

Arquitetura de Comunicação Automotiva Aplicada para Veículos

Matheus Nunes Franco¹, Eryk K. Suguiura¹, Gian Ricardo Berkenbrock¹

¹Laboratório de Sistemas Embarcados
Universidade Federal de Santa Catarina
Joinville, Santa Catarina, Brasil

mnunesfranco@gmail.com, gian.rb@ufsc.br, erykksuguiura@hotmail.com

Abstract. *Efficient communication between vehicle modules is essential for optimizing performance and safety. This work proposes a communication architecture in AADL for terrestrial, naval, and aerial vehicle systems. Prototypes were tested using the OSATE platform, focusing on bus budgeting and energy consumption. The results confirm the effectiveness of the developed solution, which employs CAN and Ethernet communication protocols for interaction between ECUs. This approach provides a solid foundation for future advancements in vehicular communication.*

Resumo. *A comunicação eficiente entre os módulos de um veículo é essencial para otimizar desempenho e segurança. Este trabalho propõe uma arquitetura de comunicação em AADL para sistemas veiculares terrestres, navais e aéreos. Protótipos foram testados usando a OSATE, com foco em orçamento de barramentos e consumo energético. Os resultados confirmam a eficácia da solução desenvolvida, que emprega os protocolos de comunicação CAN e Ethernet para a interação entre ECUs. Essa abordagem fornece uma base sólida para futuros avanços em comunicação veicular.*

1. Introdução

Veículos modernos integram diversos componentes eletrônicos, sensores, atuadores e unidades de controle, organizados em arquiteturas que gerenciam funções essenciais como navegação, entretenimento, conforto e telemetria. A comunicação eficiente entre esses componentes depende de uma arquitetura de comunicação adequada, crucial para a transmissão de dados.

Este trabalho visa analisar e comparar modelos de arquiteturas de comunicação para veículos terrestres, navais e aéreos, utilizando a *Open Source AADL Tool Environment* (OSATE), com ênfase em dois protocolos de comunicação, *Controller Area Network* (CAN) e Ethernet. O foco está em avaliar a eficiência energética, a responsividade do sistema e a viabilidade de futuras expansões e atualizações tecnológicas. A metodologia inclui a modelagem de diferentes topologias de rede e protocolos de comunicação, seguidas por análises teóricas e simulações. O objetivo é fornecer uma documentação abrangente para apoiar futuros desenvolvimentos e integrações no sistema de veículos.

2. Fundamentação Teórica

Para a estruturação desta seção, são apresentados os conceitos de arquitetura de software veicular, abordagens de projeto de arquiteturas automotivas, protocolos de comunicação, linguagens de descrição para arquiteturas e plataformas de desenvolvimento.

2.1. Arquitetura de Software

A arquitetura de software é fundamental para a estruturação de sistemas robustos e escaláveis, especialmente em ambientes desafiadores como os de veículos terrestres, navais ou aéreos. Esses sistemas precisam atender a requisitos de segurança e operar em tempo real, enfrentando condições adversas, como alta pressão e corrosão. Nesse contexto, a robustez da arquitetura é crucial para garantir o funcionamento confiável dos veículos [Malaterre 1998].

2.1.1. Arquitetura de Software Veicular

Historicamente, a arquitetura veicular *Offene Systeme und deren Schnittstellen für die Elektronik in Kraftfahrzeugen* (OSEK) foi amplamente utilizada devido à sua simplicidade para microcontroladores com recursos limitados e sem gerenciamento de memória. No entanto, a *Automotive Open System Architecture* (AUTOSAR) evoluiu a partir do OSEK, oferecendo uma abordagem modular e adaptável às novas tecnologias de interconectividade automotiva [Lee et al. 2013]. A AUTOSAR destaca-se por permitir a integração e reutilização de componentes de software entre diferentes fabricantes e modelos, promovendo padronização na indústria [Staron 2021].

A arquitetura AUTOSAR é composta por três camadas principais:

- *Application Software* (ASW): Gerencia os componentes de aplicação [Fürst and Bechter 2016].
- *Realtime Environment* (RTE): Facilita a comunicação entre as camadas através de interfaces e buffers de tempo real [Fürst and Bechter 2016].
- *Basic Software* (BSW): Gerencia o acesso ao hardware, serviços de criptografia e *drivers* [Fürst and Bechter 2016].

