CAN COMMUNICATION NETWORKS

CONTENTS

AVI'S MISSION

The mission of Automotive Video Innovations (AVI) is to produce automotive repair training courses and information for the automotive aftermarket, providing current and future repair technicians with the highest level of technical training available. AVI's goal is for its course participants is to acquire the necessary skill set they seek and to apply those skills in a competent manner due to their training. AVI produces automotive technician training through instructor-led courses, technical training manuals, and other media, with a commitment to continually update this training to stay in step with ever-changing automotive technology.

COURSE OBJECTIVES

Upon successful completion of this course, you should be able to:

- ✓ Identify different network types
- ✓ Measure operating speed of the network
- ✓ Determine best network testing methods
- ✓ Decode CAN communications
- ✓ Interpret CAN lab scope waveforms
- ✓ Analyze transceiver operations

AVI ON DEMAND
ONLINE TRAINING FOR THE AUTOMOTIVE PROFESSIONAL

INTRODUCTION

In the modern automotive landscape, electronic systems play a critical role in enhancing vehicle performance, safety, and efficiency. At the core of this technological integration is the Controller Area Network (CAN) protocol—a high-speed communication system that has fundamentally changed how electronic control units (ECUs) within vehicles interact. Developed by Bosch in the 1980s, CAN has become the standard for in-vehicle networking, enabling efficient and reliable data exchange among various electronic components.

As vehicles have evolved to include advanced driver-assistance systems (ADAS), infotainment, and electronic control of critical functions like braking and steering, the need for a robust and scalable communication network has grown. CAN addresses these needs by providing a decentralized, fault-tolerant network capable of maintaining seamless operation even under the challenging conditions typical of automotive environments.

CAN was initially introduced to simplify the increasingly complex and cumbersome point-to-point wiring systems in vehicles. By enabling multiple ECUs to communicate over a single bus, CAN reduced the complexity and weight of wiring harnesses, resulting in cost savings and enhanced reliability. Today, CAN networks are not just replacements for traditional wiring but are foundational to the advanced functionalities in modern vehicles. Real-time communication made possible by CAN is essential for systems like anti-lock braking, electronic stability control, and engine management, where timely data exchange is critical for safety and performance.

The introduction of CAN FD (Flexible Data-rate) has further enhanced CAN’s capabilities, allowing for higher data rates and larger payloads. This evolution is vital as vehicles increasingly rely on more data-driven features, particularly in the context of automation and connectivity.

As the automotive industry advances towards electrification, autonomy, and greater connectivity, CAN networks will continue to evolve. Innovations like CAN XL, which offers even higher data rates and supports more complex network architectures, will meet the growing demands of next-generation vehicles. CAN’s adaptability ensures that it will remain a cornerstone of automotive communication, supporting the industry's ongoing transformation.

