

Fundamentos de

CAN, J1939 e ISOBUS

Do bit ao diagnóstico em tratores
e implementos agrícolas

ISO 11783

0CF00400

1CEBFF00

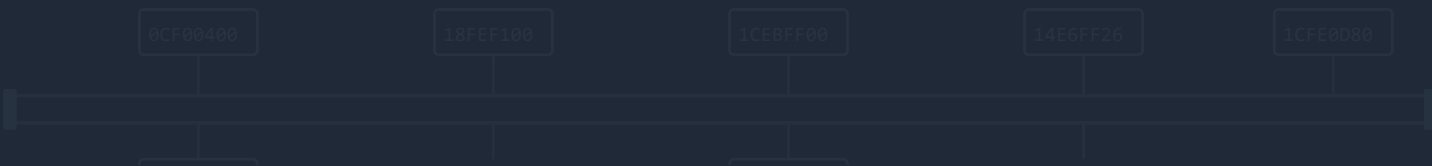
18FEF100

1CFE0D80

Fundamentos de CAN, J1939 e ISOBUS

Do bit ao diagnóstico em tratores e implementos agrícolas

Anderson R. Costa



Do bit dominante no par trançado à interface de um implemento agrícola no Universal Terminal: um percurso completo, progressivo e prático pelos protocolos que sustentam a comunicação entre funções do trator e implementos agrícolas.

PARTE I

CAN em máquinas agrícolas

Camada física, quadro, arbitragem e tratamento de erros

PARTE II

SAE J1939

ID de 29 bits, PGN, SPN, Address Claim e transporte

PARTE III

ISOBUS / ISO 11783

UT/VT, Object Pool, Task Controller, TECU e Working Set

PARTE IV

Diagnóstico e análise

DTC, DM1/DM2, estudos de caso e leitura de trace

OBRA DE REFERÊNCIA TÉCNICA

Fundamentos de CAN, J1939 e ISOBUS

Do bit ao diagnóstico em tratores e implementos agrícolas

Anderson R. Costa

1ª edição — 2026
Maringá/PR · Edição do autor

Um percurso de engenharia que parte do barramento físico de dois fios e chega à interface de um implemento agrícola exibida no terminal da cabine, passando por quadros CAN, arbitragem, identificadores de 29 bits, PGNs, SPNs, gerenciamento de rede, protocolos de transporte, serviços ISOBUS e diagnóstico.

CONTEÚDO

- 4 partes e 22 capítulos
- Diagramas técnicos originais
- Exemplos didáticos de quadros decodificados
- Exercícios com gabarito comentado
- Estudos de caso de diagnóstico
- Glossário e tabelas de referência

COMO O MATERIAL ESTÁ ORGANIZADO

- Parte I — CAN em máquinas agrícolas
- Parte II — SAE J1939
- Parte III — ISOBUS / ISO 11783
- Parte IV — Diagnóstico e análise
- Apêndices — Gabarito, referência rápida, glossário, notas complementares, referências e índice remissivo

DADOS DA PUBLICAÇÃO	
Título	Fundamentos de CAN, J1939 e ISOBUS
Subtítulo	Do bit ao diagnóstico em tratores e implementos agrícolas
Autor	Anderson R. Costa (nome completo: Anderson Roberto da Costa)
Edição	1ª edição — 2026
Ano de publicação	2026
Local de publicação	Maringá/PR
Editora	Edição do autor
Titular do ISBN	Anderson Roberto da Costa
ISBN	978-65-02-40444-7
Formato	Livro digital em PDF, A4 retrato
Idioma	Português do Brasil
Direitos autorais	© 2026 Anderson Roberto da Costa. Todos os direitos reservados.

Dados Internacionais de Catalogação na Publicação (CIP)
(Câmara Brasileira do Livro, SP, Brasil)

Costa, Anderson R.

Fundamentos de CAN, J1939 e ISOBUS [livro eletrônico] : do bit ao diagnóstico em tratores e implementos agrícolas / Anderson R. Costa. -- 1. ed. -- Maringá, PR : Ed. do Autor, 2026.

PDF

Bibliografia.

ISBN 978-65-02-40444-7

1. Agricultura 2. Eletrônica 3. Padrão ISOBUS (ISO 11783) 4. Padrão SAE J1939 5. Sistemas embarcados - Redes CAN 6. Tratores agrícolas - Máquinas e equipamentos - Redes de comunicação I. Título.

26-400755.1

CDD-629.225

Índices para catálogo sistemático:

1. Tratores agrícolas : Tecnologia 629.225

Aline Grazielle Benitez - Bibliotecária - CRB-1/3129

Nenhuma parte desta obra pode ser reproduzida, armazenada ou transmitida, por qualquer meio, sem autorização prévia e por escrito do autor (Lei nº 9.610, de 19 de fevereiro de 1998).

Sobre esta obra

Este livro foi escrito como obra de referência e material de treinamento para engenheiros, desenvolvedores de software embarcado, técnicos, estudantes e profissionais de máquinas agrícolas que estejam começando a trabalhar com CAN clássico, SAE J1939 em tratores e ISOBUS na integração de implementos.

Esta obra foi desenvolvida com base em anotações de estudos e treinamentos técnicos, aliadas à pesquisa aprofundada em normas, especificações, literatura especializada e documentação técnica relacionada a CAN, SAE J1939 e ISOBUS / ISO 11783. O conteúdo integra fundamentos teóricos e aplicações práticas em uma abordagem progressiva, voltada à compreensão dos protocolos e de seu emprego em tratores e implementos agrícolas. A terminologia e os parâmetros técnicos seguem as referências normativas adotadas.

Sobre o autor

Anderson R. Costa é desenvolvedor de *software* com trajetória profissional iniciada em 1997, acumulando ampla experiência em desenvolvimento de sistemas, *firmware*, eletrônica e sistemas embarcados.

Ao longo de sua carreira, trabalhou com diferentes gerações de tecnologias e linguagens de programação, evoluindo do desenvolvimento tradicional de *software* para aplicações de baixo nível, microcontroladores e sistemas embarcados aplicados principalmente à automação agrícola e industrial.

Sua experiência inclui desenvolvimento em C/C++, arquiteturas de microcontroladores, *firmware*, integração entre *hardware* e *software*, sistemas de tempo real e implementação de protocolos de comunicação.

Nos últimos anos, tem dedicado especial atenção às tecnologias de comunicação utilizadas em máquinas e sistemas embarcados, com estudos e trabalhos envolvendo CAN, SAE J1939 e ISOBUS (ISO 11783), além de outros protocolos empregados em sistemas distribuídos e automação.

Também desenvolve ferramentas, bibliotecas, simuladores e projetos experimentais relacionados à eletrônica e aos sistemas embarcados, mantendo uma rotina contínua de estudo, pesquisa de normas, documentação técnica e experimentação prática.

Este livro nasce da combinação dessa experiência profissional com anos de estudos, treinamentos, pesquisa técnica e análise de normas e especificações, tendo como objetivo apresentar os fundamentos de CAN, SAE J1939 e ISOBUS de maneira didática, tecnicamente consistente e próxima das aplicações encontradas no mundo real.

Aviso sobre normas, marcas e direitos

As normas SAE J1939 e ISO 11783 (bem como a série ISO 11898) são documentos protegidos por direitos autorais e comercializados pelas respectivas organizações. Esta obra **não reproduz** o texto dessas normas: os conceitos são explicados com linguagem própria, e valores, códigos e requisitos aparecem resumidos, com a indicação da norma e da cláusula de origem, na medida necessária ao estudo. A consulta aos documentos oficiais permanece indispensável para qualquer trabalho de implementação, homologação ou certificação. Os diagramas, tabelas, exemplos e procedimentos têm finalidade didática e de apoio à integração, à análise e ao diagnóstico de redes; não substituem os requisitos de projeto de produto, as dimensões mecânicas de conectores, os ensaios elétricos, ambientais e de robustez nem a avaliação formal de conformidade.

ISOBUS é a designação associada à série ISO 11783. A AEF (Agricultural Industry Electronics Foundation) promove sua implementação e mantém diretrizes de funcionalidades e um programa de certificação. Nomes de produtos e de empresas citados ao longo do texto — ferramentas de análise, bibliotecas, interfaces de hardware — pertencem aos seus respectivos detentores e são mencionados apenas em caráter informativo, sem qualquer vínculo, endosso ou representação. Este material não utiliza logotipos, identidade visual ou material gráfico de fabricantes: toda a linguagem visual, incluindo os diagramas, é original. As ilustrações conceituais geradas por IA para esta obra estão identificadas nas legendas.