Essa estrutura em camadas garante escalabilidade, compatibilidade com diversos sistemas e protocolos, e alta eficiência energética, além de facilitar a implementação, compreensão e manutenção.

2.2. Arquiteturas Centralizadas vs Distribuídas

Segundo [Guimarães and Saraiva 2003], as arquiteturas eletroeletrônicas veiculares podem ser centralizadas ou distribuídas. As arquiteturas centralizadas concentram o processamento em unidades de controle principais, enquanto as distribuídas distribuem o processamento entre vários módulos. Cada abordagem tem vantagens e desvantagens em termos de desempenho e complexidade [Malaterre 1998].

A escolha entre as duas arquiteturas depende das características do projeto. Para sistemas com baixo custo e recursos limitados, a arquitetura centralizada é mais adequada. Em contraste, para maior compatibilidade com aprimoramentos futuros e aplicações diversas, a arquitetura distribuída é preferível [Mahmud and Alles 2005]. Embora a arquitetura centralizada seja simples e de baixo custo, ela não será explorada neste trabalho devido à sua dificuldade em suportar desenvolvimentos futuros, justamente pela existência do controlador não-modular único [Souza and Campos 2017].

2.3. Protocolos de Comunicação

Os protocolos definem a especificação e formatação dos dados trocados entre unidades de controle, sensores e atuadores em veículos [Forouzan 2012]. Os principais protocolos de comunicação utilizados em sistemas veiculares e analisados neste projeto são o *Controller Area Network (CAN)* e o *Ethernet*.

2.3.1. Controller Area Network (CAN)

O CAN é um protocolo de comunicação amplamente utilizado em sistemas automotivos para comunicação robusta entre módulos eletrônicos [Souza and Campos 2017]. Utiliza uma arquitetura de barramento compartilhado, permitindo que múltiplos dispositivos transmitam e recebam mensagens simultaneamente. Seu mecanismo de detecção e correção de erros garante confiabilidade e comunicação eficiente em aplicações críticas [Forouzan 2012].

2.3.2. Ethernet

Desenvolvido inicialmente para redes de computadores, o Ethernet foi adaptado para o setor automotivo, oferecendo flexibilidade e alta velocidade na transmissão de dados [Forouzan 2012]. Ele facilita a integração de diversos dispositivos e sistemas no veículo, permitindo o gerenciamento de informações de entretenimento e sistemas avançados de assistência ao motorista. No entanto, desafios como garantir a segurança da rede, gerenciar o consumo de energia e lidar com cabeamento volumoso precisam ser enfrentados [Forouzan 2012].

3. Metodologia

A metodologia adotada neste trabalho seguiu um processo estruturado, com foco na modelagem, implementação e análise de arquiteturas de comunicação veicular. O desenvolvimento foi conduzido a partir de uma série de etapas interligadas, abrangendo o escopo do trabalho, definição de requisitos, planejamento da arquitetura, modelagem de ECUs e cenários de comunicação, além da implementação e avaliação dos protótipos.

3.1. Escopo de trabalho

A implementação para este trabalho foi definida, para a modelagem e prototipagem de duas arquiteturas de comunicação, mais um módulo de telemetria simplificado. Essas restrições permitiram a implementação de modelos menos sofisticados, mas reduzindo o escopo suficientemente para permitir o desenvolvimento do trabalho.

3.2. Requisitos de Arquitetura

Os requisitos foram estabelecidos para garantir um desempenho satisfatório em veículos terrestres, navais e aéreos, e para permitir a adição de módulos futuros. Estes requisitos são:

- **Eficiência Energética:** Veículos modernos, operando em ambientes sem conexão constante à energia elétrica, exigem alta eficiência energética para sua arquitetura de comunicação.