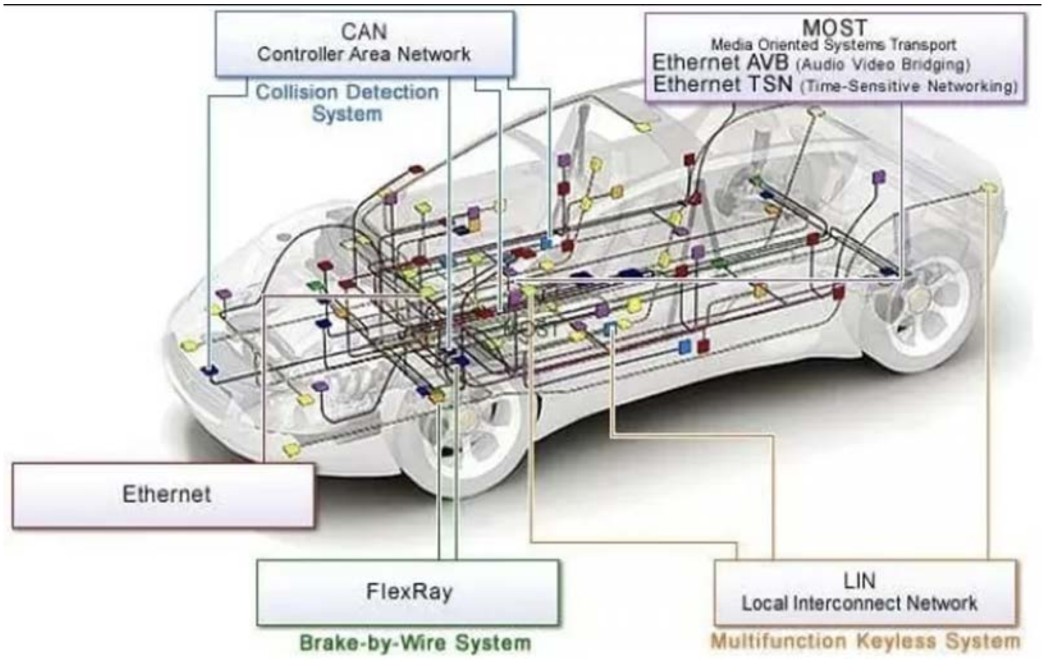
ACRONYM LIST

ACK (Acknowledgment): A signal confirming the successful receipt of a CAN message.
ADAS (Advanced Driver-Assistance Systems): Systems that provide vehicle automation, adaptive features, and enhanced safety.
BMS (Battery Management System): A system that monitors and manages the performance of vehicle batteries, often using CAN communication.
CAN (Controller Area Network): A robust vehicle bus standard designed to facilitate communication among various ECUs.
CAN FD (Controller Area Network Flexible Data-rate): An extension of CAN that supports higher data rates and larger data payloads.
CANH (Controller Area Network High): The high-level voltage signal in a differential CAN signaling system.
CANL (Controller Area Network Low): The low-level voltage signal in a differential CAN signaling system.
CANopen (CAN Application Layer Protocol): A communication protocol used in embedded systems within vehicles.
CRC (Cyclic Redundancy Check): An error-detection method used in CAN messages.
DLC (Data Length Code): A field in the CAN frame indicating the number of bytes in the data field.
ECU (Electronic Control Unit): An embedded system controlling one or more of the electrical systems or subsystems in a vehicle.
EMC (Electromagnetic Compatibility): The ability of electronic systems to operate without interfering with each other, critical in CAN networks.
ESP (Electronic Stability Program): A vehicle control system that improves stability by detecting and reducing loss of traction.
FD (Flexible Data-rate): An enhancement to CAN that allows for increased data transmission rates.
FIFO (First In, First Out): A method of managing the order in which data is processed in a CAN system.
GPS (Global Positioning System): A satellite-based navigation system that may communicate with vehicle ECUs via CAN.
HMI (Human-Machine Interface): The interface that allows interaction between the vehicle’s electronics and the driver.
HVAC (Heating, Ventilation, and Air Conditioning): A vehicle system often controlled through CAN communication.
ID (Identifier): A unique identifier within a CAN frame that indicates the message's priority and content.
J1939: A CAN-based communication protocol used for heavy-duty and commercial vehicles.
LIN (Local Interconnect Network): A communication protocol often used in conjunction with CAN in automotive applications.
LPM (Low Power Mode): A state that reduces power consumption in a CAN network when the vehicle is idle.
MCU (Microcontroller Unit): The primary computing unit in ECUs that processes CAN messages.
OBD (On-Board Diagnostics): A system that monitors vehicle performance and diagnostic data, often via CAN.
OSI (Open Systems Interconnection): A conceptual framework for understanding network protocols, including CAN.
PID (Process Identifier): A code used in OBD systems to request data from vehicle sensors.
PWM (Pulse Width Modulation): A method used to control electrical power in automotive systems, which may be managed via CAN.
RAM (Random Access Memory): Memory used in ECUs to store data for CAN communication.
ROM (Read-Only Memory): Non-volatile memory in ECUs storing the firmware for CAN communication.
RTR (Remote Transmission Request): A CAN frame type that requests data from another node.
SDO (Service Data Object): A CANopen protocol used for device configuration and data transfer.
SOF (Start of Frame): The initial bit in a CAN message indicating the start of a data frame.
SPI (Serial Peripheral Interface): A communication protocol that may be used alongside CAN in automotive systems.
SRR (Substitute Remote Request): A field in extended CAN frames, replacing the RTR field.
TPDO (Transmit Process Data Object): A type of CANopen message used to transmit process data.

Chapter 1 - Network Definition and Type

What is multi-plex?
A system or signal involving simultaneous transmission of several messages along a single channel of communication.