As informações desta obra foram preparadas com cuidado e conferidas nas fontes citadas, mas têm finalidade didática e não dispensam, em cada aplicação, a consulta às normas vigentes e à documentação do fabricante do equipamento. Medições e intervenções em máquinas devem ser feitas por pessoal qualificado, com a máquina em condição segura e de acordo com as instruções do fabricante e as normas de segurança aplicáveis. A aplicação destas informações em projeto, instalação, manutenção ou certificação é de responsabilidade de quem a realiza.

As capturas de barramento, valores de identificadores, quadros e telas apresentados são **exemplos didáticos**, construídos para serem tecnicamente coerentes, e não registros de um equipamento específico.

Convenções numéricas

- Valores hexadecimais aparecem com o prefixo `0x` ou em fonte monoespaçada: `0x0CF00400`, `FF 7D AF E0`.
- Valores binários aparecem com o sufixo `b` ou agrupados em blocos: `011 0 0 11110000 00000100 00000000`.
- PGNs são apresentados em hexadecimal e, quando útil, também em decimal.
- Bytes de um quadro CAN são numerados de **1 a 8**, seguindo a convenção usada na literatura J1939 (o byte 1 é o primeiro transmitido).

- Bits dentro de um byte são numerados de 1 (menos significativo) a 8 (mais significativo) quando a numeração SAE for relevante; nos diagramas de identificador, os bits são indicados explicitamente.
- A separação decimal é a **vírgula**, e o separador de milhar é o espaço: 1 785 bytes, 0,58 ms, 117 440 505 bytes.
- Nas capturas de barramento, as marcas de tempo seguem o formato das ferramentas de análise, em segundos e com ponto decimal: **0.148** corresponde a 0,148 s.
- Unidades de taxa aparecem como **kbit/s** e **Mbit/s**. Unidades de tamanho seguem o SI: **kB** vale 1 000 bytes e **KiB** vale 1 024 bytes.

Convenções de vocabulário

- **Quadro** designa a unidade do CAN — o que trafega no barramento, com SOF, arbitragem, dados, CRC e EOF. *Frame* aparece apenas nos rótulos de ferramentas e em nomes consagrados, como *Frame Example*.
- **Mensagem** designa a unidade do nível de aplicação: um grupo de parâmetros completo, que pode ocupar um quadro ou centenas deles, quando é segmentado por um protocolo de transporte.
- **Taxa de bits** designa a velocidade do barramento, em kbit/s. O livro não usa "*baud rate*": no CAN, os dois conceitos coincidem numericamente, mas não são a mesma grandeza.
- Onde o inglês *actual* significa "medido" ou "efetivo", o livro usa **efetivo** — a taxa efetivamente aplicada, e não a "taxa atual".
- Siglas são escritas por extenso na primeira ocorrência de cada parte e reunidas no Apêndice C.

NOTA

Ao longo do livro, a sigla **VT** (Virtual Terminal) e a sigla **UT** (Universal Terminal) referem-se ao mesmo componente: VT é o termo usado na norma ISO 11783-6, enquanto UT é o nome da funcionalidade correspondente no vocabulário da AEF. O texto usa VT/UT de forma intercambiável, preferindo VT quando o assunto é o protocolo e UT quando o assunto é a funcionalidade do ponto de vista do usuário.

Como usar este livro

Este é um livro de progressão deliberada: cada capítulo assume apenas o que foi construído nos anteriores. Ele também foi pensado para permanecer útil depois do treinamento, como referência de consulta rápida na bancada, no campo ou diante de uma captura de barramento.

Objetivos do treinamento

Ao concluir o material, o participante deve ser capaz de:

- Explicar por que redes seriais substituíram a fiação ponto a ponto em veículos e máquinas, e onde CAN, J1939 e ISOBUS se encaixam nessa evolução.
- Descrever a camada física do CAN — par trançado, transmissão diferencial, estados dominante e recessivo, terminação de 120 Ω — e relacionar taxa de bits com comprimento de barramento.
- Identificar cada campo de um quadro CAN, nos formatos padrão (11 bits) e estendido (29 bits), e explicar o mecanismo de arbitragem bit a bit.
- Interpretar os mecanismos de detecção de erro do CAN e os estados *error active*, *error passive* e *bus off*.
- Decompor um identificador J1939 de 29 bits em prioridade, páginas de dados, PDU Format, PDU Specific e Source Address, e calcular o PGN correspondente.
- Converter valores brutos de um quadro em grandezas físicas a partir da resolução, do offset e da faixa de um SPN, tratando corretamente os valores especiais.
- Explicar o NAME de 64 bits e acompanhar, passo a passo, um processo de *Address Claim*, inclusive em situação de conflito.
- Reconhecer, em uma captura, uma transferência BAM, CMTD, ETP ou Fast-Packet e saber qual delas é adequada a cada situação.
- Descrever a arquitetura de uma rede ISOBUS e o papel de TECU, UT/VT, TC, AUX, FS e do conceito de Working Set.
- Acompanhar o carregamento de um Object Pool e a execução de uma tarefa de taxa variável, do mapa de prescrição ao mapa *as applied*.
- Ler um DTC (SPN, FMI, OC), entender DM1/DM2/DM3 e conduzir um diagnóstico em camadas, do cabo ao serviço de aplicação.
- Observar uma janela de *trace* e responder metodicamente: o que está sendo transmitido, quem transmitiu, para quem, qual PGN, quais dados e por onde começar a análise.

PÚBLICO-ALVO

- Engenheiros de aplicação, de testes e de sistemas
- Desenvolvedores de software embarcado
- Técnicos de manutenção e de campo
- Estudantes de engenharia e de cursos técnicos
- Profissionais que trabalham com máquinas e implementos agrícolas
- Quem está começando agora em CAN, J1939 e ISOBUS

PRÉ-REQUISITOS

- Eletrônica básica: tensão, corrente, resistência, terra
- Noção de sinal digital e de níveis lógicos
- Numeração binária, decimal e hexadecimal
- Conceito de bit, byte e de ordem de bytes
- Familiaridade com o ambiente de um trator ou implemento

Não é pré-requisito conhecer CAN: o protocolo é construído do zero a partir do Capítulo 2.

A trilha de aprendizagem

O livro segue uma única linha de raciocínio, em quatro etapas. Cada estação da trilha depende da anterior e prepara a seguinte; nenhum conceito é usado antes de ser apresentado (Figura 1).

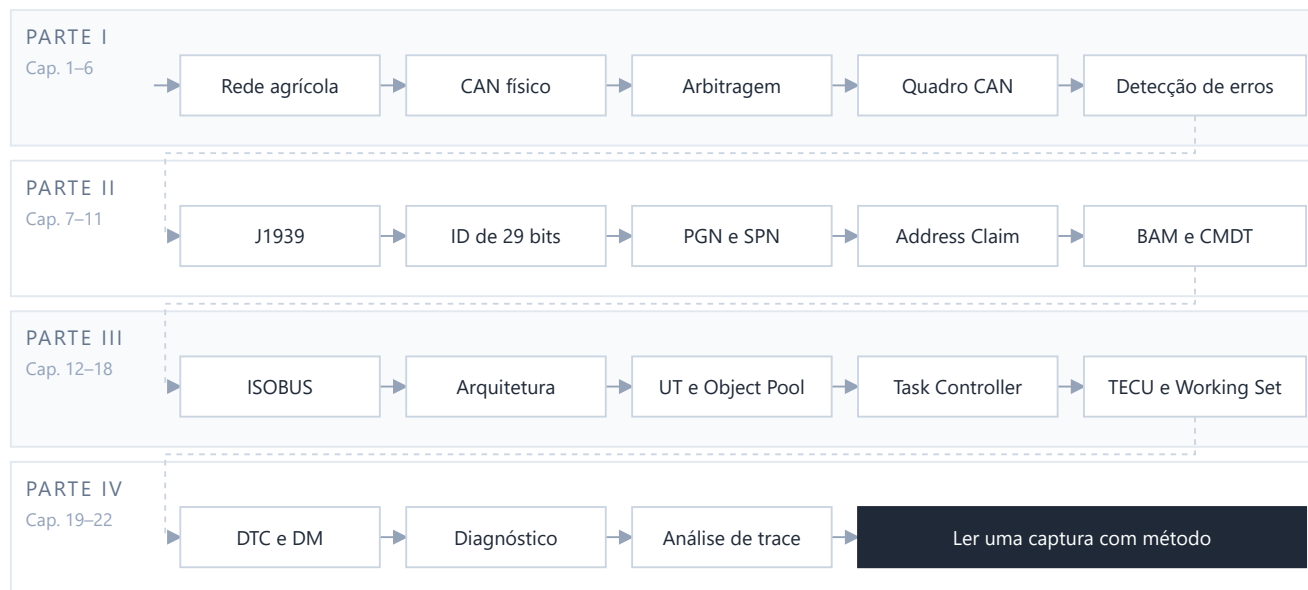


Figura 1 — Trilha de aprendizagem do livro: do conceito de rede agrícola à leitura metódica de uma captura CAN/J1939/ISOBUS. As setas tracejadas indicam a continuidade entre as partes.

