

A Hands-On Approach to Automotive and Industrial Networks



#### Copyright © [2025] by Santiago Corrales V.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopying, recording, or otherwise—without prior written permission from the author, except for brief quotations in a review or academic citation.

This book is intended for educational and informational purposes only. While every effort has been made to ensure accuracy, the author and publisher assume no responsibility for errors or omissions. The information provided is subject to change as technology evolves. The author and publisher disclaim any liability for the misuse of the information contained in this book.

For permissions, inquiries, or licensing, please contact:

INPRONIC USA LLC

sales@inpronicusa.com

www.inpronicusa.com

ISBN 979-8-89901-487-1

## DEDICATION

To Claudía Vallejo S.,

Your unwavering support, patience, and encouragement have been the foundation of this journey. Through every challenge and late-night endeavor, your belief in me has been a source of strength and inspiration.

This book is dedicated to you—not only for your love and understanding but for the countless ways you have stood beside me, making every achievement more meaningful.

With deepest gratitude love and admiration,

SANTY

### TABLE OF CONTENTS

Chapter 1: Introduction to CAN-BUS1	1
1. Understanding the CAN-BUS Network1	1
1.1 The Origin and Evolution of CAN1	2
1.2 How CAN-BUS Works1	3
1.3 Physical Characteristics of the CAN-BUS	4
1.4 Key Features of CAN-BUS	6
1.5 Applications of CAN-BUS	7
1.6 Advantages of CAN-BUS	21
1.7 Using the CAN-BUS Trainer to Learn	21
1.8 Future of CAN-BUS2	22
1.9 Summary	23
Chapter 2: Exploring the CAN-BUS Trainer2	5
2.1 Overview of the CAN-BUS Trainer2	5
2.2 Components of the CAN-BUS Trainer2	6
2.3 CAN Network Configuration Using the Trainer	1
2.4 Resistor Configuration for CAN Termination3	2
2.5 Software: CAN-Bus Multiplex Trainer V3.0	34

2.6 Summary	35
Chapter 3: CAN-BUS Multiplex Trainer Software	37
3.1 CAN-Bus Multiplex Trainer Board	38
3.2 SAE Settings J1939	39
3.3 Serial Features	40
3.4 CAN-Bus Features	41
3.5 Transmit Data	43
3.6 Transmit Tab	45
3.7 Interval Tab	46
3.8 Trainer Tab	46
3.9 Filtering and Receive Tabs	47
3.10 Sniffer Tab	49
Chapter 4: OBD-II Protocol	51
4.1 What is OBD-II?	52
4.2 OBD-II System Components	53
4.3 How OBD-II Works	54
4.4 OBD-II Communication Protocols	55
4.5 Diagnostic Trouble Codes (DTCs) and Freeze Data	Frame 56

4.6 OBD-II Modes of Operation
4.7 Practical Applications of OBD-II
4.8 Summary
Examples Using OBD-II
Diagnostic Fuel Level (PID 0x2F)59
Retrieving RPM Data via OBD-II Over CAN Bus67
Retrieving the Vehicle Identification Number (VIN)73
Diagnostic under OBD-II77
Using the SNIFFER tool under OBD-II83
Hacking my CAR using a Trainer and OBD-II CAN DATA91
Chapter 5: J1939 Protocol97
5.1 Introduction to J1939
5.2 J1939 Network Architecture
5.3 J1939 Message Structure
5.4 Addressing in J1939101
5.5 J1939 Communication Process106
5.6 J1939 Parameter Group Numbers (PGNs)106
5.7 J1939 Transport Protocol for Large Messages107

5.8 J1939 in Real-World Applications107
5.9 Advantages of J1939108
5.10 Step-by-Step Guide to Calculating the J1939 PGN ID
5.11 Summary115
Practical Applications Using J1939
Setting ECU J1939 Name in the Trainer116
Using the Sniffer Tool to Capture Turn Signal Data in a J1939 Protocol
Hacking My Truck Using a Trainer and J1939 CAN Data
Simulating RPM in J1939 Protocol132
Chapter 6: CAN-OPEN Protocol139
6.1 Introduction to CAN-open140
6.2 CAN-open Network Architecture141
6.3 CAN-open Communication Mechanisms143
6.4 CAN-open Object Dictionary (OD)145
6.5 CAN-open Device Profiles146
6.6 Advantages of CAN-open146
6.7 Summary