Why do we multiplex? Multiplexing decreases the size and weight of the harness, lowers the cost of production, shares data without redundancy, provides better diagnostics, and allows greater flexibility to add systems.



What Is a Network
A network is a group of modules or nodes connected for information sharing. The advantage of networks is that all modules on the network can share information from a single sensor, which reduces the load on the sensor because it only reports to one module. Modules communicating on the same bus also reduce repetitive wiring and module terminals.

Each module does not need a separate communication wire to talk to all the other modules. Networking improves diagnostic capabilities, and some scan tools can identify all modules on a particular network, which may aid a technician in identifying a module that is not reporting.

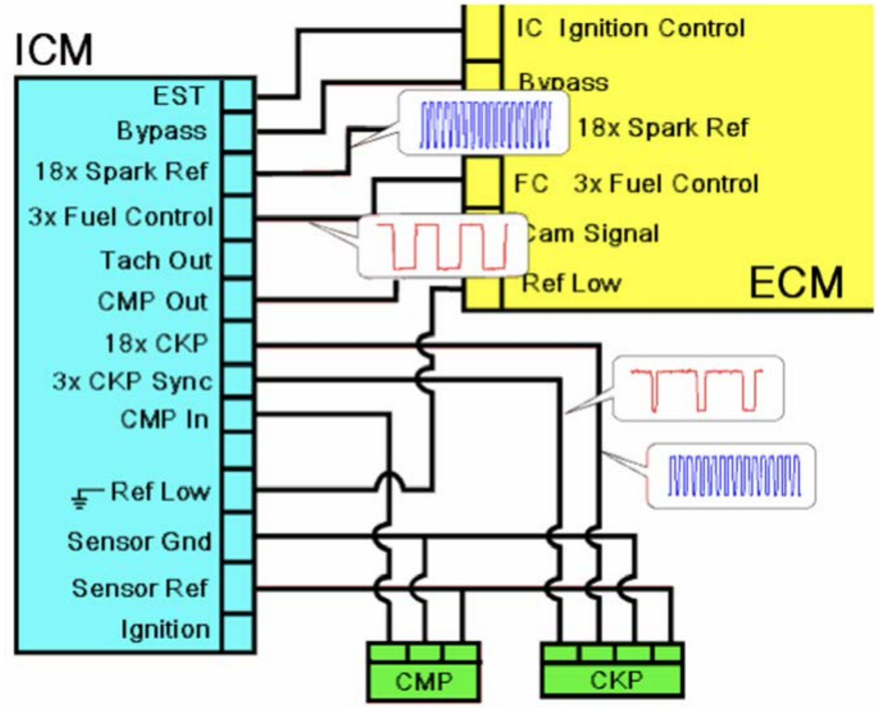
Network information sharing is often referred to as multiplexing, where multiple signals are gathered from different modules, and then the information is compiled and sent out in a message format. Networks use certain communication protocols, which are different for each network. The communication speed varies widely between the different networks.

Here are some exampls of CAN analyzers available on the market.
(https://autotechdrive.com/best-can-analyzer-tools/)

- CANLabsoftware
- CANalyzer
- CanKing
- CANtrace
- Raptor-CAN
- CanEasy
- Easy CAN
- X-Analyzer

Before Networking

In the early days of computerized vehicles, modules did not network together. There was no broadcasting a TPS or MAP sensor readings on the network. Then whichever controllers needed that information could not access it. The signals from sensors were sent to a module, and if another module needed them, the signals were recreated and sent to the next module in need. Sometimes, the sensor would employ another signal wire that could provide a signal for an additional module. In the early days, there were far fewer modules on the vehicle.