A estrutura de cada capítulo

Os capítulos técnicos seguem sempre o mesmo ritmo, de modo que o leitor saiba onde procurar cada tipo de informação:

1. **Objetivo e conceitos do capítulo** — o que será aprendido, em uma caixa destacada logo após o título.
2. **Conteúdo** — explicação curta, seguida de representação visual, e não o contrário.
3. **Exemplo e decodificação** — sempre que houver uma mensagem envolvida, ela aparece como quadro completo e é interpretada campo a campo.
4. **Aplicação prática e diagnóstico** — o que o conceito significa diante de uma máquina, de uma bancada ou de uma ferramenta de análise.
5. **Exercícios** — ao final dos capítulos principais, com gabarito comentado no Apêndice A.
6. **Resumo** — os pontos que precisam ser levados para o capítulo seguinte.

As caixas didáticas

Oito tipos de caixa se repetem ao longo do livro, sempre com o mesmo significado:

CONCEITO

Uma definição que precisa ficar retida. É o vocabulário mínimo para continuar lendo o capítulo.

EXEMPLO

Aplicação imediata do que acabou de ser explicado, normalmente com números.

FRAME EXAMPLE

CAN ID

DLC

DATA

0x0CF00400

8

FF 7D AF E0 2E 00 FF FF

Uma mensagem CAN/J1939 **construída para ensino**, tecnicamente coerente, sempre acompanhada da sua decomposição e do seu significado. Nenhum quadro aparece no livro sem ser interpretado, e nenhum deles é registro de um equipamento comercial específico.

COMO DECODIFICAR

Procedimento passo a passo, na ordem em que deve ser executado. É o tipo de caixa que se consulta com a captura aberta na frente.

ATENÇÃO

Erro frequente, armadilha de interpretação ou detalhe que costuma custar horas de depuração.

DIAGNÓSTICO

Procedimento de investigação: o que medir, o que observar e em que ordem.

NA PRÁTICA

A relação entre o conceito e o que se vê em um veículo, em uma máquina agrícola ou em uma ferramenta de análise.

NOTA

Complemento de escopo ou aprofundamento técnico, identificado como **nota complementar**; esclarecimentos sobre pontos que costumam gerar confusão aparecem como **nota técnica**.

Sugestão de aplicação em treinamento

A Tabela 1 sugere como distribuir o conteúdo em um treinamento presencial ou remoto.

Tabela 1 — Distribuição sugerida para um treinamento presencial ou remoto.

Módulo	Capítulos	Carga	Atividade prática sugerida
CAN em máquinas agrícolas	1 a 6	6 h	Medição de terminação e de nível diferencial; leitura de quadros em ferramenta de análise
SAE J1939	7 a 11	7 h	Decodificação manual de identificadores; observação de um <i>Address Claim</i> na inicialização
ISOBUS / ISO 11783	12 a 18	8 h	Acompanhar o carregamento de um Object Pool e uma tarefa do Task Controller
Diagnóstico e análise	19 a 22	5 h	Interpretação de DM1; estudos de caso do Capítulo 22

A carga total sugerida é de aproximadamente 26 horas, sem contar a resolução dos exercícios. Para turmas sem experiência prévia com barramentos, recomenda-se reservar tempo adicional nos Capítulos 4, 5 e 8, que concentram a maior densidade conceitual.

Sumário

Os títulos são clicáveis na versão digital. Figuras e tabelas são numeradas por capítulo (Figura 5.3 = terceira figura do Capítulo 5).

Como usar este livro	5
PARTE I · CAN EM MÁQUINAS AGRÍCOLAS	
1 Da fiação ponto a ponto à rede da máquina	14
O problema que deu origem ao barramento	14
De sinais individuais a uma rede	15
O que é uma rede embarcada	15
A pilha: CAN, J1939 e ISOBUS	16
Onde cada protocolo aparece na máquina agrícola	17
O que este livro vai construir	17
Exercícios	17
2 Fundamentos do CAN	19
O que é CAN	19
Características principais	19
Orientação a mensagens: a ideia central	20
Os componentes de um nó CAN	21
O CAN no modelo ISO/OSI	22
Versões do CAN	22
Vantagens do CAN	23
Exercícios	23
3 A camada física do CAN	25
Por que transmissão diferencial	25
Os dois estados do barramento	26
Uma sequência de bits no osciloscópio	27
O barramento como “E lógico com fio”	27
Terminação, impedância e reflexões	27
Taxa de bits e comprimento do barramento	28
Cabo, trança e derivações	29
Tempo de bit e ponto de amostragem	29
Exercícios	31
4 Arbitragem: como o barramento decide quem fala	33
O problema: todos podem falar	33
A regra da arbitragem	33
Exemplo completo com três ECUs	34
Por que a arbitragem é não destrutiva	35
Prioridade, latência e projeto da rede	35
Exercícios	35
5 A estrutura do quadro CAN	37
Os quatro tipos de quadro	37
O quadro de dados no formato padrão	38
O quadro de dados no formato estendido	39
Exemplos de quadros	39
Quadro de erro e quadro de sobrecarga	40
Quanto tempo ocupa um quadro	40
Exercícios	42
6 Detecção de erros e confinamento de falhas	43
Cinco mecanismos complementares	43
Bit stuffing em detalhe	44
Os cinco erros, um a um	44
O quadro de erro	46

Confinamento de falhas: contadores e estados	47
Como o erro chega até você	48
Do sintoma à causa	49
Modos de falha da camada física	49
Três falhas, do sintoma ao conserto	50
Retransmissão automática	53
Exercícios	53

PARTE II · SAE J1939

7	Introdução ao SAE J1939	56
	O que o J1939 acrescenta	56
	Características principais	57
	A arquitetura das normas	57
	Broadcast e ponto a ponto	58
	O vocabulário que vem a seguir	59
	Exercícios	59
8	O identificador J1939 de 29 bits	61
	Os seis campos	61
	Do hexadecimal aos campos	62
	A prioridade: quem escolhe e quanto custa errar	62
	PDU1 e PDU2: o limiar dos 240	63
	Decodificação completa, passo a passo	65
	O caminho inverso: montar o identificador	65
	A cadeia completa de interpretação	66
	PGNs frequentes	66
	Exercícios	67
9	Parameter Groups, PGN e SPN	68
	Três siglas, três níveis	68
	Características dos Parameter Groups	68
	Anatomia de um SPN	69
	Ordem dos bytes	70
	Valores especiais	70
	Decodificação completa de um quadro	70
	SPNs frequentes	72
	Exercícios	72
10	Gerenciamento de rede e Address Claim	74
	Duas identidades por função de controle	74
	Os endereços	74
	O NAME de 64 bits	76
	Quem é quem: Vehicle System e Function	77
	O processo de Address Claim	78
	Um conflito, passo a passo	80
	Mensagens auxiliares de gerenciamento	80
	Request e Acknowledgment	81
	Exercícios	82
11	Protocolos de transporte: BAM e CMDT	84
	O problema dos oito bytes	84
	Os dois PGNs de transporte	85
	BAM — Broadcast Announce Message	86
	CMDT — Connection Mode Data Transfer	87
	Os tempos que governam uma transferência	88
	Comparação e critérios de escolha	89
	Quando 1 785 bytes não bastam	89
	Perspectivas: outras evoluções do CAN	90
	Exercícios	90

PARTE III · ISOBUS — ISO 11783

12	Introdução ao ISOBUS / ISO 11783	93
	O que é ISOBUS	93
	Objetivos	94
	Características principais	94
	Histórico	94
	A norma ISO 11783 em partes	94
	As funcionalidades e a certificação	95
	As cinco diferenças em relação ao J1939	98
	Exercícios	100
13	Arquitetura de uma rede ISOBUS	101
	Dois barramentos, dois mundos	101
	Quando o implemento tem uma rede CAN própria	102
	Como interligar segmentos: repetidor, ponte, roteador e gateway	103
	A camada física ISOBUS	104
	O conector IBBC	105
	Uma palavra sobre segurança	108
	Topologia dinâmica	109
	Exercícios	109
14	Universal Terminal e Virtual Terminal	111
	O que é o Virtual Terminal	111
	Uma tela ISOBUS por dentro	113
	Como a interface chega ao terminal	114
	Duas mensagens que explicam o resto	114
	A interação durante a operação	115
	Exercícios	116
15	Object Pool: a interface do implemento	118
	O que é um Object Pool	118
	Catálogo de objetos	119
	Como os objetos se relacionam	120
	Softkeys e máscaras de alarme	121
	Tamanho, memória e tempo de carregamento	121
	Ferramentas de criação	122
	Exercícios	122
16	Task Controller e agricultura de precisão	123
	O que é o Task Controller	123
	As funcionalidades do Task Controller	124
	Funcionalidades	124
	DDOP: como o implemento se descreve	125
	DDI: o dicionário de dados	126
	A conversa entre o TC e o cliente	127
	Do mapa de prescrição ao mapa as applied	128
	Controle automático de seções	129
	O arquivo de tarefa	131
	Exercícios	133
17	Tractor ECU, Working Set e serviços auxiliares	135
	A Tractor ECU	135
	Working Set	138
	Controle auxiliar — AUX	139
	Comando de Linguagem — quem decide km/h ou mph	140
	Solicitação de Taxa de Repetição	141
	File Server	142
	Sequence Control	142
	Exercícios	142
18	Transporte no ISOBUS: ETP e Fast-Packet	144