#### Practical Applications Using CAN-open

Checking the CAN-Open Protocol in a Product	148
Chapter 7: NMEA 2000 Protocol	155
7.1 Introduction to NMEA 2000	155
7.2 NMEA 2000 Network Architecture	156
7.3 NMEA 2000 Communication and Data Structure	158
7.4 NMEA 2000 Addressing and Device Management	159
7.5 NMEA 2000 Data Transmission Methods	160
7.6 Advantages of NMEA 2000	161
7.7 Real-World Applications of NMEA 2000	162
7.8 Summary	163

#### Practical Applications Using NMEA 2000

Using	the	CAN-BUS	Multiplex	Trainer	with	NMEA
2000	•••••	••••••••••••	••••••••••			164

### Preface

The modern era of vehicle communication, industrial automation, and marine navigation has been shaped by one fundamental advancement: the Controller Area Network (CAN-BUS) protocol. Originally developed to streamline automotive electronics, CAN-BUS has evolved into a versatile and widely adopted standard across multiple industries, ensuring robust, efficient, and scalable data exchange between electronic control units (ECUs).

This book provides a comprehensive exploration of CAN-BUS and its primary protocols—OBD-II, J1939, CAN-Open, and NMEA 2000—offering both theoretical knowledge and practical application. Designed for engineers, technicians, developers, and industry professionals, this text serves as a definitive guide to understanding and implementing CAN-BUS communication systems.

What distinguishes this book is its applied approach. Beyond conceptual understanding, it introduces hands-on simulations using the CAN-BUS Multiplex Trainer, a powerful tool for monitoring, analyzing, and simulating CAN network behavior. Each chapter delves into: Fundamental principles of CAN-BUS architecture, arbitration, and error handling.

Protocol-specific communication methods, including message structures, PGNs, PIDs, and addressing schemes.

Practical simulations and analysis using the CAN-BUS Trainer to replicate real-world communication scenarios.

Industry case studies, illustrating the application of CAN-BUS in vehicle diagnostics, industrial control systems, and marine electronics.

By the conclusion of this book, readers will not only possess an in-depth understanding of CAN-BUS but will also be equipped with the practical skills necessary to develop, troubleshoot, and optimize CAN communication networks.

As industries continue to embrace automation, data integration, and intelligent diagnostics, the relevance of CAN-BUS will only grow. Whether for professional development, research, or system implementation, this book provides the essential foundation to master CAN-BUS technology and its applications.

Santíago Corrales V.

#### C.E.O. INPRONIC USA

### CHAPTER 1

### **Introduction to CAN-BUS**

#### 1. Understanding the CAN-BUS Network

The Controller Area Network (CAN) is an advanced and highly efficient communication protocol that has revolutionized how electronic devices communicate and interact within complex systems. Originally developed by Bosch in the 1980s, the CAN protocol was designed to simplify the growing complexity of automotive wiring by reducing the number of wires required for communication between various electronic control units (ECUs). It provides a reliable, robust, and efficient means of data exchange between multiple devices, known as nodes, within a network.

The CAN protocol uses a two-wire system, known as CAN-High (CAN-H) and CAN-Low (CAN-L), to facilitate differential communication, ensuring high noise immunity and data integrity. Over the decades, CAN technology has evolved beyond its automotive origins and has become a standard protocol in industrial automation, marine applications, medical

equipment, and aerospace systems. Its scalability, reliability, and real-time communication capabilities make CAN a fundamental technology in modern electronic and automation systems.



#### 1.1 The Origin and Evolution of CAN

Before the introduction of CAN, automotive systems relied on complex and bulky wiring harnesses to connect individual components. As vehicles became more sophisticated, with an increasing number of electronic control units (ECUs), the traditional point-to-point wiring approach became unsustainable. This led to the development of the CAN protocol, which significantly reduced wiring complexity while improving reliability and scalability.

CAN was initially designed to handle the demanding requirements of automotive applications, such as real-time data exchange and robust error handling. It has since evolved into a versatile communication standard used in protocols like J1939 (heavy-duty vehicles), CAN-open (industrial automation), and NMEA 2000 (marine electronics). These adaptations demonstrate the flexibility of CAN in addressing diverse industry needs.