- **Implementação e Compreensibilidade:** O sistema projetado deve ser simples de entender, reproduzir e modificar, uma vez que o trabalho foca em apenas uma parte da comunicação dos veículos.
- **Escalabilidade:** O sistema precisa ser capaz de lidar com o aumento de dispositivos conectados. A plataforma OSATE permite a análise de fluxo de dados e eventos, garantindo a escalabilidade necessária.
- **Compatibilidade com Outros Sistemas:** A comunicação dos veículos deve ser compatível com outros sistemas, permitindo a intercomunicação entre diferentes dispositivos e protocolos.

3.3. Escolha da Topologia de Rede e Protocolos de Comunicação

A topologia de rede define a estrutura e a conexão entre dispositivos em sistemas de comunicação. A arquitetura centralizada oferece menor escalabilidade, enquanto a distribuída melhora a redundância e a tolerância a falhas. Nesta pesquisa, a arquitetura distribuída foi escolhida por proporcionar maior confiabilidade e facilitar futuras atualizações e adaptações.

Discerniu-se a compatibilidade da rede de comunicação específica para veículos como um amálgama entre funções de diagnóstico do sistema e funções relacionadas à eficiência ou dinâmica do veículo, com uma subcategoria voltada para a comunicação sem fio. As primeiras funções são centralizadas no protocolo de comunicação CAN, enquanto a última utiliza Ethernet. Como parte do processo de construção desse perfil, foi necessário realizar o levantamento e a análise dos microcontroladores que compõem as ECUs.

3.4. Critérios de Análise

Os critérios de análise, apresentados na Tabela 1 seguem alguns padrões de métrica básicos, descritos por [Bochmann and Sunshine 1980], e foram utilizados para desenvolver a arquitetura de comunicação veicular, com as análises realizadas na plataforma OSATE.

3.5. Plataformas e Ferramentas de Desenvolvimento

Dada a natureza deste projeto, que envolve controle de orçamento de transmissão de dados, a plataforma OSATE foi selecionada por oferecer uma visão abrangente da arquitetura veicular. A ferramenta permite extrair métricas essenciais, como o número de componentes e conectores, facilitando a avaliação da complexidade e interconexões do sistema.

Como o OSATE é uma plataforma de código aberto para a *Architecture Analysis and Design Language* (AADL), esta também será utilizada no desenvolvimento deste trabalho. A AADL, uma linguagem padronizada na engenharia de sistemas, especialmente em sistemas críticos de tempo real e embarcados [Feiler and Gluch 2012], permite a modelagem e análise da estrutura, comportamento e propriedades de sistemas complexos.

3.6. Modelagem das ECUs

A modelagem das Unidades de Controle Eletrônico (ECUs) foi realizada utilizando a plataforma OSATE, com base na linguagem AADL (*Architecture Analysis Design Language*). A AADL permite a modelagem integrada de hardware e software, possibilitando uma análise detalhada dos componentes e suas interações.

Tabela 1. Métricas e suas descrições

Métrica	Descrição
Número de componentes	A medida básica que quantifica o tamanho da arquitetura em termos de seu edifício básico (blocos ou componentes).
Número de conectores	A medida básica que quantifica a conectividade da arquitetura em termos de sua base de conectores.
Número de ECUs	A medida básica que quantifica o tamanho da arquitetura física em termos de unidades de processamento.
Número de tipos de comunicação	Quantifica a complexidade da arquitetura em termos da necessidade de implementar múltiplas comunicações e mecanismos.
Número de interfaces externas	Quantifica o acoplamento entre componentes da arquitetura e sistemas externos.
Número de componentes concorrentes	A medida conta os componentes em que cálculos simultâneos fazem parte de seu comportamento.

Um dos principais componentes modelados foi a ECU, responsável pelo "Central Gateway", que atua na intercomunicação entre diferentes subsistemas veiculares, garantindo interoperabilidade e segurança de dados. Adicionalmente, foi implementada a ECU de "Gateway de Conectividade", responsável pela integração de diferentes tecnologias de comunicação, como Wi-Fi e Bluetooth, assegurando conectividade externa.

Além dessas unidades centrais, será modelado o módulo de telemetria, o qual desempenha um papel importante na coleta e transmissão de dados operacionais dos veículos.