Por que dois protocolos a mais	144
ETP — Extended Transport Protocol	144
Fast-Packet	146
Os quatro protocolos lado a lado	147
Recomendações de uso	147
Exercícios	148

PARTE IV · DIAGNÓSTICO E ANÁLISE

19 Diagnóstico J1939: DTC e mensagens DM	150
O código de falha	150
Os códigos FMI	151
Decodificação de um exemplo de DTC	152
As lâmpadas	153
A família de mensagens DM	153
O ciclo de vida de uma falha	154
Exercícios	155
20 Diagnóstico ISOBUS	156
O que muda em relação ao J1939	156
Tratamento de falhas	159
Diagnóstico pelo Virtual Terminal	159
Diagnóstico em camadas	160
Ferramentas	161
Exercícios	161
21 Análise prática de capturas	162
O que uma ferramenta de análise faz	162
A janela de trace	164
Filtros e agrupamentos	165
Janelas interativas de VT e TC	165
Método: seis perguntas para ler qualquer captura	166
Leitura completa de uma inicialização	166
Laboratório: seguir um comando entre redes	167
Onde a análise se encaixa no desenvolvimento	168
Exercícios	168
22 Estudos de caso	170
Caso 1 — Uma ECU não aparece na rede	170
Caso 2 — Duas ECUs com o mesmo Source Address	172
Caso 3 — O implemento não aparece no terminal	174
Caso 4 — A seção fecha tarde na cabeceira	175
Padrões que se repetem	176
Exercícios	176
A Gabarito comentado	178
Capítulo 1 — Da fiação ponto a ponto à rede da máquina	178
Capítulo 2 — Fundamentos do CAN	178
Capítulo 3 — A camada física do CAN	179
Capítulo 4 — Arbitragem	179
Capítulo 5 — A estrutura do quadro CAN	179
Capítulo 6 — Detecção de erros	180
Capítulo 7 — Introdução ao SAE J1939	181
Capítulo 8 — O identificador de 29 bits	181
Capítulo 9 — PGN e SPN	182
Capítulo 10 — Gerenciamento de rede e Address Claim	182
Capítulo 11 — Protocolos de transporte	183
Capítulo 12 — Introdução ao ISOBUS	184
Capítulo 13 — Arquitetura de uma rede ISOBUS	184
Capítulo 14 — Universal Terminal	185
Capítulo 15 — Object Pool	185

Capítulo 16 — Task Controller	185
Capítulo 17 — TECU, Working Set e serviços auxiliares	186
Capítulo 18 — Transporte no ISOBUS	187
Capítulo 19 — Diagnóstico J1939	187
Capítulo 20 — Diagnóstico ISOBUS	188
Capítulo 21 — Análise prática de capturas	188
Capítulo 22 — Estudos de caso	189
B Referência rápida	190
Camada física CAN e ISOBUS	190
Decodificação do identificador J1939	191
Conversão de parâmetros	191
PGNs essenciais	192
Bytes de controle dos protocolos de transporte	192
Endereços e NAME	193
SPNs e DDIs de uso frequente	193
Diagnóstico	194
Roteiro de diagnóstico em seis camadas	194
Funcionalidades AEF e o que elas envolvem	194
C Glossário	195
Conceitos gerais	195
Camada física e componentes	195
Quadro CAN	196
Estrutura de mensagens J1939	196
Gerenciamento de rede	197
Protocolos de transporte	197
Componentes e serviços ISOBUS	197
Diagnóstico	198
D Notas complementares	199
Aprofundamentos técnicos	199
Como manter as informações atualizadas	200
E Referências e leituras recomendadas	201
Referências	201
Situação das normas centrais	204
Panorama das famílias normativas	205
Edições consultadas	206
Organizações e documentação pública	208
Critério de seleção das fontes	208
Origem do conteúdo deste livro	208
Ficha técnica desta edição	208
F Listas e índice remissivo	210
Lista de figuras	210
Lista de tabelas	211
Índice remissivo	214

PARTE I



CAN em máquinas agrícolas

A primeira parte constrói o alicerce. Ela parte do problema industrial que deu origem aos barramentos — chicotes que cresciam com cada nova função da máquina — e chega ao ponto em que o leitor consegue olhar para um quadro CAN e nomear cada um dos seus campos, entender quem ganha o barramento quando dois nós transmitem ao mesmo tempo e reconhecer como a rede reage a um erro.

- 1 Da fiação ponto a ponto à rede da máquina
- 2 Fundamentos do CAN
- 3 A camada física do CAN
- 4 Arbitragem: como o barramento decide quem fala
- 5 A estrutura do quadro CAN
- 6 Detecção de erros e confinamento de falhas

01

Da fiação ponto a ponto à rede da máquina

OBJETIVO DO CAPÍTULO

Entender por que veículos e máquinas deixaram de trocar sinais por fios dedicados e passaram a usar um barramento serial compartilhado, e situar CAN, SAE J1939 e ISOBUS/ISO 11783 como camadas sucessivas de um mesmo edifício técnico.

CONCEITOS QUE SERÃO APRENDIDOS

- Expansão da eletrônica embarcada e crescimento do número de ECUs
- Comunicação ponto a ponto e o crescimento do chicote elétrico
- Conceito de barramento e de rede embarcada
- Comunicação orientada a mensagens, e não a destinatários
- A relação CAN → J1939 → ISOBUS e o que cada camada acrescenta
- Domínios de aplicação na arquitetura trator-implemento

O problema que deu origem ao barramento

A mecanização agrícola passou a depender de unidades de controle eletrônico para motor, transmissão, hidráulica, direção e operação dos implementos. A ECU do motor calcula rotação e torque; outras unidades precisam dessas grandezas para coordenar suas funções. Sem uma rede, cada informação compartilhada exigiria condutores dedicados no chicote. É nesse cenário que o CAN resolve a integração entre ECUs do trator e do implemento.

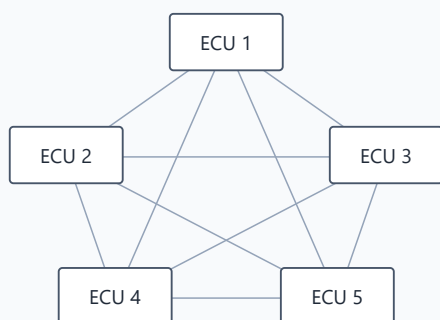
A primeira solução foi a mais direta: **um fio dedicado para cada sinal**. Cada informação que precisasse sair de uma ECU e chegar a outra ganhava seu próprio condutor, com seu próprio conector e sua própria entrada no chicote elétrico. Essa abordagem funciona para poucos sinais e se torna insustentável rapidamente, porque o número de ligações possíveis cresce com o quadrado do número de participantes (Figura 1.1).

CONCEITO

Comunicação ponto a ponto: cada par de dispositivos que precisa trocar uma informação é ligado por um canal físico exclusivo. Para n dispositivos totalmente interligados, o número de ligações é $n \times (n - 1) / 2$ — 10 ligações para 5 ECUs, 45 para 10 ECUs, 190 para 20 ECUs.

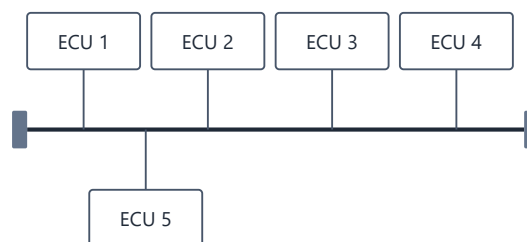
PONTO A PONTO

5 ECUs · 10 ligações dedicadas



BARRAMENTO SERIAL

5 ECUs · 1 par trançado compartilhado



As terminações nas extremidades serão estudadas no Capítulo 3.