#### **1.2 How CAN-BUS Works**

The CAN-BUS is a multi-master, message-oriented protocol. Unlike traditional networks, where a central controller manages communication, CAN allows any node to transmit data whenever the bus is free. This decentralized architecture ensures high levels of reliability and fault tolerance.

Each message transmitted on the CAN bus contains:

- *Identifier:* A unique value that determines the priority of the message.

- *Data Field:* The actual information being transmitted, which can be up to 8 bytes in standard CAN.

- Control and Error Fields: Additional bits used for arbitration, error checking, and acknowledgment.

Messages are broadcast to all nodes on the network, and each node decides whether to process the message based on its identifier. This approach simplifies communication and ensures real-time data exchange, even in complex systems.



#### **1.3 Physical Characteristics of the CAN-BUS**

The physical layer of the CAN-BUS typically consists of two twisted wires: CAN High and CAN Low. These wires form a differential pair, meaning the signal on one wire is the inverse of the other. This design provides several advantages:

- *Noise Immunity:* Differential signaling cancels out electromagnetic interference, ensuring reliable communication in electrically noisy environments.

- *Fault Tolerance:* The network can continue to operate even if one of the wires is damaged.

- *Scalability:* The twisted-pair configuration allows long cable runs and supports the addition of new nodes without significant reconfiguration.



# Hacking my CAR using a Trainer and OBD-II CAN Data



In this experiment, I'll demonstrate how to use the CAN-BUS Multiplex Trainer to send custom messages through the OBD-II port to control vehicle functions—specifically, locking the doors. By utilizing the Trainer Tab in the software and configuring one of the control buttons, I can transmit the exact CAN message that triggers the door lock. This process showcases how CAN data can be used to interact with and control vehicle systems, providing insights into how ECUs respond to specific commands.



I'll connect the trainer to the OBD-II port into my vehicle, I use this cable to connect the Trainer into a CANH, and CANL terminals.



#### **Step 2: Identify the Turn Signal Messages**

Activate the Turn Signals: Toggle the left and right turn indicators in the vehicle.



Look for Changing Data Rows: The Sniffer Tool will highlight the rows where data changes when the indicators are activated.