3.7. Modelagem e Análise dos Cenários de Comunicação

Com a modelagem das ECUs generalizadas, foi possível iniciar o processo de desenvolvimento dos dois diferentes cenários que fazem parte do objetivo principal deste trabalho. A implementação dos modelos para os cenários 1 e 2 com a plataforma OSATE foi realizada de maneira simplificada para facilitar sua realização dentro do escopo deste trabalho.

3.7.1. Cenário 1:

Neste primeiro cenário serão implementados, o *Gateway Central*, *Gateway de Conectividade* e Módulo simplificado de telemetria. Os barramentos para comunicação entre os *gateways* e a comunicação do *Gateway de Conectividade* com o módulo de telemetria utilizam o protocolo CAN como base.

Para implementação, foi utilizada a biblioteca *mcp-can.h*, permitindo a criação de barramentos CAN operando a 8 MHz, com uma taxa de transmissão de 500 kb/s. A mensagem enviada não foi estritamente formulada. Para fins de comparação, o módulo de telemetria verifica a conexão *Wi-Fi* e retorna uma mensagem de 24 bytes.

O fluxo de mensagens foi implementado de modo que o *Gateway Central* espera por uma interrupção do *Gateway de Conectividade* usando o *Barramento-CAN-1**. O

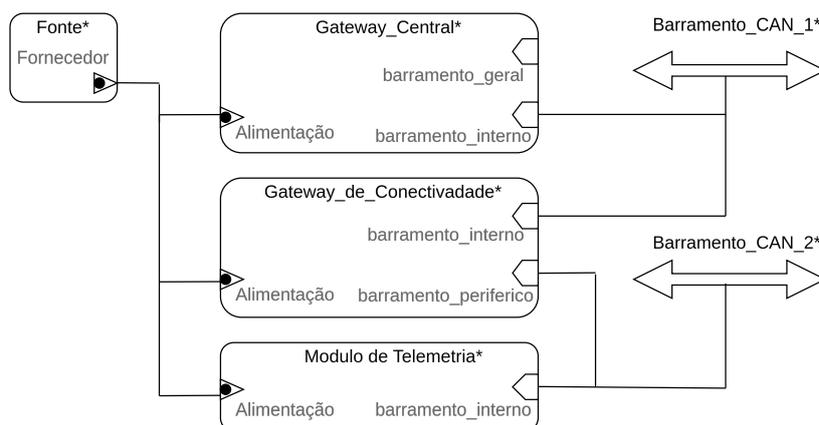


Figura 1. Modelo do cenário 1.

Gateway de Conectividade envia uma mensagem de solicitação e espera uma resposta a cada 100 milissegundos para o módulo de telemetria pelo *Barramento-CAN-2**. Esse verifica se a conexão *Wi-fi* foi estabelecida e retorna pelo *Barramento-CAN-2** a mensagem de 24 bytes mencionada, representando o estado da conexão. Quando essa mensagem alcança o *Gateway Central*, o dado é repassado para o *Gateway Central* via *Barramento-CAN-1** e o fluxo é repetido.

3.7.2. Cenário 2:

No segundo cenário, a comunicação entre o *Gateway Central* e o *Gateway de Conectividade* foi modelada utilizando o protocolo Ethernet. O barramento entre o *Gateway de Conectividade* e o módulo de telemetria permanece o mesmo do cenário 1, utilizando o protocolo CAN. A escolha do protocolo Ethernet entre as duas ECUs foi feita devido à alta capacidade de transmissão de dados desse protocolo, considerado interessante por ser o primeiro gargalo nesse cenário.

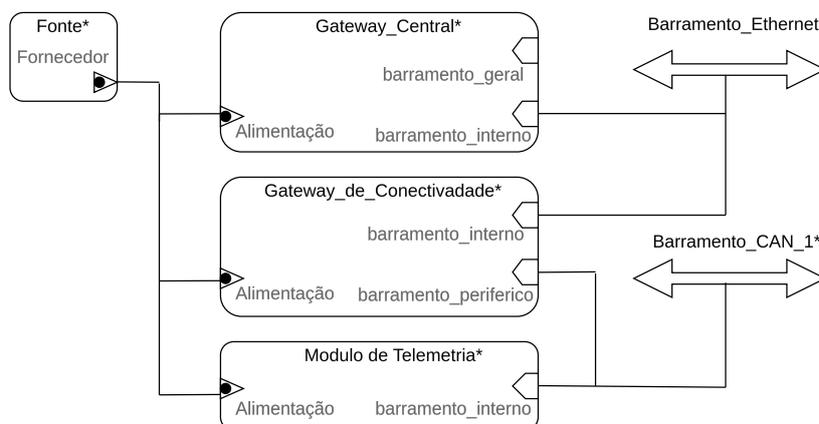


Figura 2. Modelo do cenário 2.