Figura 1.1 — O mesmo conjunto de cinco ECUs ligado de duas formas. À esquerda, cada par de participantes exige um canal físico próprio; à direita, todos compartilham um único meio e as mensagens circulam por ele.

O resultado da abordagem ponto a ponto eram chicotes de fios **complexos, pesados e caros** — difíceis de montar na linha de produção, difíceis de diagnosticar em campo e sujeitos a falhas em cada conector adicional. A solução para esse dilema veio na forma de **comunicação serial por um único canal compartilhado**, o barramento, que levou ao desenvolvimento do protocolo CAN no início da década de 1980, pela Bosch.

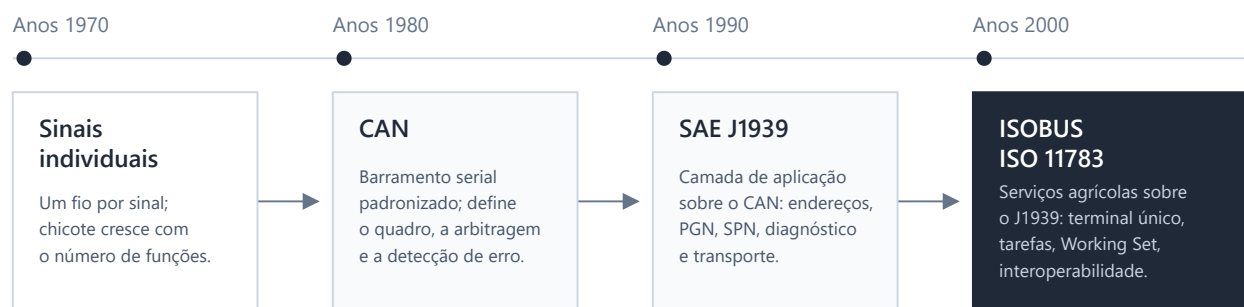
NA PRÁTICA

Oito ECUs que precisassem conversar todas entre si exigiriam 28 ligações ponto a ponto. Um barramento CAN compartilha o mesmo par de dados entre os nós; ainda são necessários alimentação, terra, terminações e derivações conforme o projeto.

De sinais individuais a uma rede

A passagem do fio dedicado para o barramento não é apenas uma economia de cobre: ela muda a natureza da comunicação. Em um fio dedicado, o significado do sinal está no *fio* — aquele condutor é a rotação do motor, e quem o lê sabe disso porque foi assim que o chicote foi projetado. Em um barramento compartilhado, o significado precisa viajar junto com o dado, dentro da mensagem. Surge então a necessidade de identificar mensagens, de priorizá-las e de definir regras para o caso de dois participantes falarem ao mesmo tempo.

Essa mudança se desdobrou em quatro fases, que este livro percorre na mesma ordem em que aconteceram historicamente (Figura 1.2):



Cada etapa preserva a anterior; o ISOBUS continua sendo, no fio, um barramento CAN de 250 kbit/s.

Figura 1.2 — Evolução da comunicação embarcada em quatro fases. O que muda a cada etapa não é o meio físico, mas a quantidade de significado padronizado que trafega sobre ele.

1. **Fase inicial (anos 1970)** — comunicação ponto a ponto com fios dedicados para cada sinal.
2. **Introdução do CAN (anos 1980)** — protocolo de barramento para controle distribuído.
3. **Padronização específica por setor (anos 1990)** — desenvolvimento do J1939 para veículos comerciais.
4. **Especialização agrícola (anos 2000)** — desenvolvimento do ISOBUS para máquinas agrícolas.

O que é uma rede embarcada

CONCEITO

Rede embarcada é o conjunto formado pelo meio físico compartilhado, pelos nós conectados a ele e pelas regras que governam o acesso ao meio, o formato das mensagens e a interpretação dos dados. O meio físico e as regras de acesso pertencem ao protocolo de baixo nível (CAN); o significado dos dados pertence às camadas de aplicação construídas sobre ele (J1939, ISOBUS).

As normas dão nomes próprios a essa rede: a SAE J1939 se apresenta como rede de controle e comunicação serial para veículos pesados (*Heavy Duty Vehicle Network*), e a série ISO 11783, como rede serial para comunicação de dados e controle em tratores e máquinas agrícolas e florestais. Este livro chama de **rede embarcada** a rede que acompanha a

máquina: a do trator e a que se forma quando o implemento é acoplado.

Três características distinguem uma rede embarcada de uma rede de computadores comum e explicam a maior parte das decisões de projeto que serão estudadas adiante:

- **Tempo real e determinismo.** Uma mensagem de frenagem não pode esperar indefinidamente porque outro nó está transmitindo estatísticas operacionais. O acesso ao meio precisa ser resolvido por *prioridade*, de forma previsível.
- **Ambiente eletricamente hostil.** Ignição, motores elétricos, alternadores e cabos longos produzem ruído. A transmissão precisa ser robusta por construção — daí o par trançado e a sinalização diferencial do Capítulo 3.
- **Custo por nó.** Cada ECU adicional multiplica o custo do controlador e do transceptor. O protocolo precisa ser simples o bastante para caber em silício barato e em microcontroladores modestos.

Uma quarta característica aparece especificamente no mundo agrícola e será central na Parte III: a **topologia dinâmica**. A rede interna do trator tende a conservar seus módulos; no barramento ISOBUS, um implemento diferente pode ser acoplado a cada operação. Os dispositivos precisam reconhecer suas funções e negociar endereços para trabalhar juntos.

A pilha: CAN, J1939 e ISOBUS

Os três protocolos deste livro não competem entre si: eles se empilham. O CAN fornece o mecanismo de transporte — como os bits viajam e como o meio é disputado. O J1939 define *como usar* esse transporte em máquinas e veículos pesados, como o trator, padronizando endereços, grupos de parâmetros e diagnóstico. O ISOBUS estende o J1939 com os serviços que uma máquina agrícola exige (Figura 1.3).

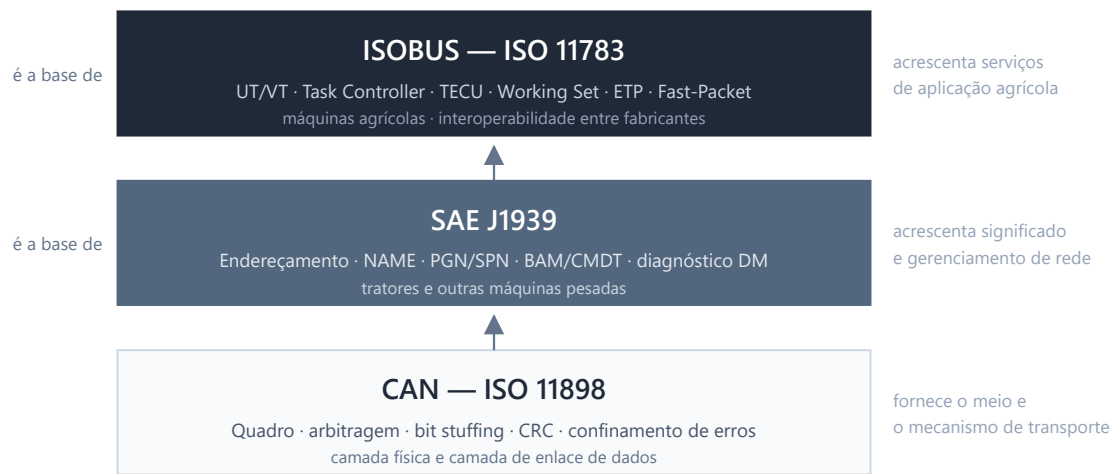


Figura 1.3 — Relação entre os três protocolos. Cada camada superior é uma convenção de uso da camada inferior — nenhuma delas substitui a anterior.

A Tabela 1.1 compara os três protocolos quanto à norma, à base, ao foco, à taxa de bits e aos setores de aplicação.

Tabela 1.1 — Relação entre os protocolos estudados neste livro.

Protocolo	Padrão	Baseado em	Foco principal	Taxa de bits	Setores de aplicação
CAN	ISO 11898	—	Comunicação básica entre ECUs	Definida por segmento	Trator e implemento
J1939	SAE J1939	CAN	Funções da máquina motriz	250 kbit/s nos exemplos	Tratores e outras máquinas pesadas
ISOBUS	ISO 11783	J1939	Máquinas agrícolas	250 kbit/s (padrão)	Tratores e implementos agrícolas

- **CAN** — protocolo básico de baixo nível, que define o meio físico e a comunicação fundamental.
- **J1939** — constrói sobre o CAN, acrescentando endereçamento e formatos padronizados usados em funções de máquinas pesadas.

