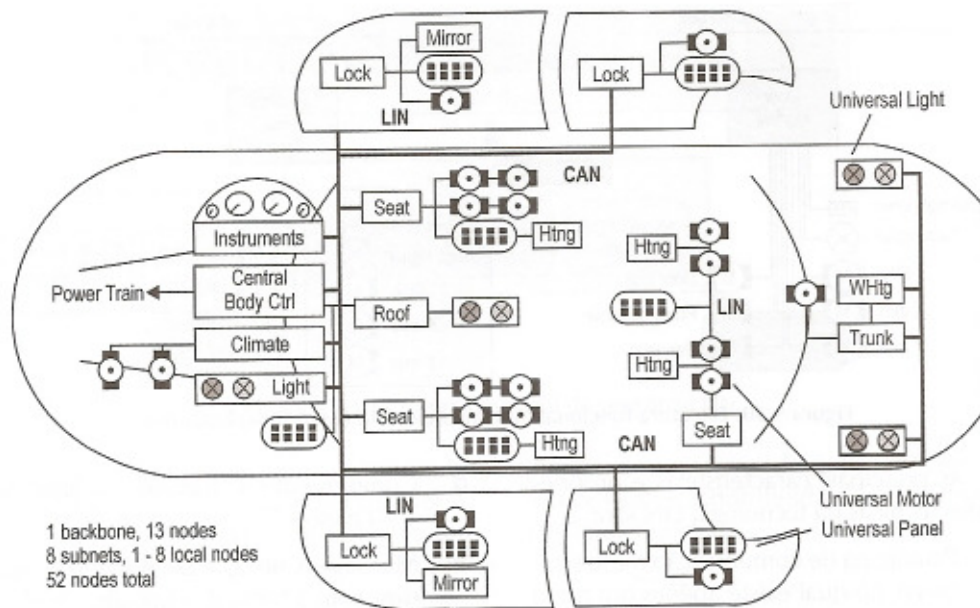


FIGURA 33: Arquitetura automotiva com tecnologias CAN e LIN
Fonte: SANTOS (2010)

O LIN é fundamentado no modelo de referência OSI já citado anteriormente, contemplando as camadas principais do modelo. A FIGURA 34 mostra os detalhes da especificação de hardware bem como os níveis de tensão para a interface de comunicação LIN.

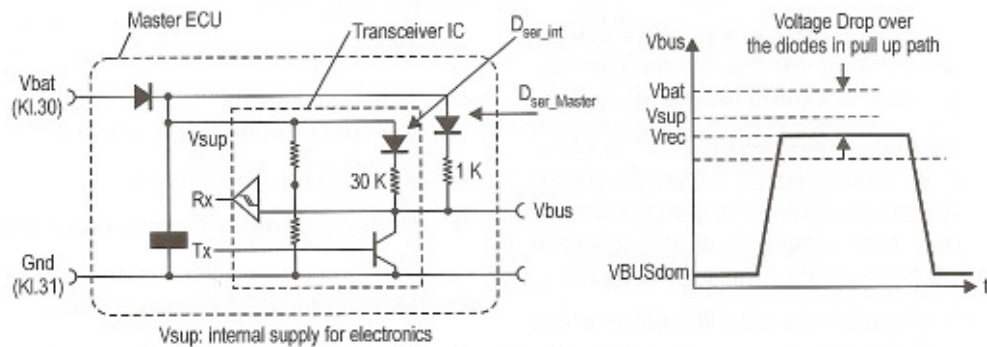


FIGURA 34: Detalhes do transceiver LIN 2.0
Fonte: SANTOS (2010)

O princípio de funcionamento do barramento LIN baseia-se em dois níveis lógicos: nível lógico baixo (0) dominante e nível lógico alto (1) recessivo. Os limites de reconhecimento dos níveis em função da tensão de alimentação (V_{SUP}) da ECU são ilustrados na FIGURA 35.

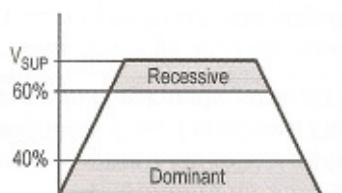


FIGURA 35: Determinação do nível lógico sob barramento LIN
Fonte: SANTOS (2010)

O LIN consiste em uma tarefa mestre e várias tarefas escravas. A tarefa mestre é responsável por determinar a prioridade de cada mensagem, monitorar os dados, servir como referência juntamente com a base do clock e receber os comando de WakeUp e Break dos nós escravos quando a rede estiver operando em modo standby.

A tarefa escrava é composta de dois a 15 ECUs no barramento e consiste em receber ou transmitir dados quando solicitado pelo mestre através do envio de um identificador (ID). A FIGURA 36 mostra a estrutura de um barramento LIN.

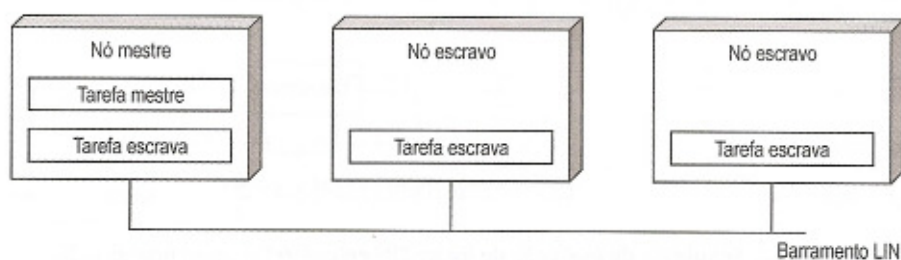


FIGURA 36: Estrutura de um barramento LIN com componentes mestre e escravo
Fonte: SANTOS (2010)

O mestre contempla a tarefa mestre, tarefa escrava, o controle global do barramento, o gerenciamento da velocidade de transmissão, o instante em que cada mensagem deve ser transferida e a monitoração de dados do barramento. Para isso, utiliza alguns sinais de controle e ações:

- Envia Sync Break;
- Envia Sync Byte;
- Envia ID-Field;
- Monitora bytes de dados e de check e verifica consistência de dados;
- Recebe o WakeUp e o Break dos nós escravos quando o barramento está inativo e solicita alguma ação.

A ECU escrava tem a capacidade de receber ou transmitir dados quando solicitada pelo mestre através de um ID. Tais IDs indicam a prioridade e os nós escravos se organizam através deles. Assim que cada nó escravo recebe seu ID, eles enviam ou recebem dados de 1, 2, 4 ou 8 bytes e, em seguida, enviam o byte de verificação (check).

Um frame LIN consiste em um cabeçalho (header) fornecido pela tarefa mestre e uma resposta fornecida pela tarefa escrava. O cabeçalho consiste em um teste padrão do Break e do Sync, seguidos por um identificador, que, por sua vez, define a finalidade do frame.

A tarefa escrava é responsável por fornecer a resposta associada àquele ID transmitido pelo mestre. Tal resposta é formada por um campo de dados e um campo de checksum. Este modelo é ilustrado na FIGURA 37.



FIGURA 37: Sequência de execução de frame LIN entre tarefas para comunicação
Fonte: SANTOS (2010)

Conforme já citado, um frame é formado por um cabeçalho e uma resposta. No cabeçalho é fornecida a informação de break e de synch e na resposta utilizam-se quatro a 11 campos de bytes. O Least Significant Bit (LSB) do dado é enviado primeiro e depois é transmitido o Most Significant Bit (MSB). A identificação de início de byte é dada por um bit de início com o valor 0 (dominante) e o final é identificado por um bit de parada de valor 1 (recessivo), conforme apresentado na FIGURA 38.

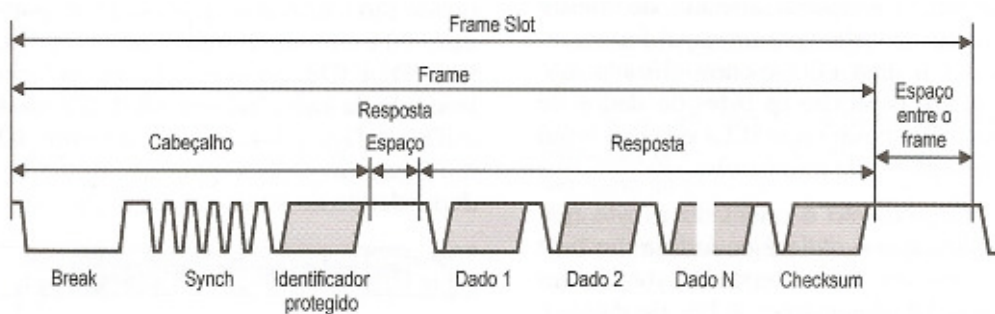


FIGURA 38: Estrutura de um frame LIN
Fonte: SANTOS (2010)

O último campo de um frame é o checksum, o qual é transmitido em um campo de byte e traz a soma invertida de 8 bits com carga de todos os bytes de dados.

A tecnologia de rede LIN vem sendo amplamente utilizada pelas montadoras devido às suas características técnicas serem superiores a outras tecnologias. O poder de flexibilização, onde nós adicionais podem ser colocados no barramento, ajuda na escalabilidade dos projetos das montadoras. Outro fator que fortalece a utilização do protocolo LIN, segundo NAVET (2009), é a possibilidade do envio de mensagens aos nós mesmo com os mesmos estando em modo de baixo consumo de energia (sleep mode), o que é extremamente conveniente para uma aplicação automotiva.

4.5 TTP/A (Time-Triggered Protocol/Class A)

Em SANTOS (2010), assim é definido o protocolo TTP/A:

O TTP/A (Time-Triggered Protocol/Class A) é um protocolo de comunicação do tipo time-triggered (gatilhado no tempo) direcionado para ser utilizado em conexões de baixo custo, entre ECUs com transdutor inteligente. O time-triggered consiste num mecanismo de acesso ao barramento, em que a arbitragem é realizada a partir de tabelas de tempo bem definidas.

(SANTOS, 2010, p.105)

Neste protocolo, as ECUs comunicam-se somente através de slots time predefinidos, sendo as origens das mensagens determinadas pelos instantes em que são enviadas, retirando a necessidade de endereçamento dentro do protocolo.

No seu conceito, um mestre controla o acesso ao barramento através de uma base de tempo comum entre os escravos. É realizada uma sincronização dos clocks das ECUs escravo com o clock da ECU mestre, alcançando-se assim um tempo global bem definido, o que minimiza o efeito jitter (variação do atraso) na transmissão das mensagens.

Sua gama de aplicações do ponto de vista de microcontroladores é bem vasta, podem ser utilizados em tecnologias de 8 bits até microcontroladores de 32 bits.

Conforme já mencionado, este protocolo é utilizado na conexão entre ECUs com transdutores inteligentes, cujas interfaces podem ser assim classificadas:

- Interface de Serviços em Tempo Real (RS): fornece os serviços de transporte em tempo real para os componentes durante a operação do sistema;
- Interface de Diagnóstico e Manutenção (DM): define os parâmetros de calibração e recuperação de informações sobre o funcionamento interno de um determinado componente;
- Interface de Gerenciamento (MS): está disponível durante a operação do sistema sem perturbar os serviços de tempo real do sistema.

A largura de banda para uma aplicação TTP/A é definida antes de sua implementação, sendo definida claramente a capacidade de transmissão de dados, porém limitando a flexibilidade da aplicação.

As ECUs escravos dentro do TTP/A geralmente são aquelas de mais baixo custo, as quais terão interface direta com as grandezas físicas do veículo através do controle de sensores e atuadores. Já a ECU mestre possui um maior poder de processamento, fazendo todo o gerenciamento do sistema, além de proporcionar a interface entre o protocolo TTP/A e outros protocolos de nível superior.

O protocolo TTP/A é bem flexível do ponto de vista do meio físico, podendo-se utilizar apenas um fio único até uma fibra óptica com alta capacidade de transmissão.

A FIGURA 39 mostra a forma como a comunicação é organizada no TTP/A, que é em ciclos (rounds). Nela, existem quatro rounds subsequentes na forma de MP-Round (Multi-Partner Round) separados através de um IRG (Inter Round Gap).

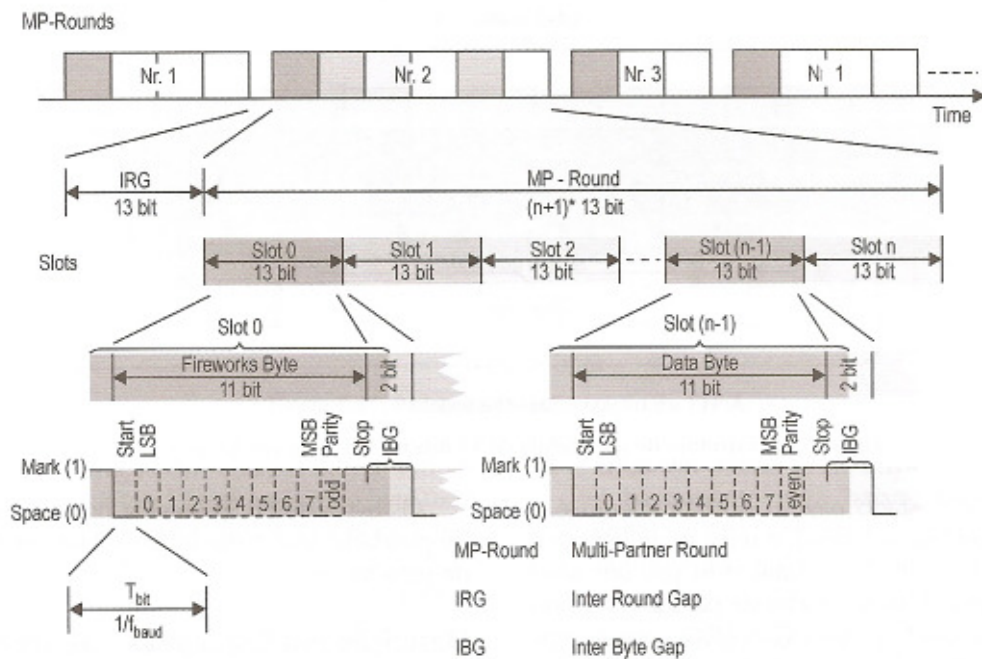


FIGURA 39: O round e quadros do protocolo TTP/A
Fonte: SANTOS (2010)

A transmissão de um byte no TTP/A é composta de 13 bits no padrão UART, sendo um start-bit, oito bits de dados, um bit de paridade, um stop-bit e dois bits que formam o IGB (Inter-Byte Gap). A FIGURA 40 ilustra esta estrutura.

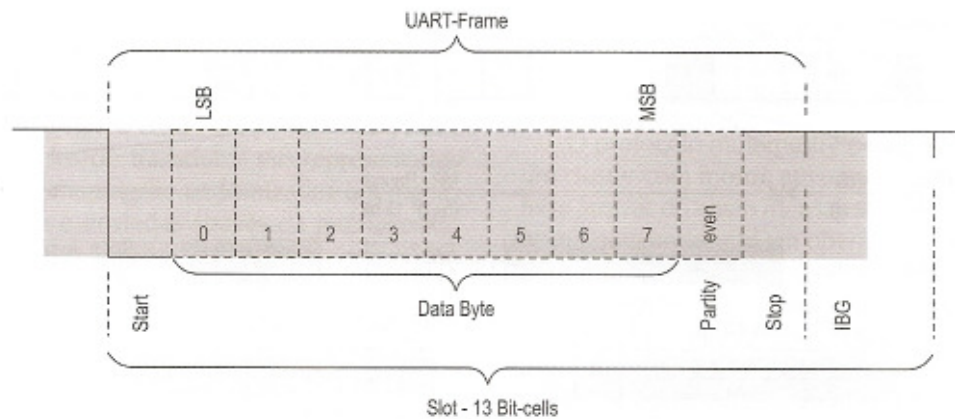


FIGURA 40: Estrutura de uma célula de 13 bits do TTP/A no protocolo UART
Fonte: SANTOS (2010)

O barramento é considerado passivo quando está em nível lógico 1 (alto) e um nível lógico 0 (baixo) inicia um novo byte de dados. O bit de parada é seguido por um Inter-Byte Gap de dois bits, conforme já mencionado. Assim, têm-se o padrão UART de 11 bits mais os dois bits do IGB, totalizando a célula de 13 bits do TTP/A.