O fluxo de mensagens seguiu essencialmente, o mesmo do cenário 1. O *Gateway-Central* espera por uma mensagem do *Gateway de Conectividade*, neste cenário pelo *Barramento-Ethernet**. As mensagens entre *Gateway de Conectividade* e o módulo de

telemetria seguem o mesmo padrão do cenário 1. Quando a mensagem alcança o *Gateway de Conectividade*, o dado é repassado para o *Gateway Central* via *Barramento-Ethernet** e o fluxo é repetido.

3.8. Etapa de implementação

A escolha dos materiais neste estudo priorizou praticidade e disponibilidade. Optou-se pelo microcontrolador Arduino UNO e pelo módulo Wi-Fi ESP8266 devido à sua compatibilidade com a plataforma selecionada. O módulo MCP2515 TJA1050 foi escolhido para barramento CAN, oferecendo um controlador CAN de alta velocidade e um transceptor integrado em uma solução compacta para comunicação automotiva.

O Arduino Ethernet Shield foi utilizado para adicionar conectividade Ethernet ao Arduino UNO, suportando velocidades de até 100 Mbps. Para os testes de potência, utilizou-se a Fonte Digital PS-4000 da ICEL, que permite ajuste preciso de tensão e corrente, facilitando a análise do consumo de energia necessário para complementar os modelos na OSATE.

4. Resultados e Discussões

Para a implementação dos modelos, optou-se por focar em dois modelos específicos para a arquitetura de comunicação veicular. O primeiro modelo, demonstrado na Fig.3, adota a abordagem do cenário 1, na qual as ECUs se comunicam exclusivamente por meio de barramentos CAN.

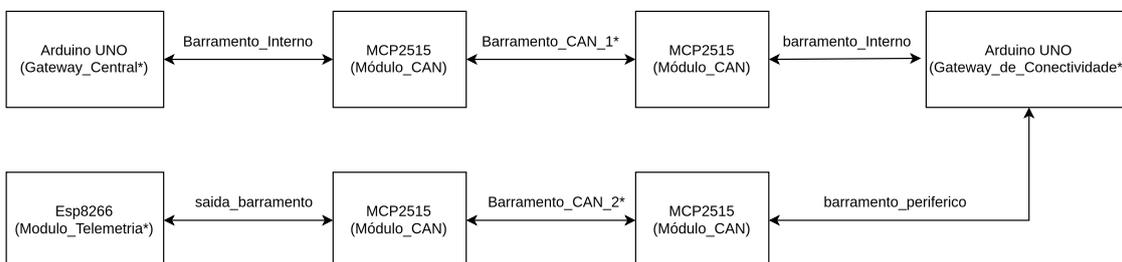


Figura 3. Fluxograma da modelagem com o protocolo CAN em todos os barramentos.

O segundo modelo, representado na Fig 4 incorpora um barramento CAN conectado a módulos periféricos, enquanto as ECUs principais responsáveis pela telemetria se comunicam mediante um barramento Ethernet, como proposto no cenário 2.

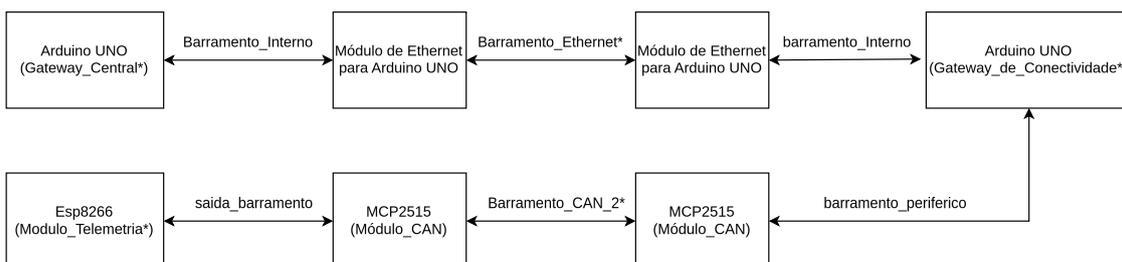


Figura 4. Fluxograma da modelagem com o protocolo Ethernet entre "Gateways".