- **ISOBUS / ISO 11783** — estende o J1939 para aplicações agrícolas específicas, acrescentando funcionalidades como o Terminal Virtual e o Controlador de Tarefas.

Onde cada protocolo aparece na máquina agrícola

O trator pode ter uma ou mais redes CAN internas, com mensagens J1939 ou convenções do fabricante. O barramento ISO 11783 liga funções do trator e dos implementos. Uma ECU do implemento também pode usar outra rede interna para sensores e atuadores; essa rede não se torna ISOBUS apenas por usar CAN (Tabela 1.2).

Tabela 1.2 — Aplicação dos protocolos no conjunto trator-implemento.

Segmento	Protocolo possível	Exemplos de funções
Rede interna do trator	CAN e, conforme o projeto, J1939	Motor, transmissão, hidráulica e diagnóstico
Rede trator-implemento	ISOBUS / ISO 11783	TECU, terminal, controlador de tarefas e ECUs de implementos
Rede interna do implemento	CAN ou outro meio definido pelo fabricante	Sensores, válvulas, seções e atuadores locais

ATENÇÃO

É comum ouvir que “o ISOBUS é um protocolo diferente do CAN”. Do ponto de vista do fio e do osciloscópio, **não é**: um barramento ISOBUS é um barramento CAN de alta velocidade a 250 kbit/s, com quadros CAN de identificador estendido. Tudo o que o ISOBUS acrescenta é convenção sobre o conteúdo e sobre o comportamento dos nós. Essa observação é prática: qualquer ferramenta que capture CAN captura ISOBUS — o que muda é a capacidade de *interpretar* o que foi capturado.

O que este livro vai construir

A ordem dos capítulos reproduz a ordem em que os conceitos se sustentam. Nos próximos cinco capítulos, o CAN será desmontado peça por peça: o meio físico e os níveis elétricos, a arbitragem, a estrutura do quadro e o tratamento de erros. Só então o identificador estendido de 29 bits ganhará significado J1939, e só depois disso os serviços ISOBUS farão sentido — porque todos eles são, no fim, mensagens J1939 trafegando em quadros CAN.

Exercícios

EXERCÍCIO 1.1

Um subsistema de uma colheitadeira tem 8 ECUs que precisam trocar informações entre si. Quantas ligações dedicadas seriam necessárias em uma arquitetura ponto a ponto totalmente interligada? E quantos condutores de dados são necessários com um barramento CAN?

EXERCÍCIO 1.2

Explique, em duas frases, a diferença fundamental entre o significado de um sinal em um fio dedicado e o significado de um dado em um barramento compartilhado.

EXERCÍCIO 1.3

Associe cada necessidade à camada que a resolve — CAN, J1939 ou ISOBUS:

- Garantir que a mensagem mais urgente vença a disputa pelo barramento.
- Definir que o valor de rotação do motor está nos bytes 4 e 5 de um grupo de parâmetros específico.
- Fazer com que um pulverizador de um fabricante apareça no terminal instalado na cabine de um trator de outro fabricante.
- Detectar que um quadro chegou corrompido e provocar sua retransmissão.
- Atribuir automaticamente um endereço único a cada ECU na inicialização.

EXERCÍCIO 1.4

Um técnico afirma: “Preciso de um analisador ISOBUS; meu analisador CAN não serve para essa máquina”. A afirmação está tecnicamente correta? Justifique, distinguindo captura de interpretação.

RESUMO DO CAPÍTULO

- A expansão da eletrônica embarcada multiplicou o número de ECUs; a fiação ponto a ponto cresce com o quadrado do número de participantes e se tornou inviável.
- O barramento serial substitui o significado “no fio” pelo significado “na mensagem”, o que exige identificação, priorização e regras de acesso ao meio.
- Uma rede embarcada é definida por três exigências: tempo real com prioridade, robustez elétrica e baixo custo por nó. No mundo agrícola, soma-se a topologia dinâmica.
- CAN, J1939 e ISOBUS formam uma pilha: o CAN transporta, o J1939 dá significado padronizado e gerencia a rede, e o ISOBUS acrescenta os serviços agrícolas.
- Um barramento ISOBUS é eletricamente um barramento CAN a 250 kbit/s com identificadores de 29 bits.

08

O identificador J1939 de 29 bits

OBJETIVO DO CAPÍTULO

Dominar a decomposição do identificador estendido em campos J1939 e ser capaz de extrair, de qualquer linha de captura, a prioridade, o PGN, o endereço de origem e, quando existir, o endereço de destino — com papel e caneta, sem ferramenta.

CONCEITOS QUE SERÃO APRENDIDOS

- Os seis campos do identificador: Priority, EDP, DP, PF, PS e SA
- Formatos PDU1 e PDU2 e o significado do limiar PF = 240
- Cálculo do PGN a partir do identificador
- Reconstrução do identificador a partir do PGN, da prioridade e dos endereços
- A prioridade: quem escolhe, quais são os padrões e como lê-la
- A cadeia de interpretação: CAN ID → campos → PGN → SPNs → valores físicos

Os seis campos

O J1939 organiza os 29 bits do identificador CAN estendido em seis campos de tamanho fixo. A leitura é feita do bit mais significativo (o primeiro transmitido, e portanto o primeiro a participar da arbitragem) para o menos significativo (Figura 8.1).

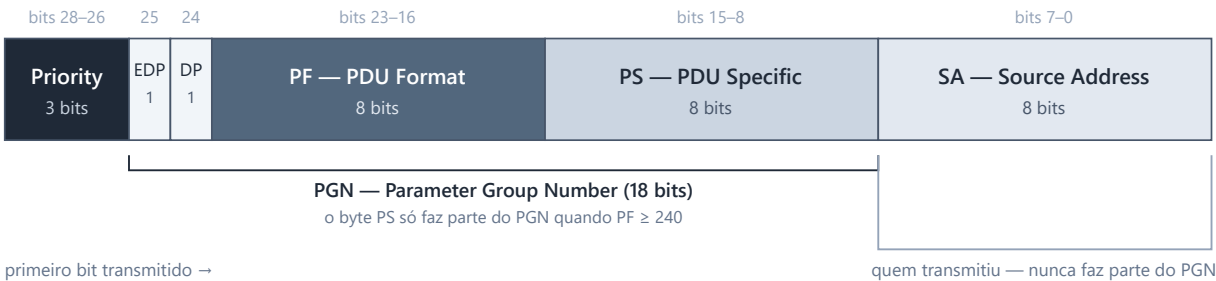


Figura 8.1 — Decomposição do identificador J1939 de 29 bits. As larguras são proporcionais ao número de bits de cada campo.

A Tabela 8.1 descreve a função de cada campo.

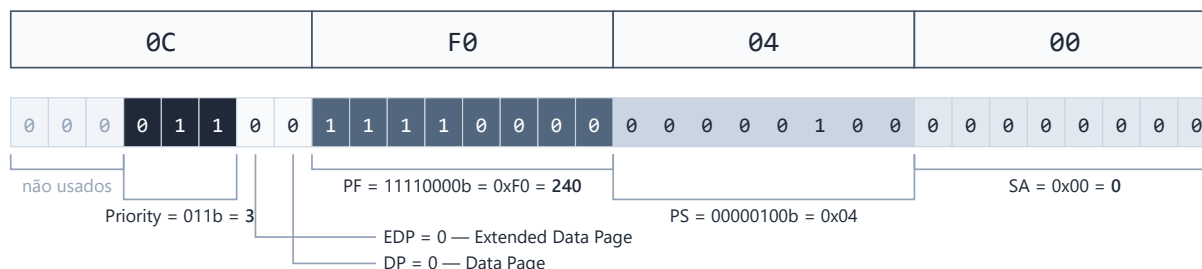
Tabela 8.1 — Campos do identificador J1939.

Campo	Bits	Função
Priority	3	Prioridade da mensagem, de 0 (maior) a 7 (menor) . Por estar no início do identificador, é o campo que domina a arbitragem. Não altera o significado da mensagem.
EDP	1	<i>Extended Data Page</i> . Junto com o DP, seleciona a página de definições à qual o grupo de parâmetros pertence.
DP	1	<i>Data Page</i> . Página de dados 0 ou 1; amplia o espaço de PGNs disponíveis.
PF	8	<i>PDU Format</i> . Determina o tipo de mensagem e, sobretudo, define se a mensagem carrega um destinatário no PS (PF < 240) ou uma extensão de grupo (PF ≥ 240).
PS	8	<i>PDU Specific</i> . Tem dois significados possíveis: Destination Address quando PF < 240, ou Group Extension quando PF ≥ 240.

Campo	Bits	Função
SA	8	Source Address. Endereço da ECU que transmitiu a mensagem. É a única identificação de origem existente no quadro.