4.6 J1587/J1708

O protocolo J1587/J1708 foi padronizado pela SAE com o intuito de se garantir uma padronização nas ECUs e na consistência de dados de software para veículos pesados e semipesados. Tal protocolo consiste em um padrão de baixo custo, sendo uma união de um protocolo de camada de aplicação (J1587) utilizado em conjunto com o padrão J1708, o qual descreve as características físicas do hardware de comunicação.

As principais aplicações deste protocolo estão ligadas às informações do veículo (tais como desempenho, manutenção e diagnóstico), navegação/horários (descrição do percurso e estimativa do tempo), informações ao motorista sobre os detalhes da viagem e informações de carga com dados sobre o lugar e a rota de entrega.

O comprimento de uma mensagem é limitado a 21 bytes, sendo formada pelo identificador de mensagem (MID), identificador de parâmetro (PID), bytes de dados e uma soma de verificação (checksum).

Alguns detalhes da especificação da estrutura de mensagem do J1587 são os seguintes:

- O MID (Message Identification) identifica a ECU. O J1587 define o intervalo de 128-255 para os MIDs;
- O PID (Parameter Identification) tem um byte de comprimento, podendo, assim, ter valores entre 0 e 255. Uma mensagem pode conter mais de um PID;
- Cada PID é seguido por um número de bytes de dados deste parâmetro. A interpretação deste dado depende do PID ao qual ele está relacionado;
- O checksum fecha a mensagem do protocolo J1587/J1708.

A FIGURA 41 mostra a estrutura de mensagem do J1587.

MID	PID	Data	PID	Data 1	Data 2	Checksum
128	21	50	12	05	48	248

FIGURA 41: Uma mensagem J1587 contém dois PIDs, 21 e 12
Fonte: SANTOS (2010)

A prioridade e a taxa de transmissão de uma mensagem J1587 são determinadas pelo fabricante do dispositivo. Porém, o padrão J1587 tem recomendações a respeito destas características para evitar sobrecarga no barramento. Quando vários parâmetros são enviados em uma só mensagem, a prioridade é definida com base no parâmetro de maior prioridade. Para uma solicitação de diagnóstico, tal tipo de mensagem vai receber uma prioridade baixa para evitar o tráfego no barramento e o comprometimento no envio de mensagens mais prioritárias.

5 REDES AUTOMOTIVAS CLASSE B

As redes automotivas Classe B são redes com taxas de transmissão na ordem de 250 kbps, 500 kbps ou superior, fornecendo um tempo de resposta semelhante ao de redes de alta velocidade, devido aos frames otimizados que possuem.

Em geral, são compostas por pares trançados de fios de cobre, sendo mais toleráveis a ruídos eletromagnéticos. As funções que este tipo de classe suporta dentro de um veículo são gerenciamento do motor, embreagem, transmissão, dashboard, diagnóstico, entre outras.

A seguir são apresentadas as principais características das tecnologias disponíveis das redes automotivas Classe B, tais como CAN (Controller Area Network), SAE J1939, CCP (CAN Calibration Protocol), VAN (Vehicle Area Network), K-Line (ISO 9141), KWP2000 e SAE J1850.

5.1 CAN (Controller Area Network)

O protocolo CAN foi desenvolvido por Robert Bosch na década de 1980 com o intuito de conectar as diversas ECUs do veículo, dispondo-as de forma a ficarem mais próximas dos componentes que controlam e consequentemente reduzindo a quantidade de cabos necessários.

Trata-se de um protocolo de comunicação serial síncrono e é padronizado por órgãos como a SAE e a ISO. Constitui em um conjunto de especificações de camada física, como o Medium Access Control (MAC) e o Logic Link Control (LLC) do modelo OSI (Open System Interconnection) já visto anteriormente, sendo necessário implementar camadas superiores com serviços adicionais para as aplicações. A FIGURA 42 mostra a comparação da estruturação das camadas da tecnologia CAN com o modelo OSI.

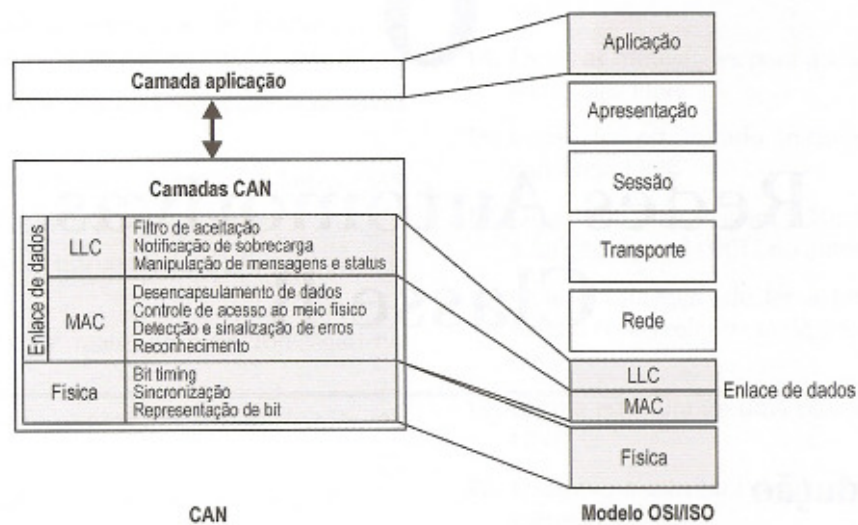


FIGURA 42: Comparativo da estrutura em camadas para os padrões CAN e OSI
Fonte: SANTOS (2010)

As principais características da rede CAN, as quais permitem o seu alto grau de aplicação em veículos, são:

- Prioridade de mensagens;
- Atraso de tempo garantido para transmissão de mensagens;
- Configuração flexível;
- Consistência dos dados no sistema;
- Paradigma multimestre;
- Controle e sinalização de erros;
- Distinção entre erros temporários e permanentes;
- Significativa imunidade a ruídos;
- Recepção Multicast com tempo de sincronização.

A transmissão de mensagens na rede CAN é extremamente controlada, sem destruição das mensagens, devido à utilização do protocolo de acesso ao meio CSMA/CD with NDA (Carrier Sense Multiple Access/Colision Detection with Non-Destructive Arbitration). Além disso, a priorização de mensagens garante a confiabilidade na transmissão. Assim, toda ECU que queira transmitir uma mensagem, primeiramente observa se o barramento está livre antes de enviar a mensagem. Caso o barramento esteja ocupado, a ECU pode tentar transmitir a mensagem em uma próxima oportunidade. Mesmo se duas ECUs tentarem acessar o

barramento ao mesmo tempo, a atribuição de prioridade de mensagens define qual mensagem será transmitida, garantindo que não haja conflito no barramento.

A Tabela 6 mostra os tipos de rede CAN disponíveis, no que diz respeito à taxa de transmissão e ao número de bits do identificador de mensagens.

TABELA 6 – TIPOS DE REDE CAN

Nomenclatura	Padrão	Taxa Mínima	Identificação
CAN 1.0A	ISO 11519	125 kbps	11 bits
CAN 2.0 A	ISO 11898: 1993	1 Mbps	11 bits
CAN 2.0 B	ISO 11898: 1995	1 Mbps	11 ou 29 bits

FONTE: Santos, 2010.

Conforme apresentado na FIGURA 43, a taxa de transmissão em uma rede CAN é inversamente proporcional ao comprimento do barramento.

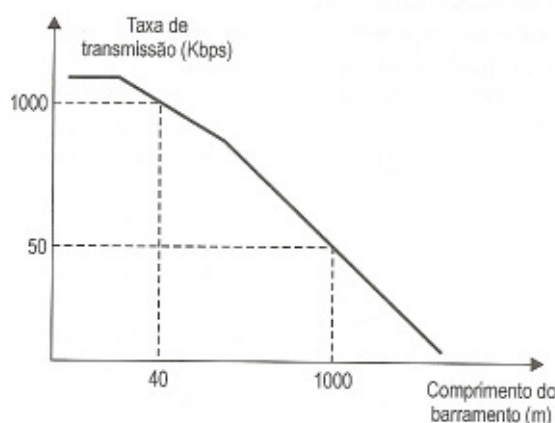


FIGURA 43: Relação entre comprimento da rede e taxa de transmissão
Fonte: SANTOS (2010)

O frame da rede CAN, conforme indicado na tabela anterior, podem ter dois tipos de identificação, os quais fazem seu tamanho diferir. O CAN 2.0A possui 11 bits de identificação e o CAN 2.0B possui 29 bits de identificação. Para um frame de 11 bits de identificação é possível ter-se 2048 mensagens, o que pode não ser suficiente dependendo da aplicação.

Já para 29 bits de identificação, o número de mensagens fica na casa dos 537 milhões, garantindo sua aplicação para sistemas que exigem maior número de mensagens. Porém, os 18 bits adicionais de identificação afetam diretamente o tempo de transmissão, aumentando-o e podendo se tornar um problema para aplicações em tempo real, mas sem comprometer o

desempenho do sistema. Na FIGURA 44 pode-se ver a estrutura dos frames do CAN 2.0A e CAN 2.0B.

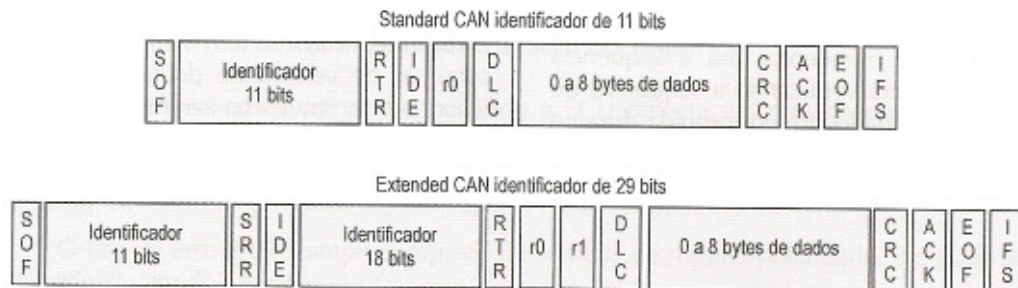


FIGURA 44: Formato de frames CAN 2.0A e CAN 2.0B
Fonte: SANTOS (2010)

Os detalhes do frame CAN podem ser vistos a seguir:

- SOF (1 bit) Start of Frame: marca o início do frame. Todas as ECUs utilizam este bit para sincronismo do clock;
- Identificador (11 bits CAN 1.0 e 2.0A e 29 bits CAN 2.0B): são transferidas na ordem de ID10 para ID0. Um menor valor neste campo indica maior prioridade na mensagem;
- RTR (1 bit) Remote Transmission Request: indica se o frame é de dados ou remoto, utilizado no identificador de 11 bits;
- SRR (1 bit) Substitute Remote Request: semelhante ao RTR, porém usado no identificador de 29 bits;
- IDE (1 bit) Identifier Extension: um bit dominante (0) é transmitido para indicar que não existem mais bits identificadores na mensagem;
- R0 (1 bit): bit reservado, transmitido como dominante;
- DLC (4 bits) Data Length Code: indica o número de bits no campo de dados, contendo o código de comprimento de dados;
- Dados (64 bits): traz os dados da mensagem;
- CRC (16 bits) Cyclic Redundancy Check: a ECU receptora utiliza esta sequência de bits para verificar se os dados foram corrompidos;
- ACK (2 bits) Acknowledge: campo de confirmação de mensagem correta recebida, enviado pela ECU receptora. Ela subscrive o bit recessivo (1) da mensagem original com um bit dominante (0);
- EOF (7 bits) End Of Frame: indica o fim do frame CAN, sendo uma sequência de 7 bits recessivos;

- IFS (7 bits) Inter Frame Space: contém a quantidade de tempo necessária para que o controlador CAN mova um frame recebido para sua posição na área de armazenamento de mensagens.

A prioridade de mensagens faz com que o protocolo CAN seja altamente utilizado em sistemas com aplicações em tempo real. O menor número binário na mensagem caracteriza-a como prioritária, conforme pode ser visto FIGURA 45.

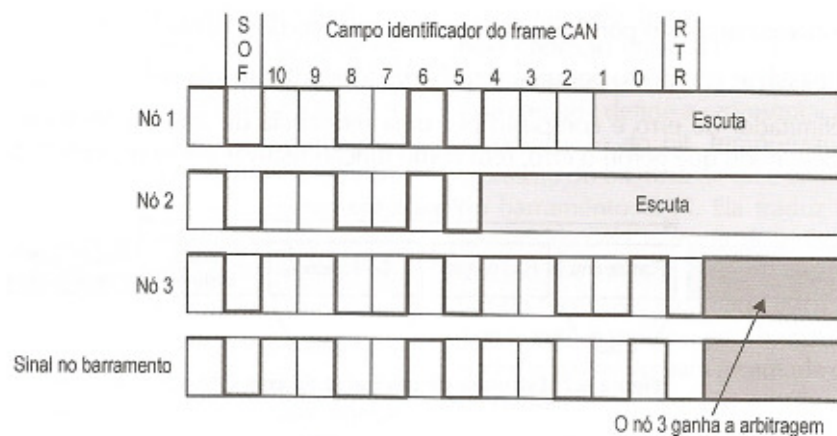


FIGURA 45: Processo de arbitragem entre frames sob o barramento CAN
Fonte: SANTOS (2010)

Quatro são os tipos de frames que podem ser transmitidos na rede CAN: frame de dados, frame remoto, frame de erro e frame de sobrecarga.

Na FIGURA 46 apresenta-se o frame mais comum, que é o de dados. Ele é praticamente todo fixo com variações limitadas ao comprimento dos dados.

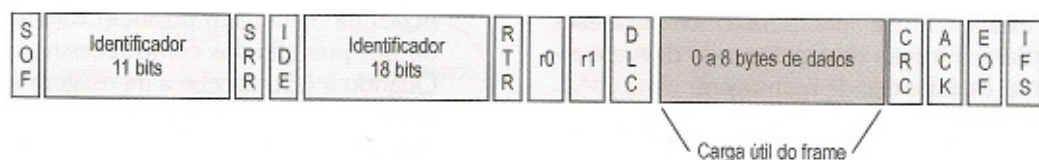


FIGURA 46: Estrutura de um frame de dados CAN
Fonte: SANTOS (2010)

Já o frame remoto é utilizado para uma solicitação de envio de dados de outra mensagem. Como apresentado na FIGURA 47, ele é semelhante ao frame de dados, porém sem a carga útil (dados) do frame e com o bit RTR em nível lógico 1 (recessivo).