Courtesy of flatrater.com

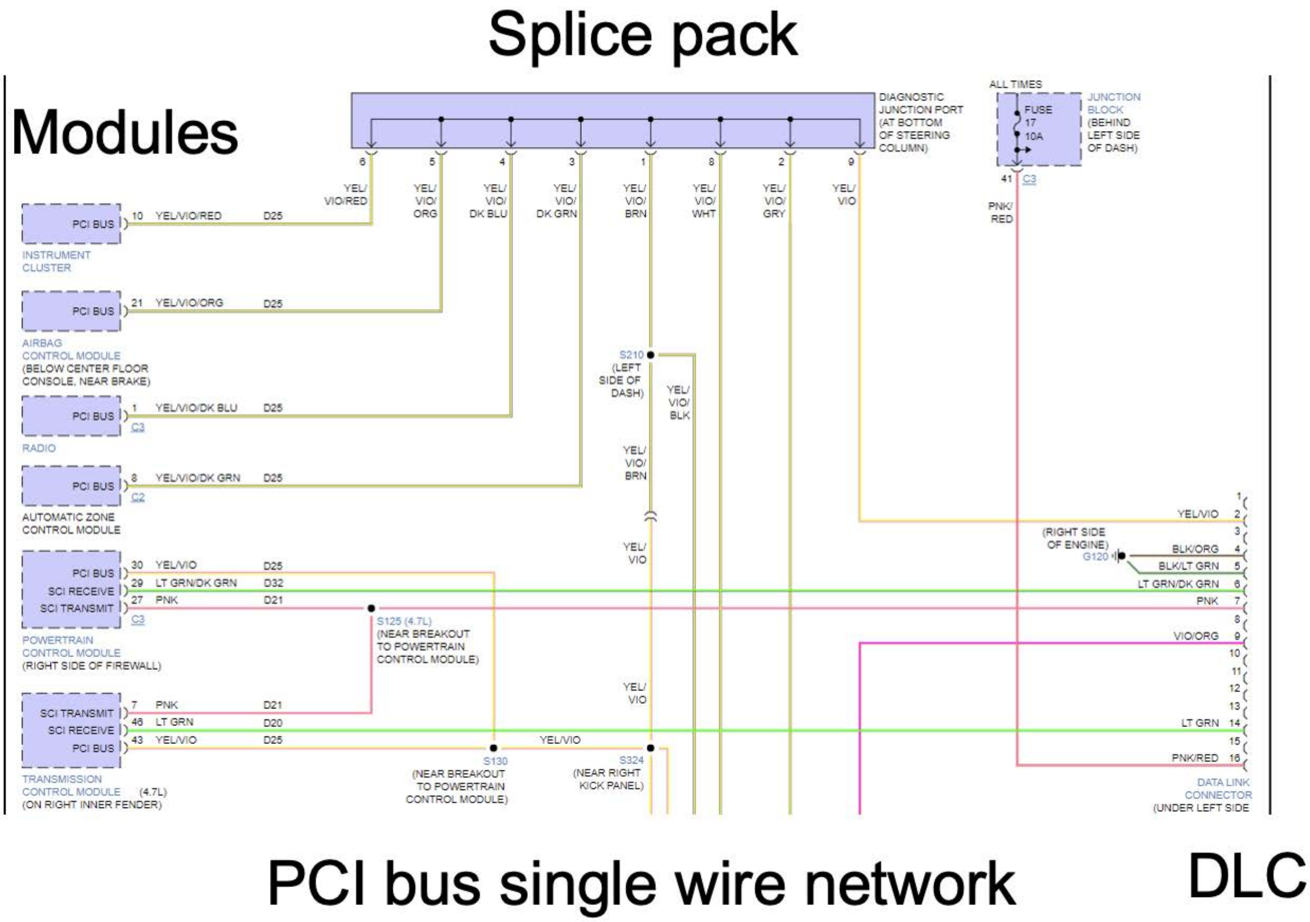
GM Ignition Signals Recreated

Examples of signals being recreated can be found on this 1998 Grand Prix.
In this example, the crank and cam sensors report to the ignition control module (ICM). Instead of broadcasting a message on a bus, the ICM recreates the signals and sends them to the ECM.

In this configuration, the ignition module may develop an issue that may cause distortion or noise in the recreated signals, or the recreated signals may be absent altogether. The crank sensor inputs a 3X and an 18X signal into the ICM, and the ICM recreates them as a 3X and an 18X signal. The ECM can now use these signals to send an EST, electronic spark timing, signal to advance or retard timing.

This ignition system was used on 3800 engines; a reference example would be a 1998 Grand Prix.

How is it supposed to work? When the engine is first cranked over, the
ICM (bypass mode) controls the ignition timing. Upon receiving the second fuel control pulse, the PCM will apply 5v to the bypass circuit, and the ignition control signal will be used to indicate ignition timing. After the vehicle starts, the EST circuit will allow the PCM to control timing.