Do hexadecimal aos campos

Uma ferramenta de análise exibe o identificador em hexadecimal, normalmente como quatro bytes. A Figura 8.2 mostra, para o identificador `0x0CF00400`, como esses quatro bytes se relacionam com os campos do J1939.



O primeiro byte hexadecimal concentra três campos:

byte `0x0C` = $0000\ 1100b \rightarrow \text{Priority} = (0x0C \gg 2) \& 0x07 = 3$

EDP = $(0x0C \gg 1) \& 0x01 = 0$

DP = $0x0C \& 0x01 = 0$

Os três bits mais significativos do valor de 32 bits não existem no quadro: o identificador CAN estendido tem 29 bits.

Figura 8.2 — Do identificador hexadecimal aos campos J1939. O byte inicial `0x0C` carrega prioridade e páginas de dados; os três bytes seguintes são, respectivamente, PF, PS e SA.

COMO DECODIFICAR

Dado um identificador de 29 bits em hexadecimal, na ordem:

1. **Separe os quatro bytes.** `0x0CF00400` → `0C | F0 | 04 | 00`.
2. **Extraia a prioridade e as páginas do primeiro byte.** Priority = (byte $\gg 2$) & 7; EDP = (byte $\gg 1$) & 1; DP = byte & 1.
3. **Leia PF, PS e SA diretamente do segundo, terceiro e quarto bytes.**
4. **Compare PF com 240 (0xF0).** Se PF < 240, a mensagem é PDU1 e PS é o endereço de destino. Se PF ≥ 240 , a mensagem é PDU2 e PS é a extensão de grupo.
5. **Monte o PGN.** PDU1: PGN = $EDP \cdot 2^{17} + DP \cdot 2^{16} + PF \cdot 2^8 + PS$. PDU2: PGN = $EDP \cdot 2^{17} + DP \cdot 2^{16} + PF \cdot 2^8 + PS$.
6. **Consulte o PGN** em uma base de parâmetros para descobrir quais SPNs ele transporta.

Para ISOBUS na ABNT NBR ISO 11783-3:2021, 5.2.3 e 5.2.4, a transmissão usa **EDP = 0**; DP seleciona a página 0 ou 1. A fórmula acima permite extrair os campos, mas um identificador com EDP = 1 não deve ser automaticamente interpretado como um PGN ISOBUS definido por essa edição.

A prioridade: quem escolhe e quanto custa errar

Os três bits de prioridade são o campo mais mal compreendido do identificador. Duas afirmações resolvem a maior parte da confusão. A primeira: **menor número significa maior prioridade** — 0 é a mais alta, 7 a mais baixa — porque a arbitragem é vencida pelo identificador numericamente menor. A segunda: **a prioridade não faz parte do PGN**. Ela é escolhida pelo transmissor, e o mesmo grupo de parâmetros pode trafegar com prioridades diferentes sem deixar de ser o mesmo grupo de parâmetros.

Isso não significa que a escolha seja livre na prática. A ABNT NBR ISO 11783-3:2021, 5.2.2 (adoção da ISO de 2018), estabelece dois valores de referência — **3** para mensagens orientadas a controle e **6** para as demais, informativas, proprietárias, de requisição e de confirmação (ACK) — e atribui uma **prioridade recomendada a cada PGN** quando ele

entra na camada de aplicação (Tabela 8.2). Essa edição recomenda que o campo permaneça reprogramável, para permitir ajuste fino da rede pelo fabricante.

ATENÇÃO — EDIÇÃO DE 2026

A ISO 11783-3:2026, 5.2.2.1 a 5.2.2.3, distingue **prioridade atribuída** e **prioridade-padrão**. Uma mensagem com prioridade atribuída deve usar esse valor; a recomendação de reprogramação da edição anterior não permite alterá-lo livremente. Mensagens de TP e ETP usam sua própria prioridade atribuída, e não a do grupo transportado. O receptor deve ignorar o campo de prioridade ao interpretar a mensagem. A categoria de prioridade-padrão continua prevista.

Os valores da tabela seguinte pertencem às edições usadas nos exemplos.

Tabela 8.2 — Prioridade-padrão dos grupos de parâmetros usados neste livro.

Prioridade	Grupo de parâmetros	Uso
3	61444 · EEC1	Dados do motor
3	65096 · WBSD	Velocidade e distância calculadas da roda
3 a 5	51968 · Process Data	TC ↔ implemento; varia com o comando
5	58880 e 59136 · VT ↔ ECU	Comandos e respostas do terminal
6	60928 · Address Claimed	Gerenciamento de rede
6	59904 · Request	Requisição de um grupo de parâmetros
6	65226 · DM1	Códigos de falha ativos
6	65039 · Comando de Linguagem	Idioma e unidades
7	60416 e 60160 · TP.CM e TP.DT	Protocolo de transporte
7	51200 e 50944 · ETP.CM e ETP.DT	Transporte estendido
7	65037 · Working Set Master	Composição do conjunto de trabalho

NA PRÁTICA

A tabela explica um efeito que se observa em qualquer captura congestionada: o protocolo de transporte trafega em **prioridade 7**, a mais baixa. Isso é deliberado — uma transferência de *Object Pool* com milhares de quadros não pode atrasar um comando de controle. A consequência é que o carregamento do *pool* demora mais em uma rede ocupada, e não é defeito.

No sentido inverso, encontrar uma mensagem informativa transmitida em prioridade 0 ou 1 é motivo para investigar: alguém elevou a prioridade para “resolver” uma latência, e essa mensagem agora atropela o controle. Ao comparar um quadro capturado com a tabela acima, a diferença é informação: ou o fabricante ajustou a rede conscientemente, ou alguém copiou um identificador sem entender o campo.

PDU1 e PDU2: o limiar dos 240

A distinção entre os dois formatos de PDU (*Protocol Data Unit*) é o ponto em que mais se erra ao decodificar identificadores manualmente. A regra é simples e depende de um único campo:

PDU1 — PF < 240 (0X00 A 0XEF)

- Mensagem **endereçada** a uma ECU específica.
- PS = **Destination Address (DA)**.
- O byte PS **não faz parte do PGN** — ele é zerado no cálculo.
- DA = 255 (0xFF) significa destino global: a mensagem é endereçada “a todos”.
- Exemplos: *Request*, *Address Claim*, protocolos de transporte, serviços do Virtual Terminal.

Cuidado com a simplificação “PDU1 é endereçado, PDU2 é broadcast”. O PDU1 **pode** ser broadcast: basta que o DA seja 255. O *Address Claim* é obrigatoriamente assim, e um *Request* ao endereço global pergunta a toda a rede de uma vez. O que distingue os formatos não é o alcance, é se o campo PS carrega um destinatário ou uma extensão de grupo.

PDU2 — PF ≥ 240 (0XF0 A 0XFF)

- **Sempre** broadcast: o formato não tem onde colocar um destinatário.
- PS = **Group Extension (GE)**.
- O byte PS **faz parte do PGN**, ampliando o espaço de grupos disponíveis.
- Não existe endereço de destino no identificador.
- Exemplos: dados periódicos de motor, velocidade, temperatura, estado de sistemas.

A Figura 8.3 resume a decisão entre os dois formatos.

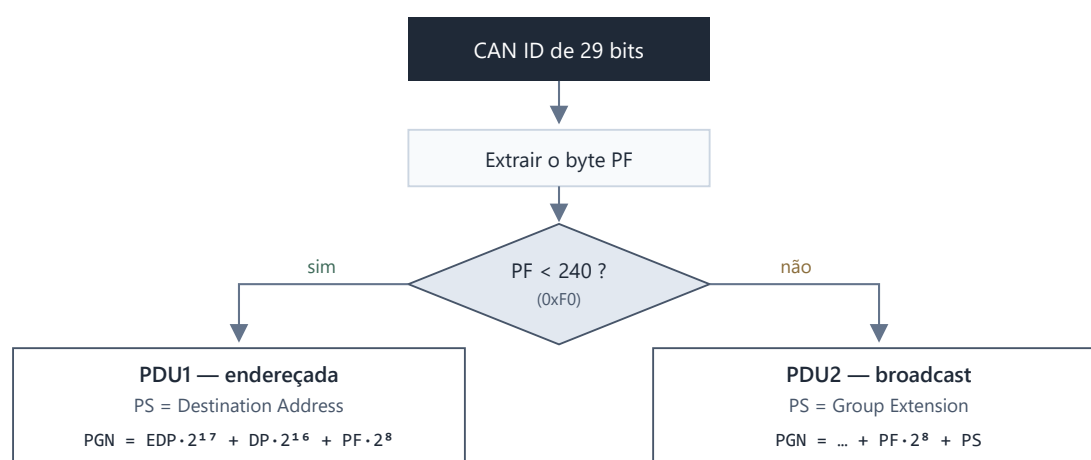


Figura 8.3 — Fluxo de decisão entre PDU1 e PDU2. Toda decodificação manual de identificador J1939 passa por essa bifurcação.