FIGURA 47: Estrutura de um frame remoto CAN

Fonte: SANTOS (2010)

O frame de erro é transmitido quando é detectado um erro na mensagem. Ao verificar que existe um erro na mensagem recebida, a ECU transmissora para imediatamente o envio do quadro atual e se prepara para um novo envio da mesma mensagem. Conforme a FIGURA 48, este frame é composto do flag de erro e do delimitador de erro.

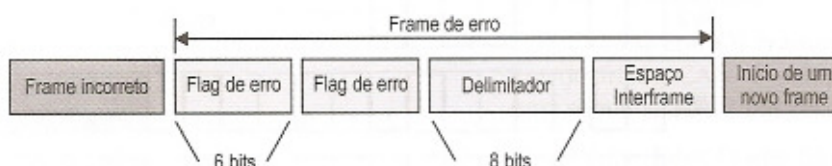


FIGURA 48: Estrutura de um frame de erro CAN

Fonte: SANTOS (2010)

O flag de erro possui seis bits de mesma polaridade e pode ser classificado em erro ativo (com seis bits dominantes) ou erro passivo (com seis bits recessivos). O delimitador de erro é enviado pelo nó que gerou o erro e consiste em uma sequência de seis bits recessivos. Tem como função reativar o barramento CAN. Os frames de sobrecarga são semelhantes aos flag de erro ativo, porém é indicado no último bit do EOF ou no campo de IFS. É utilizado quando uma ECU entra em uma situação que impossibilita o processamento de uma mensagem recebida.

A respeito da camada física CAN, SANTOS (2010) descreve o seguinte:

A camada física CAN é responsável pela conexão entre duas ECUs na rede e na transmissão real de impulsos elétricos. Essa camada define a representação do bit, a codificação do bit, temporização e sincronismo do bit, taxa de bit e comprimento do barramento CAN. Ela traduz os dados fornecidos pela camada de enlace de dados do transmissor em sinais elétricos. Ao fim da recepção, a camada física traduz os sinais, retornando-os para um formato de dados e passando para a camada de enlace de dados. (SANTOS, 2010, p.121)

O barramento CAN é composto de um par de fios trançados, onde a leitura dos sinais é feita pelo diferencial da tensão elétrica entre os fios. Com isto, aumenta-se a robustez da rede contra interferências eletromagnéticas e consequentemente a confiabilidade da leitura dos

sinais. Além destas características, em GUIMARÃES (2011) pode-se ver mais uma importante característica no que diz respeito à parte física da rede CAN:

Toda rede CAN possui dois terminadores, que nada mais são que resistores com valores entre 120 e 124 ohms, conectados à rede para garantir a perfeita propagação dos sinais elétricos pelos seus fios. Os resistores, um em cada ponta da rede, garantem a reflexão dos sinais no barramento e o correto funcionamento da rede CAN. (GUIMARÃES, 2011, p.223)

A FIGURA 49 mostra um barramento CAN com resistores de terminação.

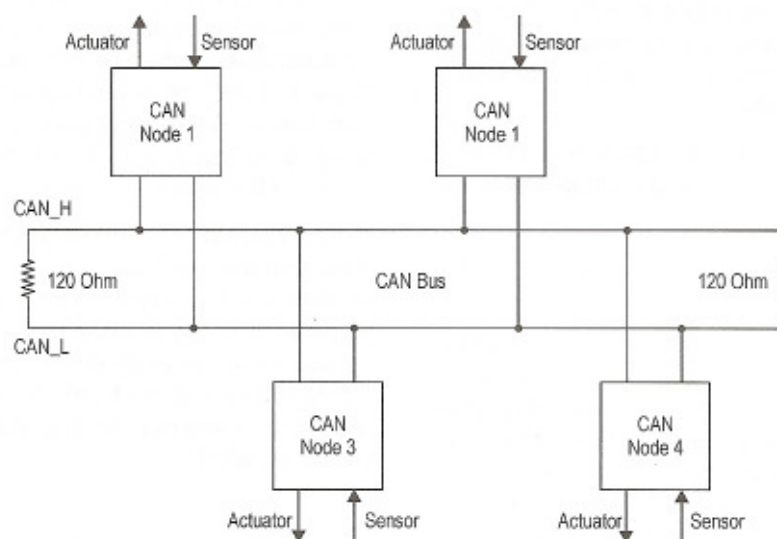


FIGURA 49: Diagrama de ligação de um barramento CAN com resistores de terminação
Fonte: SANTOS (2010)

A rede CAN possui alto grau de confiabilidade, sendo hoje sua aplicação estendida além das fronteiras automotivas. Um exemplo é o segmento industrial, onde a base de algumas tecnologias de rede industrial (DeviceNet, SDS e CANOpen) é o protocolo CAN.

5.2 SAE J1939

O protocolo J1939 foi desenvolvido para utilização em veículos comerciais, objetivando a comunicação entre motor, transmissão, caixa, trailer e outras ECUs do veículo. Foi desenvolvido a partir da rede CAN 2.0B, com identificador de 29 bits.

Além dos 29 bits de identificação de mensagem, o protocolo J1939 possui comunicação broadcast e ponto a ponto, gerenciamento de rede, definição do grupo de parâmetros e protocolo de transporte de até 1785 bytes de dados.

É um padrão utilizado para comunicação e diagnóstico, especificado pela SAE como Classe B e projetado para suportar funções de controle em tempo real. O J1939 também foi estruturado no modelo de referência OSI (Open Systems Interconnection), sendo definidas cinco camadas do modelo OSI, conforme se pode ver na FIGURA 50.

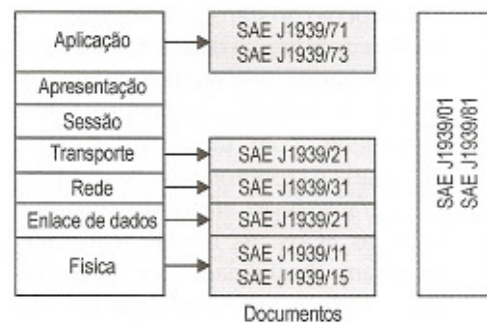


FIGURA 50: Especificações do padrão SAE J1939
Fonte: SANTOS (2010)

A seguir são descritas as funcionalidades de cada camada do protocolo:

- Camada física (J1939/11): a camada física do protocolo J1939 é feita para suportar uma taxa de transmissão de 250 kbps com resistores de terminação para reduzir reflexões. Uma rede J1939 pode ter uma ponte interligando várias seções de barramento, fornecendo isolamento elétrica entre os segmentos diferentes de rede, para que uma falha em um trecho do barramento não atinja outro trecho. O comprimento máximo do barramento é de 40 metros, podendo suportar até 30 ECUs.
- Enlace de dados (J1939/21): é o coração do protocolo. Descreve as mensagens geralmente utilizadas como pedido, reconhecimento e mensagens do protocolo de transporte. No J1939, o protocolo de transporte é responsável pela quebra de bytes de dados maiores que os suportados pelo frame de mensagens CAN em múltiplos quadros CAN. Isto permite que a ECU transmissora dos dados especifique o número de frames CAN a serem enviados.
- Camada de rede (J1939/31): responsável pela funcionalidade da ponte, determinando como as mensagens CAN são transferidas de segmento para segmento do barramento.

- Camada de aplicação (J1939/71): cada mensagem CAN é referenciada pelo PGN (Parameter Group Number). Esta camada define os formatos das mensagens, buscando a eliminação de qualquer ambiguidade de interpretação.
- Camada de aplicação (J1939/73): responsável pela padronização dos serviços de diagnóstico.
- Gerenciamento de rede (J1939/81): traz os diagramas de estado para iniciação e define o uso dos endereços de origem.

Conforme apresentado, algumas características são muito importantes dentro do protocolo J1939. Uma delas são os grupos de parâmetros. De acordo com a especificação J1939/71, estes grupos são definidos como sinais semelhantes entre si.

Cada grupo é definido por um número, o PGN (Parameter Group Number) e os grupos com até 8 bytes são transmitidos no frame CAN. Para grupos maiores, um protocolo de transporte é utilizado.

Dois tipos de PGN estão presentes no J1939: PGNs globais, utilizados para grupos de parâmetros direcionados para todas as ECUs e PGNs específicos, utilizados para grupos de parâmetros que são enviados para ECUs em particular (ponto a ponto).

O gerenciamento de rede é outra característica do protocolo J1939. Cada ECU contém um endereço no barramento, desde que o mesmo seja registrado previamente na rede. Tal procedimento consiste na “requisição de endereço”, onde uma ECU solicita determinado endereço para se registrar. Caso este endereço já esteja sendo utilizado, então, a ECU com maior prioridade fica com o endereço escolhido.

Conforme descrito no início desta seção, o protocolo J1939 é fundamentado no CAN 2.0B, o que indica a presença de um identificador de mensagens de 29 bits. A FIGURA 51 mostra a estrutura do identificador CAN na mensagem J1939.

Os três primeiros bits são utilizados para determinar a prioridade da mensagem, sendo utilizada a mesma lógica do protocolo CAN, onde um menor valor binário representa uma maior prioridade. Um quarto bit é reservado para futuras aplicações.

Prioridade 3 bits	Reservado 1 bit	Página de dados 1 bit	Formato PDU 8 bits	PDU específico 8 bits	Endereço de origem 8 bits
----------------------	--------------------	-----------------------------	-----------------------	--------------------------	------------------------------

FIGURA 51: Identificador CAN na mensagem J1939
Fonte: SANTOS (2010)

O bit de páginas de dados utiliza o valor 0 para indicar a mensagem atual que está sendo definida e o valor 1 indica o próximo campo PDU. Os 16 bits dos campos Formato PDU e PDU específico formam o PGN e é utilizado para definir mensagens, identificar comandos e dados. Os últimos oito bits (Endereço de origem) indica o endereço da ECU que está enviando a mensagem.

5.3 CCP (CAN Calibration Protocol)

O CCP (CAN Calibration Protocol) é um protocolo utilizado para calibração e aquisição de dados da ECU. Ele foi definido pela ASAM (Association for Standardization at Automation and Measure System), uma organização internacional composta por certo número de fabricantes de veículos, tais como Audi, BMW, VW, entre outros.

O foco principal deste protocolo é a compatibilidade entre diferentes tipos de hardware e software. Dentre suas aplicações, destacam-se o desenvolvimento de ECU, suporte a funções e testes de ECUs, utilização em banco de ensaio para motores, transmissão ou controle de climatização, medições e calibrações em veículos e aplicações CAN fora do domínio automotivo. Quanto à sua arquitetura, o CCP é uma camada de aplicação baseada no CAN 2.0B e suporta as seguintes funções:

- Plug and play;
- Leitura e escrita para a memória da ECU;
- Aquisição de dados cíclica e síncrona de uma ECU;
- Calibração e aquisição de dados simultânea;
- Manipulação de acesso ao barramento CAN;

- Programação em Flash;
- Proteção dos recursos (aquisição de dados e calibração).

Sobre os requisitos de hardware e software, descreve SANTOS (2010):

Com base nos requisitos de compatibilidade de hardware, software e troca de informações, definiu-se um conjunto de três interfaces básicas para o CPP, sendo ASAP1, ASAP2 e ASAP3. Posteriormente essas interfaces foram alteradas pelo ASAM, sendo denominadas ASAM-MCD1, ASAM-MCD2 e ASAM-MCD3 respectivamente. O processo de medição e calibração está conectado diretamente à ECU, através da interface ASAP1. O ASAP1 é dividido em interfaces 1a e 1b. A interface ASAP1a é uma ligação física e de enlace de dados entre a ECU e sistemas de medição e calibração e a interface ASAP1b integra as interfaces para o sistema de medição e calibração. A descrição adequada da troca de dados com a base de dados é realizada pela especificação ASAP2 e a interface ASAP3 conecta o sistema de medição e calibração ao sistema de automação para proporcionar mecanismos de testes de automatização controlados por computador.

(SANTOS, 2010, p.130)

A FIGURA 52 ilustra a definição dada acima.

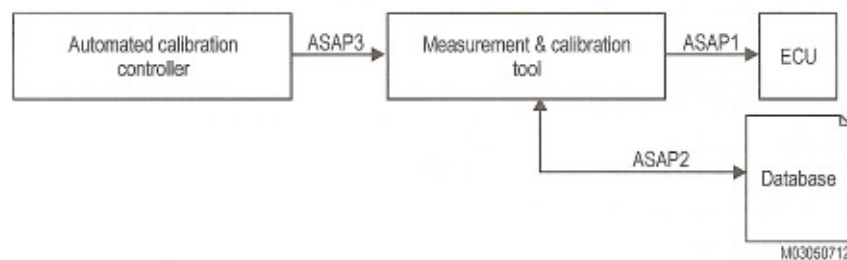


FIGURA 52: As interfaces do ASAP
Fonte: SANTOS (2010)

Conforme apresentado na FIGURA 53, o CCP tem como característica básica, a comunicação mestre/escravo, sendo o mestre que sempre inicia a comunicação.



FIGURA 53: Conexão do barramento CCP
Fonte: SANTOS (2010)

Apenas dois tipos de mensagens CAN são necessários para o CCP ser empregado:

- CRO (Command Receive Object): são enviadas do mestre para o escravo e contém comandos de controle;
- DTO (Data Transmission Object): são enviadas do escravo para o mestre.

Assim, quando um escravo recebe uma mensagem CRO, executa as funções presentes nela e responde com uma mensagem DTO CRM (mensagem de retorno de comando). O código CRM indica ao mestre se o comando de controle foi executado ou não.

A estrutura da mensagem CRO é mostrada na FIGURA 54. Consiste em um primeiro byte que identifica o código de comando (CMD), o qual descreve o propósito da mensagem. O segundo byte é utilizado para manter um rastreamento da comunicação através do contador de comando (CTR). Os bytes 2 a 7 são os dados da mensagem parametrizados de acordo com o código de comando.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
CMD	CTR	Data	Data	Data	Data	Data	Data

FIGURA 54: Organização da mensagem CRO
Fonte: SANTOS (2010)

A resposta à mensagem CRO é a mensagem DTO enviada pelo escravo. Tal mensagem também pode ser utilizada para aquisição de dados ou mesmo ser uma mensagem de evento, onde a ECU escrava registra uma alteração interna, sendo necessária uma recuperação de erros ou execução de outros serviços. A FIGURA 55 mostra a organização da mensagem DTO.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
CMD	ERR	CTR	Data	Data	Data	Data	Data

FIGURA 55: Organização da mensagem DTO
Fonte: SANTOS (2010)

O protocolo CCP é muito utilizado no campo industrial e automotivo para funções de medição e calibração. Com ele é possível a monitoração de grandezas físicas de sensores e atuadores e o empacotamento destas em mensagens no barramento.

5.4 VAN (Vehicle Area Network)

A tecnologia de rede VAN (Vehicle Area Network) foi desenvolvida em conjunto pela PSA Peugeot Citroën e Renault. Trata-se de um barramento de comunicação serial, com uma estrutura física dada em tensão diferencial com estados dominante e recessivo e codificação de dados do tipo “Enhanced Manchester”.

A ECU receptora deve usar seu clock com a mesma frequência do transmissor e definir o momento de leitura do bit, onde a amostragem é tipicamente feita entre 25% a 75% após o início do bit, conforme mostra a FIGURA 56.

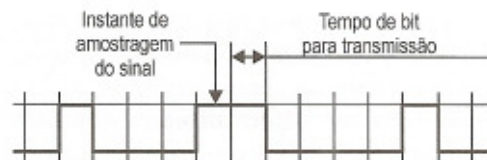


FIGURA 56: Instante de amostragem do sinal e tempo de bit para transmissão
Fonte: SANTOS (2010)

O mecanismo de transmissão de mensagens no protocolo VAN substitui o nível de tensão da informação por mudanças de estado a serem transmitidas. Assim, o bit 1 será representado por uma borda de descida e o bit 0 será representado por uma borda de subida.