SNI	FER DAT	A															
	Receive	Data															0.055
	FILTER	BYTE:		1 🗆	2	3 🗌	4 🗆	5 🗌	6 🗌	7 🗆	8					PAUSE	CLOSE.
		D 0x 1950040	0.04	E Def	E DvE	E DvE	E DvF	E DvE	E DvE	E DvF	F I P	GN 6118	13			CLEAR RX	
												_	-				
ſ	#	PGN ID	B1	B2	B3	B4	85	B6	B7	88	PGN HEX	PGN DEC	DA	SA	Ρ		
- [	19	0x18FFB11E	0x04	0x03	0x00	0x30	0x0F	0xC3	0x10	0xF0	FFB1	65457	177	30	6		
- [	20	0x18FFD14C	0x52	0x41	0x44	0x49	0x4F	0x20	0x20	0x20	FFD1	65489	209	76	6		
-1	21	0x19FF094C	0x7D	0xFD	0xFF	0xFC	0xC1	0xFF	0x01	0xFF	1FF09	130825	9	76	6		
- [	22	0x18FEAE30	0x90	0x64	0x90	0x90	0xFF	0xFF	0xFC	0x64	FEAE	65198	174	48	6		
-1	23	0x18FFFD53	0x00	0xF8	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	FFFD	65533	253	83	6		
- [	24	0x18FECA53	0xF3	0xFF	FECA	65226	202	83	6								
- 1	25	0x18FEDF00	0x85	0xFF	0xFF	0xFF	0x7E	0xFF	0xFF	0xFF	FEDF	65247	223	0	6		
-1	26	0x18FEF34A	0xFB	0xE3	0x97	0x7C	0x54	0x98	0x50	0x4E	FEF3	65267	243	74	6		
-1	27	0x18FF5F00	0xFF	0x00	0xF0	0xD4	0xF0	0xFF	0xF8	0xFF	FF5F	65375	95	0	6		1000
-1	28	0x0CFF0419	0x03	0xFF	0xFF	0xCF	0xFF	0xFF	0xFF	0xFF	FF04	65284	4	25	3		horas a factor
- 1	29	0x0CFF0217	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	FF02	65282	2	23	3		
-1	30	0x18FF5119	0xFF	0x00	0x28	0xFF	0xF4	0xFA	0xFF	0xFF	FF51	65361	81	25	6		and the second second
-1	31	0x18FEE617	0xA0	0x32	0x10	0x03	0x34	0x28	0x7D	0x7D	FEE6	65254	230	23	6		
-1	32	0x18FFD517	0x04	0x30	0x06	0x24	0xF9	0xC7	0xFF	0xF0	FFD5	65493	213	23	6		
-1	33	0x18FF5019	0xF3	0xFC	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	FF50	65360	80	25	6		
-1	34	0x18FEF527	0x94	0xFF	0xFF	0x0A	0x24	0xFF	0xFF	0xFF	FEF5	65269	245	39	6		
-1	35	0x18FDE127	0x1F	0xFF	FDE1	64993	225	39	6								
-1	36	0x18FEC1EE	0x00	0x00	0x00	0xFE	0xFF	0xFF	0xFF	0xFF	FEC1	65217	193	238	6		a la participation de la comparticipation de
-1	37	0x18FF6A47	0xFC	0x3F	0x1F	0xFF	0xFF	0xFF	0xFF	0xFF	FF6A	65386	106	71	6		
-1	38	0x18FF7A00	0xF1	0x12	0x25	0x2A	0xE0	0x03	0xFF	0xFF	FF7A	65402	122	0	6		
-1	39	0x18FEFC47	0xFF	0x7E	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	FEFC	65276	252	71	6		
-1	40	0x18FFFAE6	0xFC	0xFF	0xFF	0xFF	0xFF	0x00	0xFF	0xFF	FFFA	65530	250	230	6		
-1	41	0x18FEEE00	0x48	0xFF	0x5C	0x26	0xFF	0xFF	0xFF	0xFF	FEEE	65262	238	0	6		
-1	42	0x18ECFF00	0x20	0x1C	0x00	0x04	0xFF	0xA9	0xFF	0x00	ECFF	60671	255	0	6		
-1	43	0x18FF6019	0x61	0x00	0x09	0x04	0xFF	0xFF	0xFF	0xFF	FF60	65376	96	25	6		
-1	44	0x18EBFF00	0x04	0x80	0x9D	0x14	0x54	0xA0	0x13	0x51	EBFF	60415	255	0	6		
-1	45	0x18FF6053	0x61	0x00	0x09	0x04	0xFF	0xFF	0xFF	0xFF	FF60	65376	96	83	6		
-1																	BUY HERE

In this case, when the turn signal lever is activated, row **19** will change dynamically, and it will be highlighted in **orange**. This visual indication helps identify the specific row associated with the turn signal function, allowing us to focus on the exact CAN message responsible for controlling the lights.

#### Analyze the Data Bytes:

Look for a specific byte change that corresponds to left or right signals,



In this case:

Right Turn Signal ON: Byte 1 changes to 0x14

Right Turn Signal OFF: Byte 1 changes to 0x04

# Hacking my TRUCK using a Trainer and J1939 CAN Data



The modern truck is more than just an engine and wheels it's a rolling network of electronic control units (ECUs) communicating over the J1939 CAN bus. By tapping into this data stream with a CAN-BUS Trainer, I can decode, manipulate, and even control key vehicle functions in real time. From monitoring sensor values to intercepting and modifying messages like turn signals or engine parameters, this journey into J1939 reveals the immense power of CAN data and the risks and

opportunities it presents for diagnostics, customization, and security.

Connect the upper board to the computer and navigate to the Training Tab within the software to begin configuring and sending CAN messages for simulation and analysis.



Select TX-SW4 to transmit the data, simulating the action of moving the lever to activate the **right turn signal**. This ensures the system recognizes the command as if the turn signal lever were physically engaged.

world scenario. By using the Trainer, you can refine your design, troubleshoot potential issues, and ensure optimal performance without delays, enabling a smooth transition from simulation to actual implementation.

#### INPRONIC USA



system is functioning correctly. This confirms that the data is being transmitted accurately, processed without errors, and properly reflected on the display, ensuring seamless integration and validation of the seat belt monitoring system.