ATENÇÃO

O erro mais comum é **incluir o byte PS no PGN de uma mensagem PDU1**. Se isso for feito, cada destinatário diferente pareceria um PGN diferente — e um *Request* ao destino global pareceria outro grupo de parâmetros quando dirigido a uma ECU específica. Nos dois casos, seu PGN é **0x00EA00**: só o destino muda. O *Address Claimed*, por sua vez, sempre usa destino global.

Decodificação completa, passo a passo

FRAME EXAMPLE — PDU2 (BROADCAST)

TIMESTAMP CAN ID DLC DATA
12.440118 0x0CF00400 8 FF 7D AF E0 2E 00 FF FF

Passo	Resultado	Comentário
Separar os bytes	0C F0 04 00	Quatro bytes do identificador estendido
Prioridade	3	(0x0C >> 2) & 7 = 3 — a prioridade-padrão das mensagens de controle, como o EEC1
EDP / DP	0 / 0	Página de dados 0
PF	0xF0 = 240	240 ≥ 240 → formato PDU2
PS	0x04	Group Extension: faz parte do PGN
SA	0x00 = 0	Endereço 0 = ECU do motor
PGN	0x00F004 = 61444	$0 \cdot 2^{17} + 0 \cdot 2^{16} + 240 \cdot 256 + 4 = 61444$ — <i>Electronic Engine Controller 1</i>

Leitura em linguagem natural: “a ECU de endereço 0 (motor) transmitiu, em broadcast e com prioridade 3, o grupo de parâmetros 61444, que contém os dados básicos de operação do motor”. Os oito bytes serão interpretados no Capítulo 9.

FRAME EXAMPLE — PDU1 (ENDEREÇADA AO DESTINO GLOBAL)

CAN ID DLC DATA
0x18EEFF00 8 31 D4 60 2A 00 00 00 20

- Bytes do identificador: 18 | EE | FF | 00
- Prioridade = 6; EDP = 0; DP = 0
- PF = 0xEE = 238 → 238 < 240, portanto **PDU1**
- PS = 0xFF = 255 → **Destination Address global**
- SA = 0x00 = 0 → transmitida pela ECU do motor
- PGN = $0 \cdot 2^{17} + 0 \cdot 2^{16} + 238 \cdot 256 = 60928$ (0x00EE00) — *Address Claimed*

Trata-se de uma reivindicação de endereço: a ECU anuncia a toda a rede que passa a usar o endereço 0. Os oito bytes de dados são o NAME de 64 bits, estudado no Capítulo 10.

FRAME EXAMPLE — PDU1 (ENDEREÇADA A UMA ECU ESPECÍFICA)

CAN ID DLC DATA
0x18EA0026 3 04 F0 00

- Bytes do identificador: 18 | EA | 00 | 26
- PF = 0xEA = 234 < 240 → **PDU1**
- PS = 0x00 → destino: ECU de endereço 0
- SA = 0x26 = 38 → origem: Virtual Terminal
- PGN = $234 \cdot 256 = 59904$ (0x00EA00) — *Request*
- Dados: o PGN solicitado, três bytes do menos para o mais significativo — 04 F0 00 = 0x00F004 = 61444

Leitura: “o Virtual Terminal (endereço 38) pediu à ECU do motor (endereço 0) que transmita o grupo de parâmetros 61444”. Note a ordem invertida dos bytes do PGN solicitado: esse detalhe aparece em quase todas as mensagens J1939 e é tratado no próximo capítulo.

O caminho inverso: montar o identificador

Implementar uma ECU exige o caminho contrário — a partir do PGN, da prioridade e dos endereços, construir o identificador de 29 bits.

EXEMPLO

Transmitir o PGN 65265 (0x00FEF1, *Cruise Control/Vehicle Speed*) com prioridade 6, a partir da ECU de endereço 0.

1. PGN 65265 = 0x00FEF1 → EDP = 0, DP = 0, PF = 0xFE = 254, PS = 0xF1.
2. PF = 254 ≥ 240 → PDU2: o PS já está contido no PGN e não é substituído por endereço de destino.
3. Primeiro byte = (6 << 2) | (0 << 1) | 0 = 0x18.
4. Identificador = 0x18 FE F1 00 = 0x18FEF100.

Para um PGN PDU1 — por exemplo, o *Request* 59904 dirigido à ECU 3 a partir da ECU 38 — o byte PS é substituído pelo endereço de destino: 0x18 EA 03 26 = 0x18EA0326.

A cadeia completa de interpretação

Vale fixar o percurso que será repetido em todas as análises do restante do livro. Ele começa em uma linha de captura e termina em uma grandeza física com unidade (Figura 8.4).



Os dois primeiros passos dependem apenas de aritmética; os dois últimos exigem uma base de dados de parâmetros.

Figura 8.4 — A cadeia de interpretação de uma mensagem J1939, do identificador capturado ao valor de engenharia.

PGNs frequentes

A Tabela 8.3 reúne os grupos de parâmetros encontrados com frequência em redes J1939.

Tabela 8.3 — Grupos de parâmetros encontrados com frequência em redes J1939.

PGN (hex)	Decimal	Nome	Descrição
0xF004	61444	Electronic Engine Controller 1	Dados básicos do motor (rotação, torque)
0xF003	61443	Electronic Engine Controller 2	Posição do pedal e carga do motor
0xF002	61442	Electronic Transmission Controller 1	Engate do trem de força, conversor de torque, troca em andamento e rotações dos eixos
0xF005	61445	Electronic Transmission Controller 2	Faixa selecionada e faixa atual da transmissão
0xF001	61441	Electronic Brake Controller 1	Informações do sistema de freios
0xFE0F	65263	Engine Fluid Level/Pressure 1	Níveis e pressões de fluidos do motor
0xFE01	65265	Cruise Control / Vehicle Speed	Velocidade e controle de cruzeiro
0xFE6C	65132	Tachograph	Informações do tacógrafo
0xEE00	60928	Address Claimed	Reivindicação de endereço (PDU1)
0xEA00	59904	Request	Requisição de um PGN (PDU1)
0xE800	59392	Acknowledgment	Resposta positiva ou negativa a uma requisição

NOTA TÉCNICA

Os PGNs 61443 (*Electronic Engine Controller 2*), 61442 (*Electronic Transmission Controller 1*) e 61445 (*Electronic Transmission Controller 2*) identificam grupos distintos. A semelhança entre nomes e siglas exige atenção na leitura de capturas e na configuração de bancos de dados. Consulte o registro aplicável do *J1939 Digital Annex* para confirmar nomes e parâmetros antes de implementar.

Exercícios

EXERCÍCIO 8.1

Dado `CAN ID = 0x0CF00400`, determine: prioridade, PF, PS, SA, tipo de PDU e PGN.

EXERCÍCIO 8.2

Decodifique o identificador `0x0CF00203`. Indique prioridade, tipo de PDU, PGN e endereço de origem.

EXERCÍCIO 8.3

Uma mensagem tem `CAN ID = 0x1CECFF26`. Identifique o tipo de PDU, o destino, o PGN e diga que tipo de serviço ele representa (consulte a tabela de PGNs deste capítulo e, se necessário, o Capítulo 11).

EXERCÍCIO 8.4

Monte o identificador de 29 bits para transmitir o PGN 65262 (`0x00FEEE`) com prioridade 6, a partir da ECU de endereço 3.

EXERCÍCIO 8.5

Por que duas mensagens *Request* enviadas pela mesma ECU, uma para o destino global e outra para uma ECU específica, têm o mesmo PGN, mas identificadores diferentes? Por que isso não se aplica ao *Address Claimed*?

RESUMO DO CAPÍTULO

- Os 29 bits se dividem em Priority (3), EDP (1), DP (1), PF (8), PS (8) e SA (8).
- O primeiro byte hexadecimal concentra prioridade, EDP e DP; $\text{Priority} = (\text{byte} \gg 2) \& 7$.
- $\text{PF} < 240$ indica PDU1 (endereçada, PS = destino, PS fora do PGN); $\text{PF} \geq 240$ indica PDU2 (broadcast, PS = extensão de grupo, PS dentro do PGN).
- $\text{PGN} = \text{EDP} \cdot 2^{17} + \text{DP} \cdot 2^{16} + \text{PF} \cdot 2^8 [+ \text{PS, apenas em PDU2}]$.
- O SA nunca faz parte do PGN: ele diz quem transmitiu, não o que foi transmitido.
- A cadeia de análise é sempre a mesma: CAN ID → campos → PGN → SPNs → valores físicos.