Com isso, conforme se observa na FIGURA 57, um conjunto de sinais representado por mudança de estados define o formato da mensagem a ser transmitido no protocolo VAN.

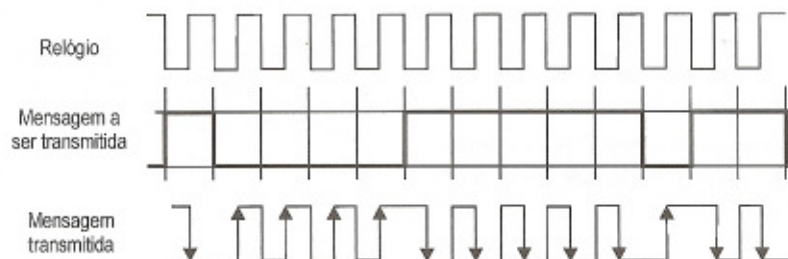


FIGURA 57: Mecanismo de transmissão de informação através da transição de estado do sinal
Fonte: SANTOS (2010)

Um frame do protocolo VAN é estruturado de acordo com a FIGURA 58. Ele é composto de oito campos:

- Preâmbulo (Start of frame): é um campo fixo que permite que as estações em uso identifiquem o início de um frame e sincronizem seus clocks;
- IDEN (Address): é o campo que contém o endereço do receptor da mensagem;
- COM (Command): permite a fixação do câmbio sobre rede;
- Dados (Data(s)): campo dos dados da mensagem;
- CTRL (FCS): campo de controle para verificação da integridade dos dados;
- EOD: fim do campo de dados;
- ACK: indica o reconhecimento no recebimento de uma mensagem;
- EOF: campo responsável por indicar o final do frame.

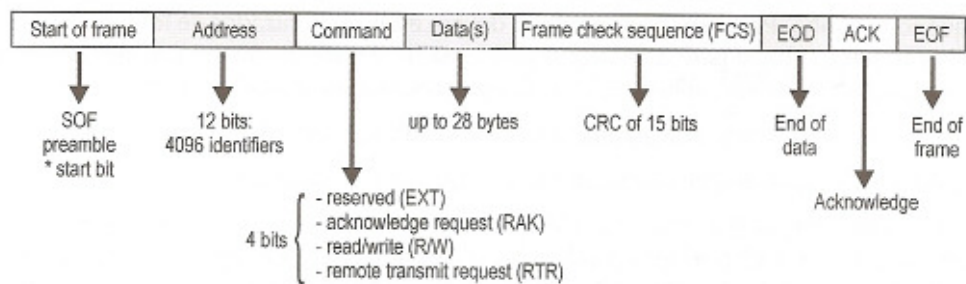


FIGURA 58: Estrutura de um frame para tecnologia VAN
Fonte: SANTOS (2010)

Na FIGURA 59, é possível ver o processo de comunicação entre duas ECUs no barramento VAN. A ECU A envia os campos de Preâmbulo, IDEN e COM mais o último bit sendo definido como 1.

Todas as ECUs do barramento recebem esta informação, mas somente a ECU B, a qual foi endereçada, responde à mensagem, enviando o campo de dados, CTRL, EOD e o último bit definido como 1. Assim, A ECU A recebe as informações desejadas e responde com um bit 0 (perfazendo ACK = 10) e finaliza a comunicação com o campo EOF.

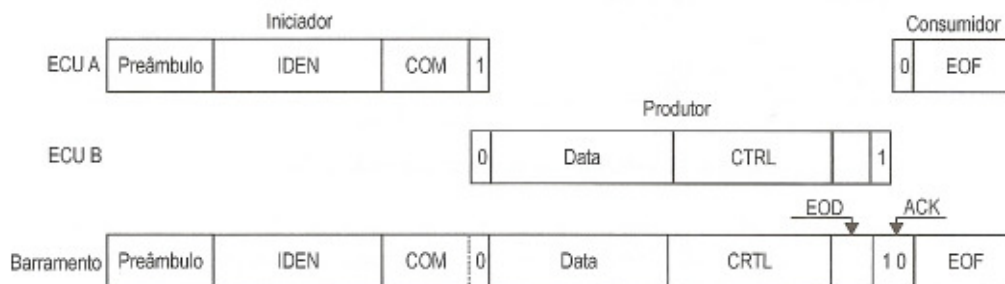


FIGURA 59: Processo de comunicação da rede VAN
Fonte: SANTOS (2010)

5.5 K-Line (ISO 9141)

A norma ISO 9141 define um protocolo de comunicação para transmissão de dados através de um ou dois fios. A taxa de transmissão é da ordem de 10,4 kbps com comprimento de mensagem de 11 bytes e mecanismo de detecção de erro checksum de 8 bits.

A atribuição de pinagens no conector DLC (Diagnostic Link Connector) está disponível na linha “K” (pino 7) e na linha “L” (pino 15). A ferramenta de diagnóstico transmite um sinal de “wake up” através destas linhas para as ECUs quando querem iniciar a comunicação.

O fluxo de dados é bidirecional entre a ferramenta de diagnóstico e a ECU através da linha “K”. Já a linha “L”, em alguns casos, funciona somente unidirecionalmente.

É utilizado somente durante a inicialização para carregar endereço ou dados com padrão “wake up” de uma ferramenta de diagnóstico para a ECU. Seu formato é uma onda quadrada de frequência uniforme e nível de tensão variável (12 Vcc ou 0 Vcc).

5.6 KWP2000 (Keyword Protocol 2000)

O KWP 2000 (Keyword Protocol 2000) é um protocolo baseado na norma ISO 14230, projetado originalmente para diagnose de ECUs que suportam este modelo de comunicação em veículos. Pode, também, ser empregado como interface de gravação de softwares em ECUs. Basicamente, a estrutura de uma mensagem no KWP 2000 é constituída de três partes (campos) principais: campo de Cabeçalho (máximo de 4 bytes), campo de Dados (máximo de 255 bytes) e Checksum (1 byte). Tal estrutura é mostrada na FIGURA 60.

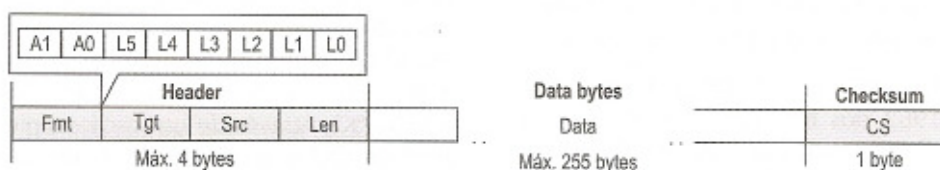


FIGURA 60: Estrutura de mensagem KWP 2000 com o campo Fmt
Fonte: SANTOS (2010)

O campo Cabeçalho é formado pelas seguintes partes:

- Fmt (Format Byte): possui 1 byte composto por 6 bits que representam informações (L5...L0) e dois bits com informação de endereçamento (A1, A0);
- Tgt (Target Address Byte): traz o endereço de destino da mensagem utilizado junto com o endereço fonte;
- Src (Source Address Byte): é o endereço físico da ECU que está transmitindo a mensagem;
- Len (Length Byte): indica a quantidade de bytes de dados que serão transmitidos no campo Data Bytes, que pode chegar a até 255 bytes.

O campo de dados (Data Bytes) traz os dados da mensagem, sendo o primeiro byte deste campo a identificação do serviço (Sid). O checksum é colocado no final do frame, sendo a soma de todos os bytes da mensagem, excluindo-se ele próprio.

A FIGURA 61 mostra uma arquitetura de diagnose baseada no protocolo KWP 2000, onde um barramento interliga as ECUs que suportam este modelo de comunicação, bem como a ferramenta de diagnóstico (Tester) que fará interface com as ECUs através do barramento e dos serviços de diagnóstico suportados pelo protocolo.



FIGURA 61: Arquitetura de diagnose para uma rede KWP 2000
Fonte: SANTOS (2010)

O conceito de funcionamento do KWP 2000 é elaborado a partir de um esquema cliente/servidor, onde a ferramenta de diagnóstico é o cliente que solicita informações para as ECUs servidoras.

Na FIGURA 62 são apresentadas as primitivas de comunicação entre elementos no KWP 2000, onde a ferramenta de diagnóstico sempre inicia o processo de comunicação através de uma solicitação à camada de transporte e, em seguida, o fluxo de comunicação obedece aos padrões OSI. As primitivas básicas de comunicação são: solicitação (request), indicação (indication), resposta (response) e confirmação (confirmation).

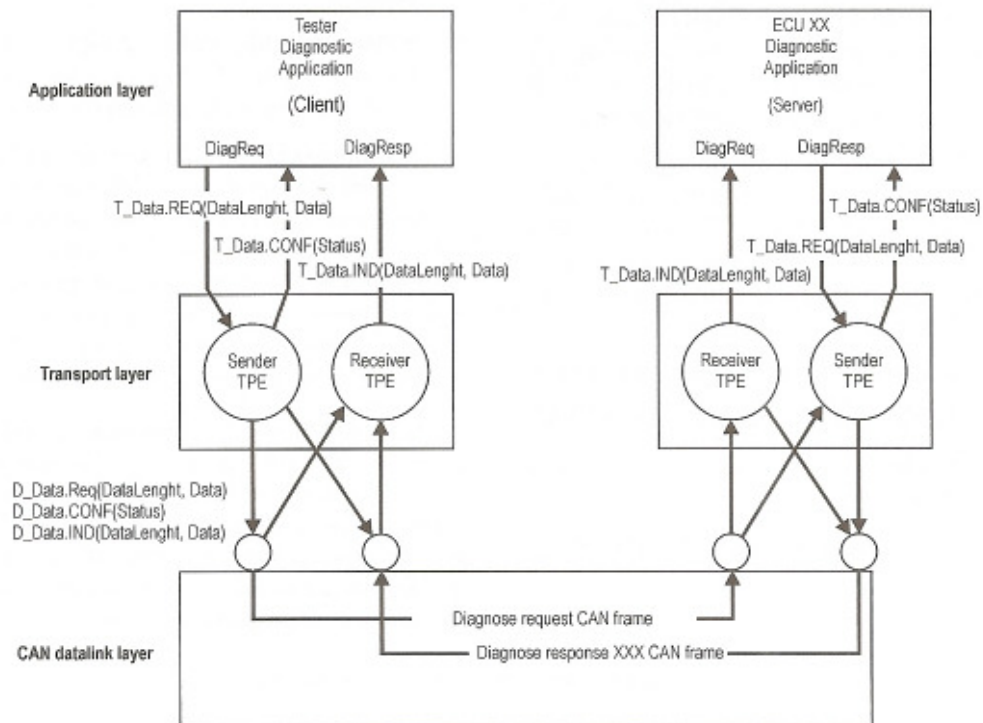


FIGURA 62: Primitivas de comunicação entre elementos no KWP 2000
Fonte: SANTOS (2010)

A FIGURA 63 ilustra um exemplo de fluxo de mensagem do KWP 2000, no qual é apresentada uma solicitação de diagnose (Tx) e uma resposta da ECU (Rx).

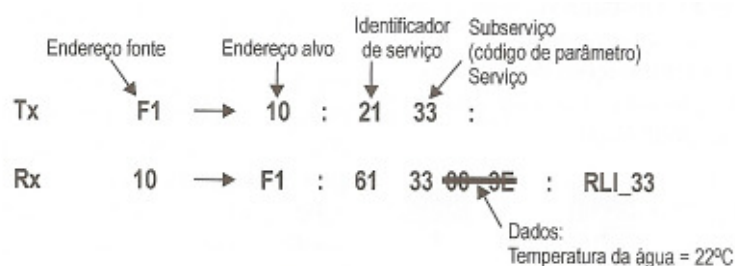


FIGURA 63: Exemplo de fluxo de mensagens para diagnose
Fonte: SANTOS (2010)

5.7 SAE J1850

O protocolo J1850 é fundamentado na transmissão de dados serial com base em dois tipos de modulação:

- PWM (Pulse Width Modulation): alta velocidade de transmissão com 41,6 kbps e tensão diferencial com dois fios.
- VPW (Variable Pulse Width): baixa velocidade de transmissão com 10,4 kbps, baseado em um fio único.

O J1850 transmite dados em frames entre as ECUs de um veículo na forma de sinais digitais, utilizando o conceito de multiplexação. O esquema de multiplexação VPW possui baixa emissão de radiação, compensação para incompatibilidade de clocks e offset a terra, sendo o esquema de codificação mais adotado nas aplicações do protocolo J1850.

O conceito de prioridade no protocolo SAE J1850 é explanado por SANTOS (2010) da seguinte forma:

A norma SAE J1850 dá atribuição de rede a cada nodo usuário com base no conceito de arbitragem, processo que determina qual dos nodos tem prioridade de acesso ao barramento quando dois ou mais deles desejam transmitir mensagem no barramento de forma simultânea. O barramento J1850 é assíncrono, sem mestre, ponto a ponto que oferece acesso igual ao barramento para cada nodo. Um atributo importante dele é que um nodo transmissor difunde sua mensagem sob o barramento. Isso significa que não só todos os outros nodos recebem a mensagem transmitida, mas o nodo de transmissão vê sua própria mensagem sendo transmitida sob o barramento. Todas as mensagens são assíncronas por natureza, e qualquer dispositivo enviando uma mensagem determina quando uma mensagem de transmissão pode iniciar, podendo realizar a transmissão entre mensagens sem um intervalo predefinido.

(SANTOS, 2010, p.143)

O frame de mensagem do protocolo J1850 é composto pelos seguintes campos:

- SOF: marca o início de um frame.
- Header: possui identificador com um ou três bytes.
- Prioridade (3 bits) com o valor 000 sendo de maior prioridade;
- Comprimento do frame (1 bit);
- Requerimento do IFR (In-Frame Response) (1 bit);

- Modo de endereçamento (1 bit), sendo 1=físico e 0=funcional;
- Tipo de mensagem (2 bits);
- Dois bytes adicionais se utilizar o endereçamento físico (endereço de destino (8 bits) e endereço fonte (8 bits)).
- Data bytes: campo dos dados a serem transmitidos;
- CRC: mecanismo de controle de erro (Cyclical Redundancy Check);
- NB: bit de normalização;
- IFR Data: dados de IFR (In-Frame Response);
- IFR CRC: check para o campo de IFR Data.

A FIGURA 64 apresenta a estrutura de um frame J1850.

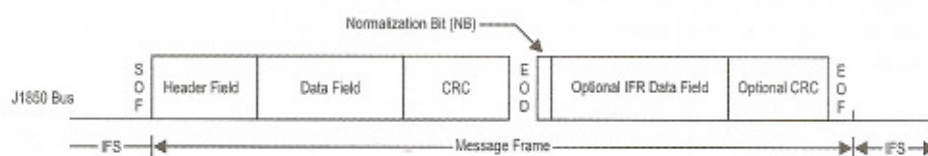


FIGURA 64: Mensagem SAE J1850 com IFR
Fonte: SANTOS (2010)

Conforme apresentado no início desta seção, o protocolo J1850 utiliza a modulação de pulsos. Assim, é de extrema importância a definição do tempo de bit dentro do J1850. A definição dos bits 0 e 1 ativos e passivos podem ser vistas na FIGURA 65.