Essa escolha simplificou as comparações, reduzindo as variáveis ao mínimo, permitindo uma observação do impacto da arquitetura selecionada, em detrimento das características específicas do hardware. Isso se alinha à flexibilidade da plataforma de desenvolvimento OSATE, que facilita a adaptação das propriedades da plataforma sem a necessidade de alterações físicas em diversos hardwares.

4.1. Análise dos modelos

As análises foram feitas usando a ferramenta interna do OSATE, que gera automaticamente planilhas com informações relevantes a análise escolhida pelo usuário. O foco deste trabalho se deu em duas análises, de orçamento, que verificam os aspectos de uso, dos barramentos de comunicação, retornando porcentagem em uso, possibilidade de erro, entre outras características.

O segundo tipo de análise é o de potência elétrica, que retorna informações relevantes a alimentação e gasto de energia dos componentes, mostrando individualmente o uso de cada componente do sistema, permitindo verificar se o sistema de alimentação elétrica do projeto permite a adição de novos módulos, ou se é possível rearranjar componentes já existentes.

4.2. Comparação dos modelos propostos

O processo de escolha do modelo arquitetural na presente pesquisa foi orientado pela busca por valores ótimos em todas as métricas apresentadas. Ressaltando que, nas métricas discutidas, um resultado bom é representado por valores menores, com exceção da capacidade de banda larga. Sendo assim, a seleção do modelo final foi conduzida levando em consideração a minimização de todas as métricas em questão. Este critério de escolha assegura que o modelo escolhido seja aquele que, de maneira geral, apresenta a menor quantidade em todas as dimensões avaliadas, consolidando assim uma abordagem eficiente e equilibrada para a arquitetura de comunicação veicular implementada.

As métricas selecionadas na Tabela 2 implicam em diferentes fatores, que providenciariam os pesos de cada uma das métricas em análises, esses fatores não aparecem diretamente na literatura por serem valores específicos de cada projeto. Descrevendo as mais importantes para este trabalho, o número de componentes indica a quantidade de componentes físicos presentes na implementação do modelo, demonstrando uma quantificação do custo monetário necessário para a implementação. O número de conectores permite uma visão da complexidade de implementação, pois quanto maior a quantidade de conexões, mais complexo o sistema se torna, e o número de tipos de comunicação tem uma ligação direta com a complexidade de implementação do software.

Das métricas incorporadas pela Tabela 3, todas permitem uma análise direta com valores mostrados. Com esse entendimento, o custo baixo, a facilidade de implementação, eficiência energética e confiabilidade são as características de maior relevância. Neste caso, o modelo com todos os barramentos usando o protocolo CAN se mostrou a melhor escolha entre os modelos comparados, possuindo menos componentes, número de tipos de comunicação, gasto em potência e custo total de banda larga.

5. Limitações

A implementação de protótipos adaptados apresenta limitações que afetam a fidelidade do modelo e restringem a profundidade das análises. A simplicidade da comunicação entre

Tabela 2. Métricas e suas descrições para CAN e Ethernet

Métrica	CAN	Ethernet
Número de componentes	11	12
Número de conectores	15	13
Número de ECUs	2	2
Número de tipos de comunicação	1	2
Número de interfaces externas	2	2
Número de componentes concorrentes	1	1

Tabela 3. Métricas energéticas para CAN e Ethernet

Métrica	CAN	Ethernet
Potência máxima fornecida (W)	40	40
Gasto em potência (mW)	980	1694
Capacidade de banda larga (KB/s)	62,5	100000
Custo de banda larga (KB/s)	44,6	42,6
Custo no barramento virtual (KB/s)	0,475	1,150

os módulos, com mensagens pequenas e barramentos dedicados a apenas dois módulos, limita a representatividade em relação a sistemas automotivos mais complexos.