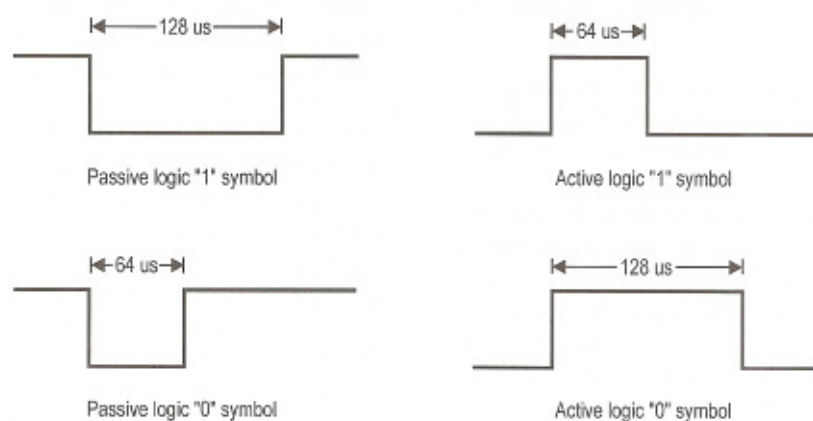


FIGURA 65: Tempo de bit do SAE J1850 VWP
Fonte: SANTOS (2010)

Ao final de cada mensagem, o mecanismo de controle de erro CRC se faz presente e, por fim, um símbolo de EOD (End of Data) é transmitido.

O IFR (In-Frame Response) pode ser solicitado pelo bit de requerimento do IFR constante no cabeçalho da mensagem. Assim, a ECU receptora deve ler este bit e, se solicitado, deve enviar o IFR, anexando-o na mensagem original enviada pela ECU transmissora, entre o símbolo EOD e EOF (End of Frame). Tal mecanismo permite maior eficiência na transmissão de mensagens. Após o IFR, um CRC correspondente somente ao dado IFR é enviado, seguido do sinal de final de frame (EOF).

Em relação à parte física do protocolo J1850, esta é formada por apenas um fio de tensão conduzido sem restrições de roteamento, porém com comprimento máximo de 35 metros dentro do veículo e 5 metros fora do veículo, totalizando 40 metros. O número total de equipamentos suportados é de 32 nós.

6 REDES AUTOMOTIVAS CLASSE C

As redes automotivas Classe C são tecnologias de comunicação que possuem propriedade de tempo real críticas e tolerantes a falhas. São constituídas por barramentos com alta taxa de transmissão (da ordem de 1 Mbps a 25 Mbps), capazes de oferecer ciclos de comunicação extremamente rápidos com atrasos e variações de atrasos para mensagens menores que 1 ms.

Tais tecnologias são aplicáveis a sistemas que demandam um processamento e transmissão de dados extremamente rápidos, com uma excelente tolerância a falhas e flexibilidade operacional. Dentro de um veículo, alguns sistemas que exigem este tipo de característica são o controle de estabilidade, funções x-by-wire, entre outras.

A seguir, têm-se as principais características de algumas tecnologias com este perfil: FlexRay, TTP/C (Time-Triggered Protocol/Class C), ByteFlight e Safe-by-Wire.

6.1 FlexRay

No final dos anos 90, a BMW e a Daimler-Chrysler analisaram as tecnologias de redes disponíveis e sentiram a necessidade de desenvolver uma nova tecnologia para atender os sistemas automotivos da próxima geração, especialmente aqueles baseados em controles x-by-wire.

Assim, foi criado um consórcio de empresas denominado FlexRay, que tinha como principal objetivo o desenvolvimento de uma nova tecnologia de rede que pudesse satisfazer a demanda de requisitos exigida pelos veículos da próxima geração. Desta forma iniciou-se o desenvolvimento da tecnologia FlexRay. Esta tecnologia possui uma combinação de transmissão de mensagens de seções síncrona e assíncrona, que oferece comunicação tolerante a falhas em alta velocidade.

Assim é definida por SANTOS (2010) a camada física da tecnologia Flex Ray:

A camada física da tecnologia FlexRay possui soluções alternativas elétricas e ópticas disponíveis na forma de canais de comunicação simples e duplicado, tendo a capacidade de formar topologias distintas. Considere uma arquitetura com várias ECUs, mas apenas um par de ECUs pode comunicar-se de cada vez, tendo assim uma ECU emissora e outra receptora, através de um canal de comunicação. No FlexRay pode existir apenas um canal de

comunicação, denominado simples (single-channel), ou dois canais de comunicação, denominados duplos (dual-channel), e mais de um par de ECUs pode comunicar-se ao mesmo instante.
(SANTOS, 2010, p.153)

A taxa de transmissão de uma rede FlexRay pode chegar a 10 Mbps e com certeza pode ser aplicada a sistemas veiculares de segurança crítica. Suas principais vantagens são a elevada confiabilidade em relação à tolerância de falhas, arquitetura simplificada de rede, redução da quantidade de subsistemas e sistemas eletromecânicos (x-by-wire) com maior quantidade de funcionalidade substituindo componentes hidráulicos.

A FIGURA 66 mostra um exemplo de arquitetura distribuída com a utilização da tecnologia FlexRay.

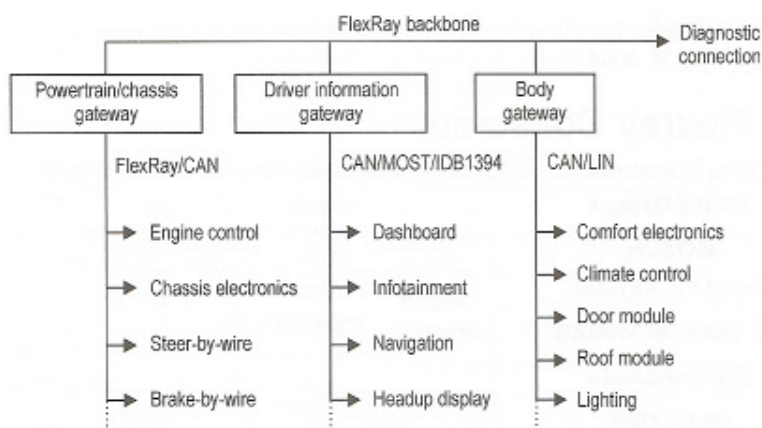


FIGURA 66: Arquitetura distribuída com backbone FlexRay
Fonte: SANTOS (2010)

Quanto às topologias utilizadas no FlexRay, citam-se basicamente três modelos, conforme apresentado na FIGURA 67. No modelo de barramento (a), as ECUs podem ser ligadas a um ou dois canais de comunicação. Na segunda topologia apresentada (b), as ECUs estão ligadas em estrelas através dos acopladores de estrela passiva ou ativa (A e B). Nesta topologia é possível fazer a ligação das redes em cascata, ligando dois acopladores de rede diretamente. O terceiro modelo (híbrido) (c) nada mais é que uma combinação da topologia por barramento com a topologia estrela.

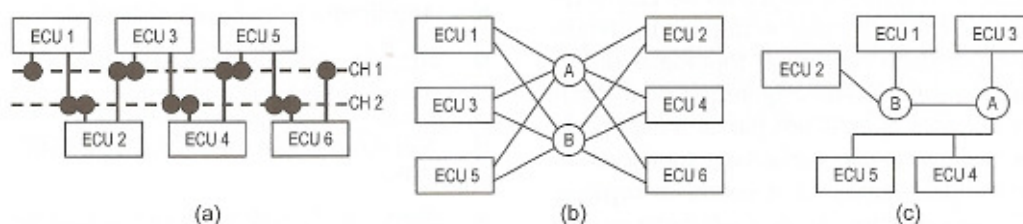


FIGURA 67: Topologias para rede FlexRay: a) barramento; b) estrela e c) híbrida
Fonte: SANTOS (2010)

No caso de utilização de canais simples entre as ECUs, o resultado é um menor custo na implementação do FlexRay, uma vez que a quantidade de cabos é reduzida. Porém, o canal duplicado tem a vantagem de apresentar uma maior tolerância a falhas, pois as mensagens podem ser enviadas por ambos os canais, introduzindo um cenário de redundância, requerido especialmente em sistemas x-by-wire. Outra vantagem do canal duplicado é a possibilidade de dobrar a largura de banda, caso uma redundância não seja requerida. Assim, mensagens diferentes seriam enviadas em cada canal.

As mensagens FlexRay são enviadas em frames contendo de 0 a 254 bytes de dados e 5 bytes de informações de controle da comunicação.

O FlexRay funciona de acordo com o princípio da TDMA (Time Division Multiple Access), onde os componentes ou mensagens alocam slots de tempo fixos nos quais eles possuem acesso exclusivo ao barramento. A desvantagem deste modelo é que esta alocação fixa de tempo para cada componente leva a uma situação onde a largura de banda não é totalmente explorada. Por isso, o FlexRay subdivide seu ciclo em dois segmentos: estático e dinâmico. Os slots de tempo fixos são situados no segmento estático do ciclo, no início de cada ciclo do barramento. Já no segmento dinâmico, os slots de tempo são atribuídos dinamicamente. Assim, a largura de banda é otimizada na tecnologia FlexRay. Na FIGURA 68, é possível ver o ciclo de comunicação FlexRay.

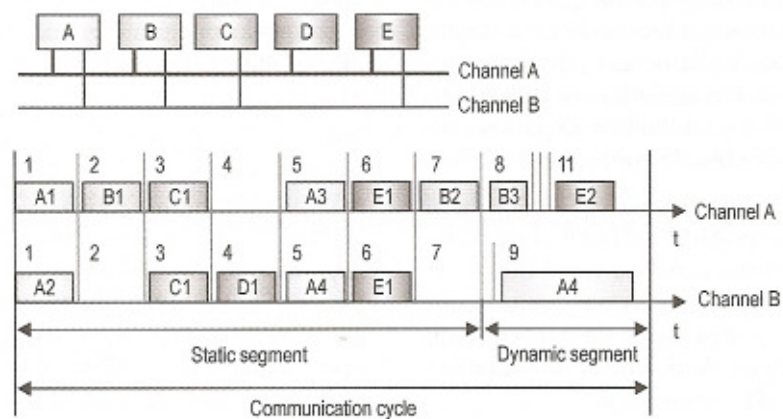


FIGURA 68: Ciclo de comunicação FlexRay
Fonte: SANTOS (2010)

6.2 TTP/C (Time-Triggered Protocol/Class C)

O protocolo TTP/C (Time-Triggered Protocol/Class C) consiste em uma segunda versão do protocolo Time Triggered Protocol (TTP) e foi concebido para aplicações de tempo real, tolerante a falhas, utilizando o TDMA (Time Division Multiple Access) como estratégia de acesso ao barramento, onde para toda ECU é permitida periodicamente a utilização de toda a capacidade de transmissão do barramento.

Este protocolo é tipicamente aplicado no meio automotivo para sistemas de segurança crítica, tais como freio ABS, airbag, injeção eletrônica, x-by-wire, entre outras. A FIGURA 69 mostra um exemplo de aplicação da tecnologia TTP/C para um sistema steer-by-wire.

O TTP/C possui algumas características que o classificam como um protocolo tolerante a falhas, tais como:

- Clock de sincronização tolerante à falha que estabelece o tempo base global sem precisar confiar no tempo central do servidor;
- Serviço que informa a cada ECU a consistência da transmissão dos dados;
- Técnica de replicação dos canais de comunicação.

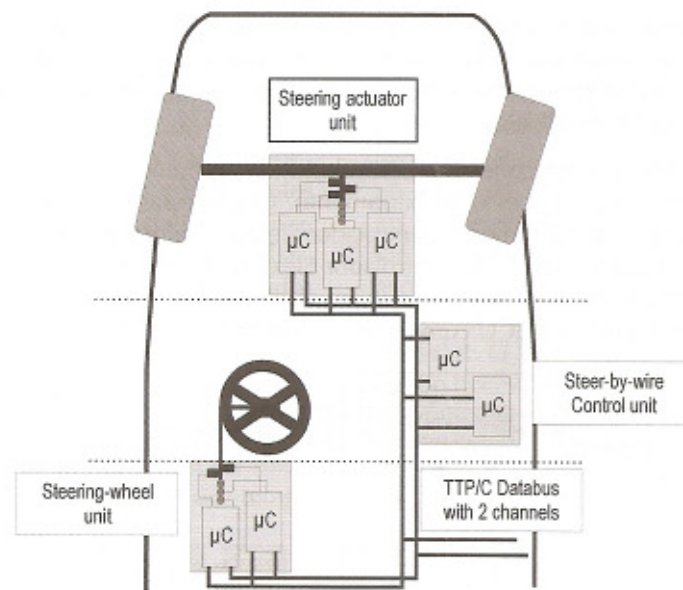


FIGURA 69: Arquitetura steer-by-wire com tecnologia TTP/C
Fonte: SANTOS (2010)

A FIGURA 70 mostra a topologia em barramento do protocolo TTP/C, a qual é semelhante a que foi apresentada no FlexRay, onde existe a possibilidade de comunicação com os dois barramentos de forma distinta.



FIGURA 70: Topologia em duplo barramento para o TTP/C
Fonte: SANTOS (2010)

O TTP/C apresenta, ainda, um serviço onde outras ECUs podem executar algumas tarefas de uma ECU com defeito. Para isso, é necessário um limite superior de tempo para que as ECUs detectem a falha, distribuam as atividades da ECU com defeito entre elas e executem as tarefas antes que uma falha mais grave do sistema ocorra.

A taxa de transmissão da tecnologia TTP/C varia entre 2 Mbps e 25 Mbps, podendo ter algumas opções de meio físico:

- Cabeamento par trançado: 2 Mbps em ISO 11898-2;
- Cabeamento par trançado: 5 Mbps em RS-485;
- Fibra óptica: 5 Mbps;

- Ethernet PHY (100Base T/F): 25 Mbps.

A FIGURA 71 mostra a estrutura de redundância do protocolo TTP/C. O conceito está na duplicação e agrupação de nodos, que formam uma FTU (Fault Tolerant Unit). Assim, o nodo é uma pequena unidade duplicada denominada SRU (Smallest Replaceable Unit), onde as SRUs dentro de uma mesma FTU são capazes de executar as mesmas funções e possuem acesso a ambos os barramentos.

Existe ainda outro nodo dentro da FTU denominado Shadow (sombra). Os Shadows são componentes redundantes em standby e realizam os mesmos processos de recepção de mensagens, porém não podem realizar transmissões de dados. Em caso de falha de um nodo SRU, o nodo Shadow assume sua função e garante a atividade no barramento. Tal estrutura reforça e aumenta a confiabilidade do sistema, caracterizando o protocolo TTP/C como uma tecnologia de redes automotivas Classe C.

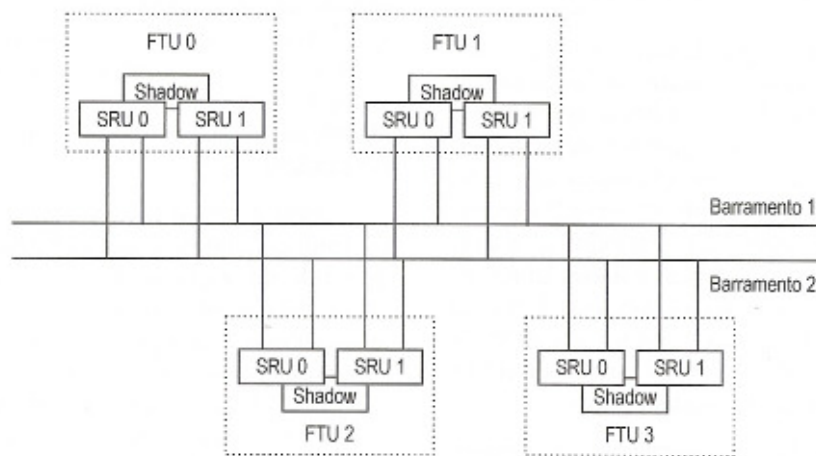


FIGURA 71: Topologia com um cluster TTP/C com unidades FTU e SRU
Fonte: SANTOS (2010)

O protocolo TTP/C possui dois tipos distintos de frames (os I-frames e os N-frames). Cada um deles possuem, basicamente, três campos:

- Header (cabeçalho): contém o bit de início de frame, seguido de 3 bits utilizados para troca do modo de operação e mais 4 bits de acknowledgement para reconhecimento das SRUs;
- Data Field (campo de dados): traz os dados da mensagem;
- CRC: 2 bytes que fazem parte do mecanismo de detecção de erros.

Os I-frames são utilizados na inicialização do sistema. São enviados pelo subsistema de comunicação durante o startup do protocolo e predeterminam intervalos durante a operação normal do protocolo para tornar mais fácil o restabelecimento dos nodos com falha. Já os N-frames são enviados durante a operação normal do protocolo e são utilizados para a transmissão de dados. A FIGURA 72 mostra os dois tipos de frame existentes no protocolo TTP/C.

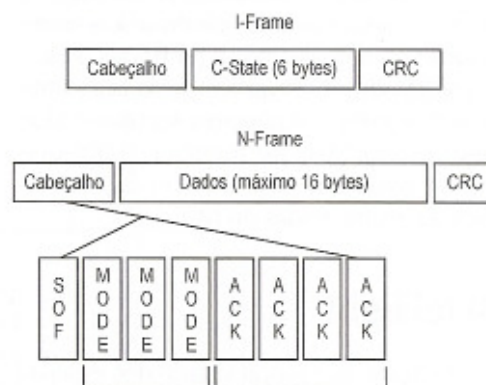


FIGURA 72: Tipos de frame do TTP/C
Fonte: SANTOS (2010)

Para que o TTP/C seja um protocolo tolerante a falhas, uma FTU envia quatro cópias redundantes de uma mensagem dentro de dois slots de tempo TDMA consecutivos. Assim, no primeiro slot TDMA de determinada FTU, a SRU 0 transmitirá a mensagem no barramento A, enquanto a SRU 1 transmitirá pelo barramento B. No slot seguinte, ocorre a inversão, quando a SRU 0 transmite no barramento B e a SRU 1 transmite no barramento A. Tal estratégia é denominada de transmissão cruzada (crossed sending) e pode ser vista na FIGURA 73.

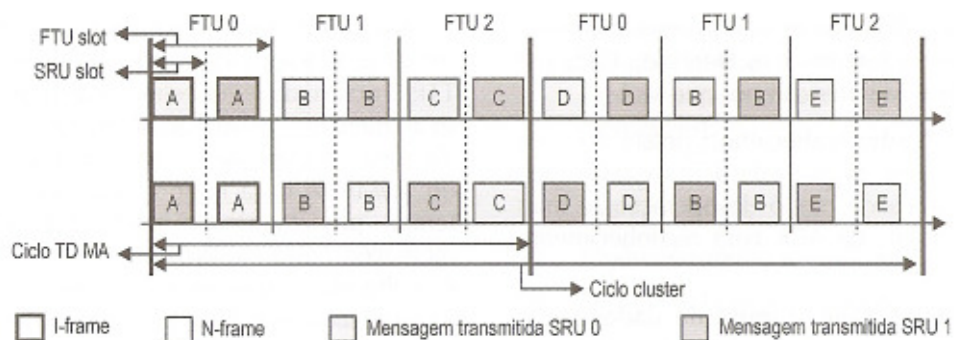


FIGURA 73: Esquema de acesso ao meio do TTP/C
Fonte: SANTOS (2010)

6.3 ByteFlight

O protocolo ByteFlight foi desenvolvido por um consórcio de empresas liderado pela BMW, com o objetivo de sua utilização em sistemas críticos de segurança. A FIGURA 74 mostra o grupo de empresas responsáveis pelo protocolo ByteFlight.



FIGURA 74: Grupo de empresas responsáveis pelo ByteFlight
Fonte: SANTOS (2010)

O protocolo ByteFlight busca satisfazer as aplicações de segurança crítica (ativa ou passiva) em substituição à tecnologia CAN, pois requisitos de alta largura de banda e previsão temporal são necessários para algumas aplicações, tais como: airbag, seat belt pretensioners (pré-tensores dos cintos de segurança), freios ABS etc.

Conforme já foi mencionado no primeiro capítulo, o veículo BMW 7 possui em sua arquitetura eletroeletrônica o protocolo ByteFlight. Além da aplicação desta tecnologia por parte da BMW, áreas da aviação também já estão utilizando o ByteFlight, tornando este protocolo um candidato (ao lado do FlexRay) à aplicação em sistemas x-by-wire.

Quanto ao mecanismo de acesso ao barramento, SANTOS (2010) descreve:

O ByteFlight utiliza o mecanismo de acesso ao barramento FTDMA (Flexible Time Division Medium Access), tipicamente baseado na topologia estrela (embora as topologias em barramento e cluster sejam também possíveis), podendo garantir um atraso previsível para uma quantidade limitada de mensagens em tempo real de alta prioridade.
(SANTOS, 2010, p.164)

Além do mecanismo de acesso descrito acima, existe a possibilidade de se enviar mensagens que não tenham requisitos de tempo real com baixa prioridade em uma forma priorizada pelo mecanismo de mini-slotting.

Quanto à sincronização de clock, esta é fornecida por um nodo mestre, realizando pulsos de disparo na ordem de $100\mu\text{s}$. Vale ressaltar que no ByteFlight qualquer nodo pode ser configurado como mestre.

Outro ponto importante deste protocolo é a possibilidade de mascarar modos de falha, como o babbling idiot. (tagarelator) com a utilização de acopladores estrela (star coupler).

A FIGURA 75 mostra o diagrama temporal do protocolo FTDMA, onde as mensagens são escalonadas de forma cíclica, onde todos os nodos mantêm um contador de slot. Um nodo faz o papel do sync master (ou mais de um, quando se deseja uma situação de redundância), o qual inicia o ciclo de comunicação pelo envio de um sync pulse (pulso de sincronização). O período de um ciclo de sincronização é de $250\mu\text{s}$.

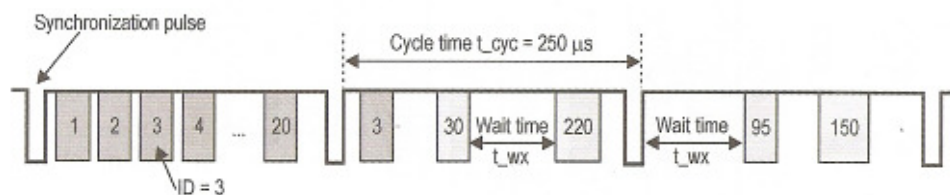


FIGURA 75: Diagrama temporal do protocolo FTDMA
Fonte: SANTOS (2010)

Após o pulso de sincronização, os contadores de slots começam em zero e vão até o maior valor possível do identificador. Quando um contador de slot atinge um valor de identificador para que um pedido de transmissão esteja presente, a mensagem é transmitida através do barramento e todos os contadores de slot param no valor atual durante a duração da transmissão.

O protocolo ByteFlight possui uma alta taxa de transferência (10 Mbps) e uma taxa de atualização da informação de $250\mu\text{s}$, com a garantia da integridade dos dados.

O meio físico selecionado para este protocolo é a fibra óptica, a qual traz redução da interferência eletromagnética e garante uma boa largura de banda.

6.4 Safe-by-Wire

O protocolo Safe-by-Wire (SbW) é assim definido por SANTOS (2010):

O Safe-by-Wire (SbW) é um protocolo do tipo mestre/escravo com uma especificação que descreve as camadas física, enlace de dados e aplicação de acordo com o modelo de referência OSI. Consiste em um barramento com dois pares de tensão diferencial com fio trançado, com três níveis de dados e um nível de potência. O nível de tensão diferencial varia de 11 a 0 volt, com bits dominantes e recessivos, similar ao CAN. A ECU mestre fornece alimentação às ECUs escravas através da linha de potência. (SANTOS, 2010, p.167)

O SbW permite algumas variações de topologia, tais como paralelo, Daisy-chain, árvore e anel. A FIGURA 76 mostra um exemplo de uma rede SbW com sensores separados e fios de par trançado.

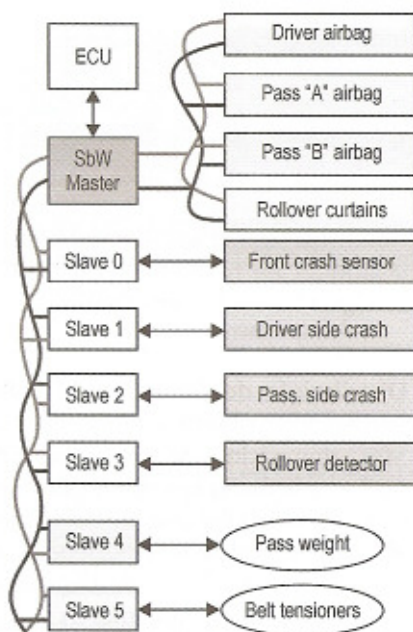


FIGURA 76: Exemplo de uma rede Safe-by-Wire
Fonte: SANTOS (2010)

O barramento do protocolo SbW é um par trançado e a comunicação entre as ECUs (mestre e escravos) se dá pelos níveis de tensão e corrente. O bit dominante é representado pelo nível de tensão baixo no barramento e o bit recessivo pelo nível de tensão alto no barramento. O mestre configura um frame onde os escravos inserem os dados na posição adequada.

A FIGURA 77 mostra os níveis de tensão do barramento, os quais são assim definidos:

- Distribuição de potência – 11 volts: é a fase onde o barramento fornece alimentação para os escravos. Através deste nível de tensão colocado pelo mestre no barramento, os escravos mantêm a alimentação necessária para seu funcionamento.
- L0: Nível Dado “0” – 6 volts: este é um nível recessivo e significa que a tensão no barramento pode ser colocada ao nível de dados “1”, mas não para P. Somente o mestre pode colocar o barramento em P e apenas na fase de distribuição de potência.
- L1: Nível Dado “1” – 3 volts: nível dominante representado pela tensão diferencial de 3 volts. Tanto mestre como escravo podem colocar o nível de 3 volts no barramento, porém nenhum dispositivo pode colocar 6 volts no barramento.
- LS0: Nível Dado Especial “0” – 0 volt: é um nível dominante sobre todos os outros e é utilizado para que um dispositivo defina um nível que tenha maior prioridade sobre todos os outros dispositivos. O LS0 é utilizado como um sinal de interrupção.



FIGURA 77: Estados de tensão do barramento diferencial
Fonte: SANTOS (2010)

O protocolo SbW possui dois tipos de frame: o D-frame e o S-frame. Os D-frames são utilizados para transporte de dados de diagnose e controle e os S-frames são utilizados para transmitir informações do sensor de colisão a uma elevada velocidade.

Os frames trazem alguns campos como cabeçalho, endereço escravo, dados do mestre ou escravo e CRC.

Os D-frames podem ser utilizados com endereçamento ponto a ponto (FIGURA 78) ou bitmap (FIGURA 79). O D-frame é identificado pelo bit T em “1”.

O endereçamento ponto a ponto é utilizado para comunicação de diagnóstico, podendo enviar comandos para escravos. Já o endereçamento bitmap é utilizado para controlar um ou mais dispositivos utilizáveis num determinado instante.

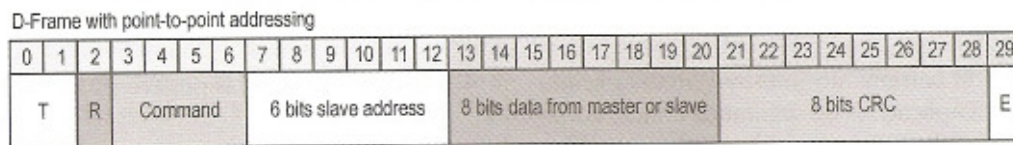


FIGURA 78: D-frame com o endereçamento ponto a ponto
Fonte: SANTOS (2010)

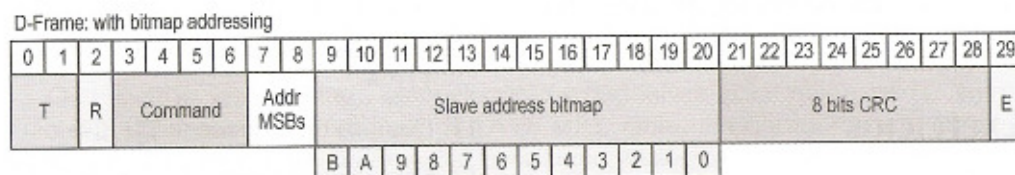


FIGURA 79: D-frame com endereçamento bitmap
Fonte: SANTOS (2010)

O S-frame é identificado pelo bit T em “0” e possui comprimento variável de acordo com o número de escravos no sistema e o comprimento de dados. A FIGURA 80 mostra um exemplo de S-frame considerando três escravos no sistema.

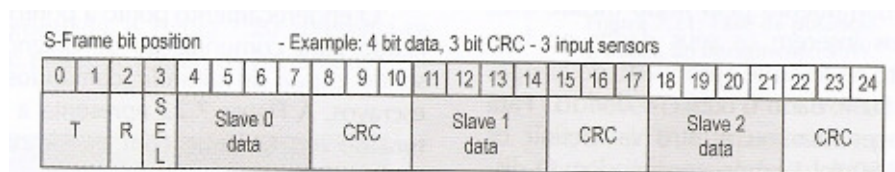


FIGURA 80: S-frame com três sensores e 4 bits de dados
Fonte: SANTOS (2010)

O barramento Safe-by-Wire é indicado para aplicações de segurança crítica, como airbags distribuídos pelo veículo, onde uma alta taxa de transmissão é requerida e a confiabilidade é essencial. A mesma lógica se aplica aos pré-tensores dos cintos de segurança, onde o funcionamento se dá em condições semelhantes às do funcionamento dos airbags.

7 REDES DE ENTRETENIMENTO

O número de dispositivos de entretenimento presentes em um veículo aumenta cada vez mais, agregando funções e trazendo facilidade e conforto ao motorista.

Alguns exemplos de dispositivos e funções de entretenimento podem ser listados, como tocadores de DVD, MP3, rádio, amplificadores, alto-falantes, microfones, conversores de áudio, sistemas de reconhecimento de voz, sistemas de informação, displays, sistemas de navegação, telemetria, processamento de dados, entre outros.

Com tantas opções de dispositivos, é natural que se queira interligá-los para que propiciem ainda mais funções ao usuário. Assim, tecnologias de rede dedicadas à interconexão deste tipo de dispositivos foram desenvolvidas para atender os requisitos de confiabilidade, largura de banda, atraso, variação de atraso, escalabilidade, meios de comunicação, interfaces plug and play etc.

A seguir, algumas destas tecnologias de rede serão apresentadas, sendo elas o D2B, IDB-1394 (FireWire®) e MOST.