Embora a literatura sugira melhorias com o uso de Ethernet e maior volume de mensagens ou dispositivos conectados, a ausência de módulos adicionais e o tamanho restrito das mensagens usadas nos testes não permitiram avaliar essas melhorias. Além disso, a limitada quantidade de métricas usadas na análise prejudicou a avaliação completa de aspectos como escalabilidade, confiabilidade e eficiência, o que dificultou a comparação de diferentes modelos de arquitetura no contexto de veículos.

6. Conclusão

Neste pesquisa, o foco foi compreender e aplicar arquiteturas de redes de comunicação, com o objetivo principal de definir um projeto específico de comunicação veicular. Esse objetivo foi alcançado por meio do desenvolvimento de modelos, implementação de protótipos, comparação de métricas e análise dos dados obtidos.

É importante notar que os modelos escolhidos são os mais viáveis dentro do escopo do projeto, considerando limitações de tempo, recursos e complexidade técnica, e não necessariamente os mais eficazes em todos os cenários. Embora os modelos selecionados sejam válidos, abordagens mais completas poderiam extrapolar o âmbito deste trabalho.

Conseguimos diferenciar arquiteturas de comunicação automotivas, apresentando suas classificações e topologias, o que fundamentou o desenvolvimento dos modelos e cenários. A modelagem foi facilitada pelo uso da linguagem AADL, que permite representar a estrutura, comportamento e propriedades dos componentes em diferentes camadas de abstração.

A implementação foi realizada com protótipos que possibilitam a comunicação entre controladores e um módulo simplificado de telemetria. Embora a implementação

idealmente abrangeria partes estruturais, eletrônicas e de software completas, o trabalho foi concluído com uma implementação funcional. A plataforma OSATE forneceu métricas para comparar os modelos descritos em AADL, considerando os requisitos definidos na fase de planejamento.

Apesar das limitações, estabelecemos uma base para futuros projetos de pesquisa em arquiteturas de comunicação automotiva, com análises e métricas que oferecem uma compreensão inicial e protótipos que servem como ponto de partida para desenvolvimentos futuros.

Referências

- Bochmann, G. and Sunshine, C. (1980). Formal methods in communication protocol design. In *IEEE Transactions on Communications*, volume 28, pages 624–631.
- Feiler, P. and Gluch, D. (2012). *Model-Based Engineering with AADL: An Introduction to the SAE Architecture Analysis Design Language*. Carnegie Mellon University, Software Engineering Institute's Digital Library.
- Forouzan, B. A. (2012). *Data Communications and Networking Global Edition 5e*. McGraw Hill.
- Fürst, S. and Bechter, M. (2016). Autosar for connected and autonomous vehicles: the autosar adaptive platform. In *Proceedings of the 2016 46th annual IEEE/IFIP international conference on dependable systems and networks workshop (dsn-w)*, pages 215–217.
- Guimarães, A. D. A. and Saraiva, A. M. (2003). Um roteiro de implementação de uma rede can (controller area network). In *CONFERÊNCIA INTERNACIONAL DE ENGENHARIA AUTOMOTIVA-SIMEA*, pages 40–43.
- Lee, Y. H., Kim, J. H., and Jeon, J. W. (2013). Applying autosar network management in osek/vdx for compatibility of autosar and osek/vdx. In *Proceedings of the FISITA 2012 World Automotive Congress*, pages 693–704.
- Mahmud, S. M. and Alles, S. (2005). In-vehicle network architecture for the next-generation vehicles. *SAE Transactions*, 114(7):466–475.
- Malaterre, P. (1998). Automotive trends: rationale and motivations. In *IEE Seminar on OSEK/VDX Open Systems in Automotive Networks (Ref. No. 1998/523)*, London, United Kingdom.
- Souza, A. G. and Campos, G. L. (2017). Rede can veicular: levantamento bibliográfico e apresentação de conceitos iniciais. *ForScience*, 5(1):122–153.
- Staron, M. (2021). *Automotive Software Architectures*. Springer.