7.1 D2B (Domestic Digital Bus)

O D2B (Domestic Digital Bus) é uma tecnologia de comunicação em fibra óptica com topologia em anel e largura de banda de 12 Mbps.

Trata-se de uma arquitetura aberta para comunicação de dados multimídia, conectando áudio, vídeo, periféricos de entretenimento, dispositivos de telecomunicação, entre outros. Possui escalabilidade, onde um novo dispositivo pode ser acrescentado à rede sem alteração de suas características, além de ser compatível com tecnologias anteriores.

Tanto fibras ópticas quanto pares trançados podem ser utilizados como meio físico para este protocolo, sendo uma melhor propagação sendo atingida com a utilização de fibra óptica.

O desenvolvimento do D2B foi feito pela Philips no início dos anos 80, com aplicações para comunicação de dispositivos domésticos do tipo áudio e vídeo. Porém, naquela época

percebeu-se que o D2B estava à frente do seu tempo e isto despertou o interesse de outros segmentos da indústria, dentre eles o setor automotivo.

O barramento do D2B opera de modo assíncrono sob um par de fios diferencial. O frame de mensagem possui cinco campos, a saber: header (cabeçalho), master address (endereço mestre), slave address (endereço escravo), control (controle) e information (informação). A estrutura do frame D2B é mostrada na FIGURA 81.

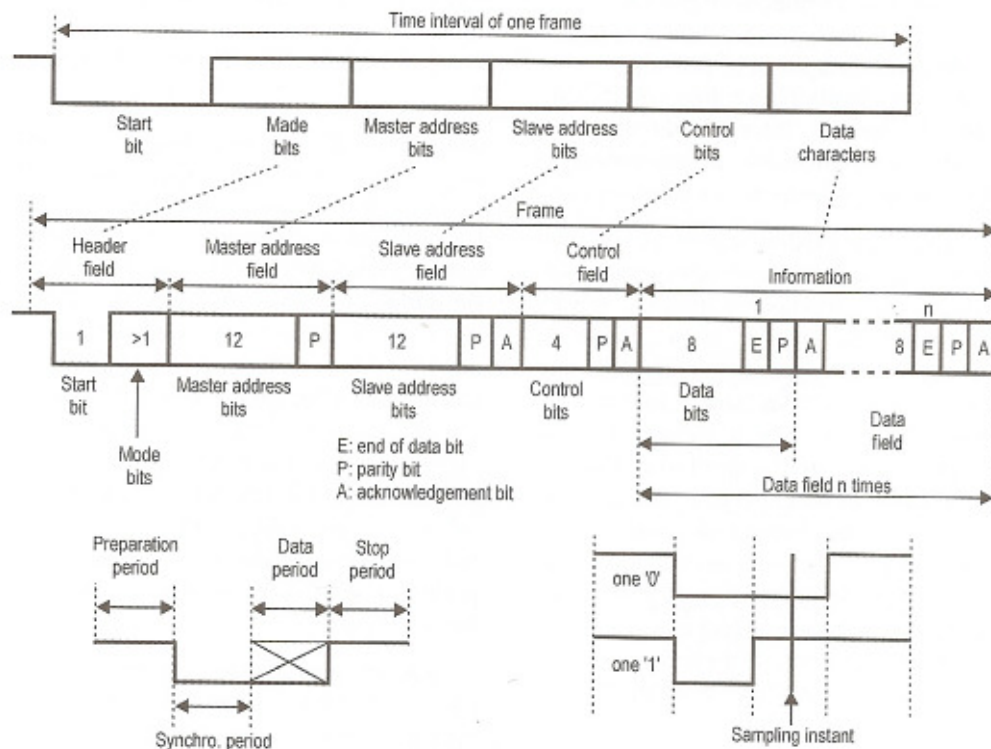


FIGURA 81: Estrutura de um frame D2B
Fonte: SANTOS (2010)

O D2B opera a uma distância máxima de 150m. Ele foi projetado desta forma, pois esta é a distância máxima ponto a ponto em uma instalação padrão para residências ou veículos.

Os dispositivos na rede D2B são conectados através de uma topologia em anel com cabos e fibra óptica entre cada dispositivo e em topologia estrela através de linha de wake-up, conectados ao mesmo ponto. A linha de wake-up é alimentada pela tensão de bateria do mestre da rede e um sinal wake-up é enviado na linha de wake-up do mestre D2B. Tal estrutura pode ser vista na FIGURA 82.

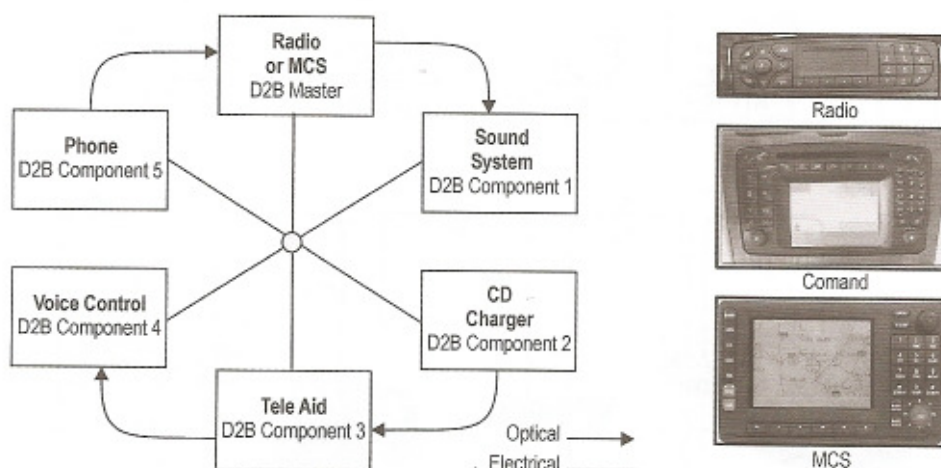


FIGURA 82: Configuração em anel D2B (Class M)
Fonte: SANTOS (2010)

7.2 IDB-1394

O protocolo IDB-1394 é uma versão de alta velocidade do padrão IEEE1394 conhecida como FireWire® e foi planejada para utilização em aplicações multimídias de alta velocidade, sendo capaz de atingir uma taxa de transmissão de até 400 Mbps.

Consiste em um barramento serial bidirecional, normalmente utilizado para câmeras de vídeo digital, discos rígidos e outros dispositivos de alta velocidade. O meio físico é composto por fibras ópticas, podendo ser fibras ópticas plásticas (POF) ou fibras de sílica endurecidas (HCS). Dependendo do meio, as taxas de transmissão podem chegar até 100 Mbps e 200 Mbps para um comprimento de barramento de 50 m (POF) e 100 m (HCS). Para comprimentos menores pode-se chegar até 400 Mbps, com possibilidade no futuro de chegar até a 3200 Mbps.

O número máximo de nodos que podem ser conectados ao barramento é 63, sendo a topologia física podendo ser um barramento em “daisy-chain” ou em árvore. Para aplicações automotivas, ainda é possível a utilização da topologia em anel, que proporciona confiabilidade na disponibilidade do sistema, pois mesmo com um ou mais nodos defeituosos a comunicação não é interrompida.

O IDB-1394 possui especificação da camada 1 até a camada 4 do modelo OSI, as quais são descritas a seguir:

- Camada física: inclui a porta de conveniência (CCP), a qual fornece a interface para o protocolo.
- Pilha e driver IEEE1394: o driver é um driver de dispositivo e a pilha gerencia o barramento.
- Protocolos de alto nível: inclui o protocolo IEC-61883 que gerencia a comunicação, definindo os padrões de sinais para dados de áudio e vídeo. O protocolo IDB-1394, desenvolvido para aplicações automotivas, fica nesta camada. Ele controla os estados de “sleep mode” e “wake-up” dos dispositivos na rede.
- Camada de comunicação: exerce a função de camada de transporte automotiva e fornece interface para outros protocolos, como, por exemplo, Bluetooth.

A configuração da rede se dá em três passos:

- Inicialização do barramento: todas as informações de topologia de barramento de todos os nodos são apagadas. Cada nodo somente sabe se isso é uma folha ou um ramo.
- Identificação da árvore: cada nodo determina sua conexão a um nó pai ou a um nó filho. O nó que está somente conectado ao nó filho é o nó raiz.
- Auto identificação: todos os nós solicitam auto identificação de seus nós filhos. Nesta fase, cada nó obtém o endereço no barramento.

A transmissão de mensagens é brevemente descrita por SANTOS (2010) da seguinte maneira:

Em geral, o IEEE-1394 é baseado em TDM (Time Division Multiplex) e um período de ciclo leva 125 μ s. O nodo raiz começa um ciclo com o envio de um pacote de inicialização no início de cada ciclo para sincronizar os nodos. Até 80% do total do ciclo pode ser usado para transmissão síncrona. No restante do tempo é possível a transmissão assíncrona. Um nodo que pretende enviar pacotes síncronos deve reservar um canal síncrono e com a largura de banda necessária.
(SANTOS, 2010, p.180)

Para que um nodo transmita mensagens no espaço de tempo assíncrono, ele deve enviar um pedido para seu nodo raiz. Assim, o nodo raiz informa a todos outros nodos filhos esta ação e os impede de transmitir neste período, deixando o barramento livre para o nodo que requisitou

a transmissão. Então, o nodo que requisitou a transmissão, envia a mensagem pelo barramento.

Dentre os recursos que o IDB-1394 propicia dentro de um veículo, destacam-se:

- Transporte simultâneo de três diferentes fontes de vídeo.
- Descompactação e transmissão de imagens digitais de uma câmera traseira para assistência de estacionamento.
- Conexão de sistema de navegação e acesso plug and play para dispositivos da Apple (iPod, iPhone etc.).
- Acesso para múltiplas telas do tipo touch screen.

7.3 MOST (Media Oriented System Transport)

A tecnologia MOST (Media Oriented System Transport) é mantida por uma associação de empresas, a qual iniciou seu desenvolvimento em 1997. Tal tecnologia é utilizada atualmente pela BMW em alguns de seus modelos.

Tal protocolo se destaca pelo baixo custo, comunicação eficiente, com boa taxa de transferência de dados, da ordem de 25 Mbps.

Trata-se de uma rede plug and play, ou seja, com escalabilidade e capaz de gerenciar até 64 nodos, com camada física baseada em fibra óptica plástica (POF). Seu desenvolvimento objetivou a aplicação para dispositivos multimídia automotivo, com a intenção de interligar os mais variados sistemas dentro de um veículo: navegação, rádio digital, displays digitais, celulares, leitores de CD e DVD etc.

A tecnologia MOST baseia-se em uma rede síncrona. O mestre fornece o clock para todos os outros dispositivos na rede, para que os mesmos sincronizem suas operações.

Há dois tipos de canais na rede: de dados e de controle. Os canais de controle são utilizados para definir quais canais de dados serão utilizados no envio e recepção de informações. Assim que uma comunicação é estabelecida, as informações podem fluir pelo canal de dados escolhido.

Um ponto importante da rede MOST são as APIs (Application Programming Interface – Interface de Aplicação de Programação) presentes nos dispositivos. São elas que permitem uma padronização na implementação do protocolo, permitindo que dispositivos de fabricantes diferentes possam interagir entre si. Assim, independente dos dispositivos utilizados, é possível obter o sucesso na conexão entre os diversos sistemas multimídia do veículo, uma vez que todos utilizam a mesma API. Isto simplifica a utilização do protocolo MOST, permitindo a utilização completa das funcionalidades de cada dispositivo.

Em relação à transmissão de áudio e vídeo através do barramento MOST, até 15 canais de áudio estéreo não comprimido podem ser enviados pela rede, ou 15 canais de MPEG1 de áudio e vídeo. Vídeos de alta resolução não comprimidos ainda não podem ser enviados através de uma rede MOST.

A rede MOST tem seu uso bastante grande num ambiente automotivo devido a algumas características específicas:

- Capacidade de tratar aplicações com taxas de transferência baixa até aplicações com taxas da ordem de 25 Mbps;
- Elevado grau de integridade de dados e baixo jitter;
- Suporte a transferências de dados síncronas e assíncronas;
- Suporte de vários mestres;
- Suporte para mais de 64 dispositivos;
- Transmissão simultânea de múltiplos fluxos de dados, como informações de controle e pacotes de tempo real.

8 REDES SEM FIO

As aplicações de redes sem fio têm-se tornado cada vez mais comuns em diversas áreas, como educação, telefonia, aviação, militar, médica, entre outras.

E no meio automotivo não é diferente. É cada vez maior a necessidade de se conectar dispositivos portáteis pessoais aos sistemas multimídias do veículo, como MP3 players, telefones celulares, GPS, smartphones, entre outros.

A seguir são apresentados dois protocolos elaborados a partir das tecnologias de rede sem fio para aplicação automotiva, o Bluetooth: o UWB e o ZigBee.

8.1 Bluetooth

O Bluetooth foi originado em 1994 quando a Ericsson Mobile Communications viu a necessidade de se ter um mecanismo de comunicação sem fio de baixo custo e baixo consumo de energia para interconectar telefones e acessórios.

Assim, com a cooperação de mais quatro empresas (Nokia, IBM, Intel e Toshiba), em 1998 a tecnologia Bluetooth foi divulgada ao público e foi formado o consórcio Bluetooth SIG (Special Interest Group), com o objetivo de expandir o conceito Bluetooth e estabelecer um novo padrão para a comunicação sem fio.

Para que a conexão entre dois dispositivos Bluetooth ocorra, é necessária a formação de uma rede Piconet (com até 8 dispositivos) e várias Piconets podem ser interligadas formando uma rede Scatternet. A faixa de frequência utilizada fica entre 2.4 GHz e 2.8 GHz, podendo haver congestionamento se vários dispositivos interagirem mutuamente. Para minimização deste problema, o dispositivo Bluetooth utiliza a técnica do salto de frequência de alargamento de banda, onde em uma única transmissão ele pode saltar de uma frequência para outra (mantendo-se na banda total do Bluetooth) até encontrar caminho livre para a transmissão dos dados. A FIGURA 83 ilustra esta técnica.

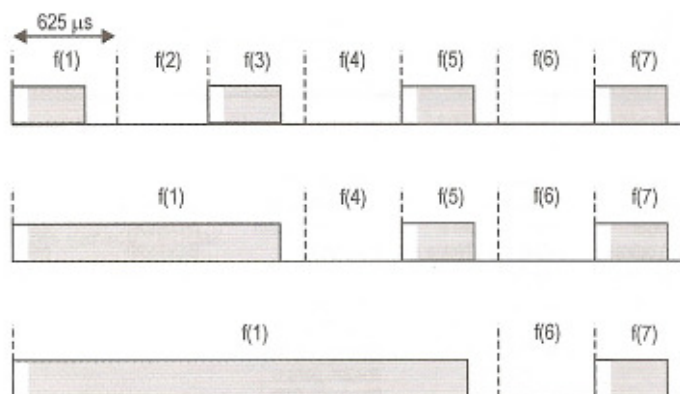


FIGURA 83: Salto de frequência em uma rede Bluetooth
Fonte: SANTOS (2010)

A tecnologia Bluetooth foi projetada para manipular dois tipos de sinais: sinais de dados e sinais de voz. Os sinais de dados utilizam a tecnologia de comutação de pacotes, onde os dados são divididos em pequenos pacotes antes de serem enviados. Quando os pacotes de dados de uma mensagem são recebidos por um dispositivo Bluetooth, este reconstitui a mensagem.

Já os sinais de voz utiliza a tecnologia de comutação de circuito, onde um canal dedicado é estabelecido durante a transmissão. A taxa de transferência total de dados (máxima) é de 1 Mbps, porém na prática a mesma é ligeiramente mais baixa. Para uma comunicação full-duplex, a taxa é de 432,6 kbps.

Existem dois estados de operação do Bluetooth: Connection e Standby. No estado Connection, dois dispositivos estão conectados entre si e executando alguma ação. Já no modo Standby os dispositivos estão conectados entre si, porém sem a presença de atividades entre eles. Neste último modo há uma economia de energia, pois o dispositivo monitora somente a cada 1,28 segundo se existe alguma mensagem oriunda do outro dispositivo.

Os dispositivos Bluetooth sempre iniciam no modo Standby e o mestre do sistema será aquele que reconheceu primeiramente o outro dispositivo. Assim, o dispositivo reconhecido será o escravo da conexão. O mestre então envia a solicitação de conexão ao escravo para reconhecimento do endereço. Reconhecido o endereço, o comando página é emitido pelo mestre, despertando o escravo e estabelecendo-se a conexão. Tal lógica é ilustrada na FIGURA 84.

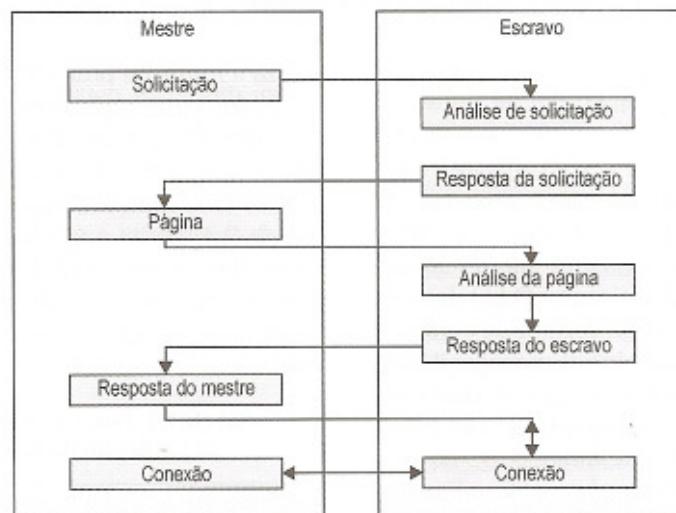


FIGURA 84: Fluxo de comandos para reconhecimento Bluetooth
Fonte: SANTOS (2010)

Assim, é estabelecida uma rede Piconet, podendo esta ser expandida a até oito dispositivos (um mestre e sete escravos). Para a formação de uma rede com mais de oito dispositivos, é preciso estabelecer a conexão entre várias Piconets (até 10), formando uma Scatternet, conforme mostra a FIGURA 85.

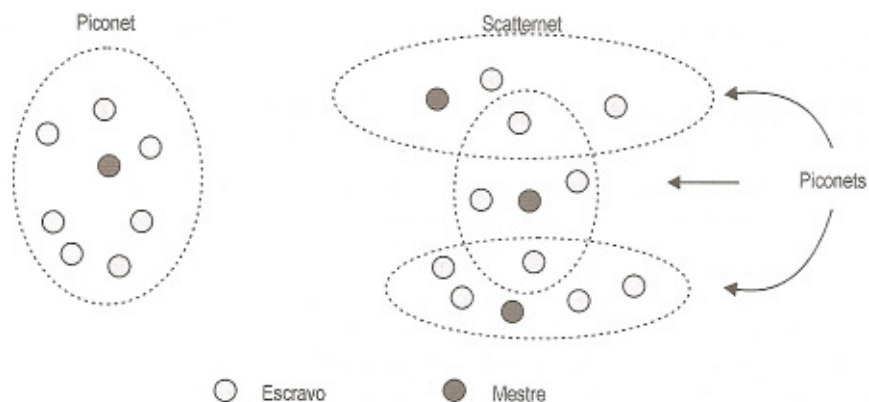


FIGURA 85: Configurações de redes Bluetooth: piconet e scatternet
Fonte: SANTOS (2010)

A segurança é um ponto importante a ser considerado na tecnologia Bluetooth, já que a mesma consiste em uma tecnologia de rede sem fio e administra e transporta informações pessoais. Existem três modos de segurança no Bluetooth:

- Modo de segurança 1: não há nenhum nível de segurança. O dispositivo fica totalmente aberto.

- Modo de segurança 2: o dispositivo inicia os procedimentos de segurança somente após o estabelecimento da conexão.
- Modo de segurança 3: é o modo mais seguro, onde os procedimentos de segurança são iniciados antes do estabelecimento da conexão.

8.2 UWB (Ultra Wide Band)

O protocolo de comunicação UWB (Ultra Wide Band) é uma tecnologia promissora para a aplicação em veículos. Sua alta taxa de transmissão com utilização de baixa potência o credenciam para isto.

De acordo com RIBEIRO (2008, p.9), o uso da tecnologia UWB em conjunto com outras redes é perfeitamente aplicável, podendo-se aproveitar da alta taxa de transmissão desta tecnologia para a transferência de informações de multimídia, como som e vídeo, entre os dispositivos da rede.

A aplicação da tecnologia UWB está para a comunicação sem fio assim como a rede CAN está para as redes cabeadas, porém seu uso ainda não foi validado no ambiente veicular, conforme explica RIBEIRO (2008):

Em suas devidas proporções, o UWB pode estar para as redes sem fio intra-veicular assim como o CAN Bus está para comunicações cabeadas. Os testes validaram seu uso em laboratório, mas apesar de vários autores defenderem que o UWB será o padrão que substituirá outras tecnologias, como o Bluetooth, nenhum teste em ambiente veicular real foi realizado.
(RIBEIRO, 2008, p.9)

A topologia vista na FIGURA 86, sugerida por XIANG (2007), apresenta uma aplicação da tecnologia UWB para o sistema de ABS. Nela, os quatro sensores de velocidade das rodas enviam as informações através do protocolo UWB para um coordenador (coordinator), que por sua vez converte os sinais para um padrão aceitável pelo barramento cabeado, sendo a informação transmitida para as outras ECUs.

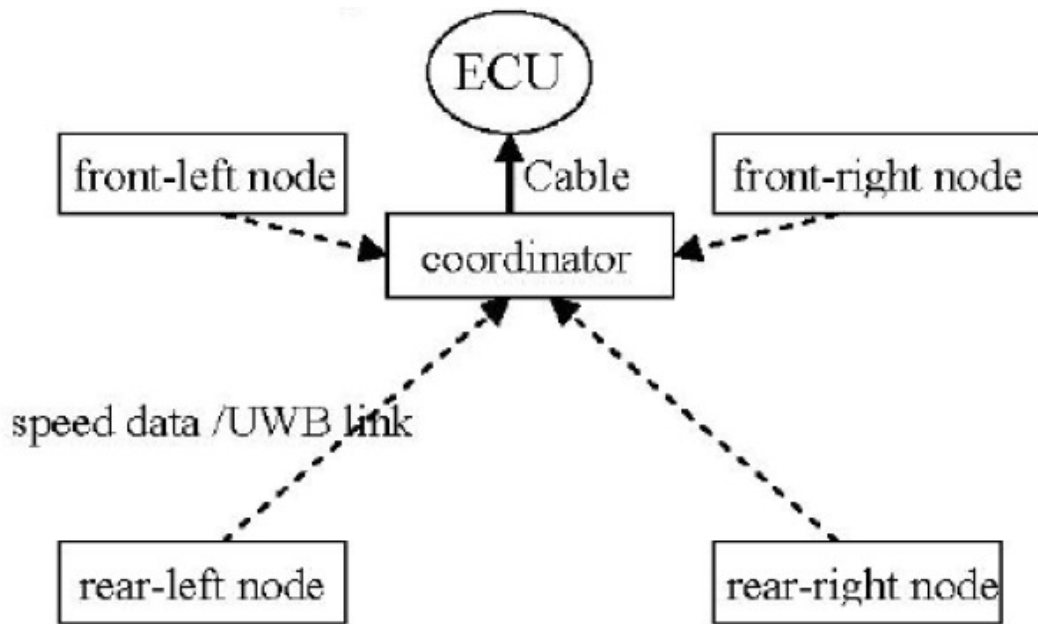


FIGURA 86: Topologia da aplicação de UWB para comunicação intra-veicular
Fonte: XIANG (2007)

8.3 ZigBee

A tecnologia ZigBee consiste em um protocolo de comunicação sem fio com baixa taxa de transmissão de dados e baixo custo, direcionado à comunicação de sensores e atuadores dentro de um veículo. As aplicações típicas do ZigBee são o monitoramento e o controle de funções que não exigem alta largura de banda, porém com exigências quanto ao consumo de energia e atraso.

A primeira especificação deste protocolo foi publicada em dezembro de 2004. A taxa de transmissão de dados pode chegar a até 250 kbps, o que é satisfatório para a interligação de dispositivos como controle de ar-condicionado, aquecimento, ventilação, iluminação etc.

Esta tecnologia diferencia-se das redes Bluetooth e Wi-Fi pelo consumo mais baixo de energia, alcance reduzido (até cerca de 10 metros) e comunicações entre dois dispositivos podem ser replicadas pelos dispositivos na rede até o destino final da mensagem.

9 O FUTURO DAS REDES DE COMUNICAÇÃO AUTOMOTIVAS

O número de funções dentro de um veículo, bem como a quantidade de sistemas eletrônicos em substituição a sistemas puramente hidráulicos e mecânicos têm a tendência em aumentar nos próximos anos.

Exemplos desta evolução poderão ser vistos em sistemas de segurança ativa, comunicação entre veículos, interação entre dispositivos pessoais (como smartphones) e sistemas do veículo etc.

Assim, a busca de tecnologias que suportem este crescimento e que sejam competitivas em custo será intensificada pelas montadoras de veículos, consolidando assim novas arquiteturas eletroeletrônicas automotivas.

Segundo pesquisa conduzida pelo CAR – Center for Automotive Research (2011, p.20), provavelmente ocorrerá uma maior aplicação de tecnologias de rede com taxas de transmissão de dados mais elevadas, devido à demanda da evolução dos sistemas eletrônicos dentro de um veículo. Por exemplo, é provável que o atual domínio do uso da rede CAN no meio automotivo abra espaço para protocolos como o FlexRay, o qual conta com uma maior taxa de transmissão de informações. O mesmo pode-se dizer da comunicação sem fio, onde a tecnologia UWB deve ser utilizada para comunicação de dispositivos dentro de um veículo.

Na FIGURA 87, verifica-se a tendência de utilização de novos sistemas eletrônicos automotivos, os quais demandarão a aplicação de tecnologias de rede automotivas diferentes das que são mais utilizadas atualmente.

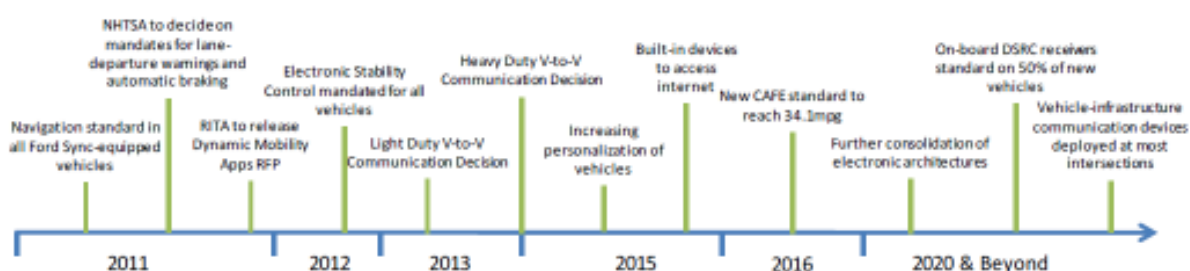


FIGURA 87: Linha do tempo para a evolução dos sistemas eletrônicos automotivos
Fonte: CAR (2011)

10 CONCLUSÃO

O objetivo deste trabalho foi alcançado, pois o estudo realizado comprovou a importância da escolha correta do modelo de arquitetura eletroeletrônica e dos protocolos de comunicação durante o projeto de um veículo.

As características de cada tipo de arquitetura, bem como suas vantagens e desvantagens, foram analisadas. Para os veículos atuais (e também futuros), fica clara a superioridade do tipo de arquitetura distribuída, pois esta tem uma flexibilidade para adição de novas funções ao veículo. E isto é de extrema importância, já que cada vez mais o mercado exige um maior conteúdo tecnológico embarcado nos automóveis, a fim de atender a demanda para que novas funções sejam agregadas aos veículos.

Sobre os protocolos de comunicação, existe um grande número de opções para as diversas aplicações. E estas opções foram criadas, na sua grande maioria, a partir de necessidades específicas que as empresas enfrentaram no tocante a comunicação de ECUs dentro de determinados sistemas automotivos. Assim, tornaram-se disponíveis diversas tecnologias de redes apropriadas para cada tipo de aplicação, adequadas para os diferentes grupos de funções presentes em um automóvel (powertrain, chassis, entretenimento, segurança crítica etc.). Ainda que para os veículos produzidos no Brasil exista uma maior concentração de utilização do protocolo CAN, é possível verificar-se várias aplicações de outros protocolos mais específicos em arquiteturas mais complexas, como o FlexRay, ByteFlight, MOST e Bluetooth.

No que diz respeito às tendências para o futuro, este estudo mostrou que será inevitável a utilização de diferentes protocolos de comunicação para os diversos sistemas eletrônicos do veículo. Ainda, a aplicação de protocolos de comunicação com taxas de transmissão de dados superiores aos protocolos atualmente utilizados será de extrema importância, devido ao aumento da quantidade de informações a serem transmitidas e dos requisitos de maior velocidade para a transmissão destes dados no futuro.

REFERÊNCIAS

CAR – CENTER FOR AUTOMOTIVE RESEARCH. **Automotive technology: greener vehicles, changing skills – electronics, software and controls report**. Ann Arbor: Driving Change Project, 2011. 31p.

GUIMARÃES, A. de A. **Arquiteturas eletroeletrônicas - “Conceituação”**. Disponível em: < http://www.alexag.com.br/CAN_Bus_Parte_1.html >. Acesso em: 02 mar. 2012.

GUIMARÃES, A. de A. **Eletrônica embarcada automotiva**. São Paulo: Érica, 2011. 326p.

MAHMUD, S. M.; ALLES, S. **In-vehicle network architecture for the next-generation vehicles**. In: SAE WORLD CONGRESS, 2005-01-1531, 2005, Detroit: SAE Technical Paper Series, 2005. 12p.

NAVET, N.; LION, F. S. **Trends in automotive communication systems**. Nancy: Embedded Systems Handbook: Networked Embedded Systems – 2nd ed. Taylor and Francis / CRC Press (Ed.) (2009) 13.1-13.24, 2009. 40p.

RIBEIRO, M. P. N. **Introduzindo tecnologia sem fio em redes intra-veiculares**. Salvador, 2008. 11p.

SANTOS, M. M. D. **Redes de comunicação automotiva – características, tecnologias e aplicações**. São Paulo: Érica, 2010. 220p.

XIANG, W. **A vehicular ultra wide band channel model for future wireless Intra-Vehicle Communications (IVC) systems**. In: VEHICULAR TECHNOLOGY CONFERENCE, 2007. VTC-2007 Fall. 2007 IEEE 66th, Baltimore: IEEE Xplore Digital Library, 2007. p.2159-2163.