Network Components

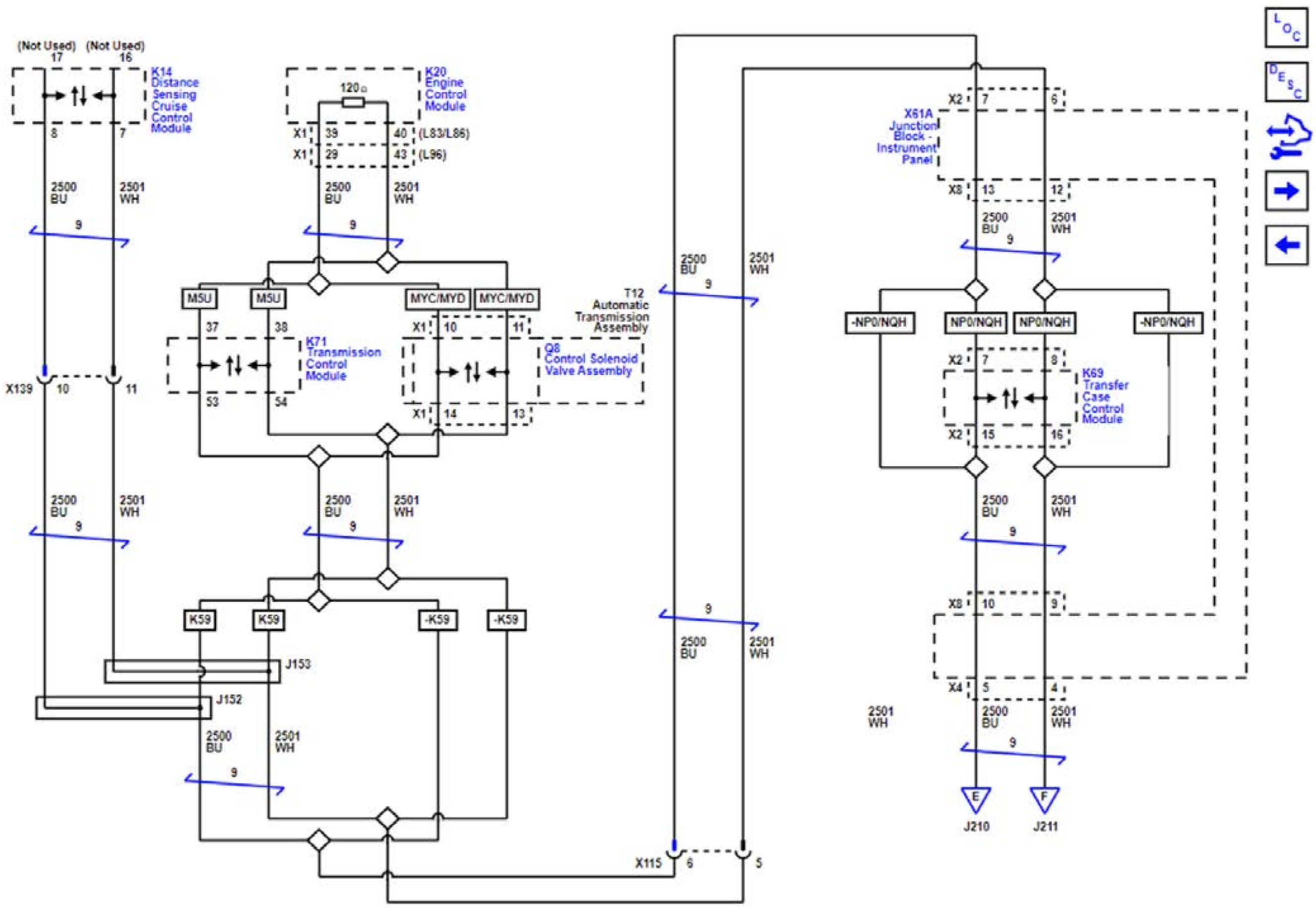
Each network has its configuration. Some standard network components are bus wires, which may be a single wire or a twisted pair, modules, central gateway, splice blocks, and data link connectors. We will discuss each components. The diagram above is from a 2001 Jeep Grand Cherokee with a PCI bus.

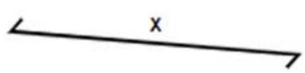
Network Components/Wires
Some networks only use a single wire for data transmission. Some networks use a three-wire configuration. CAN and a few other networks use a twisted pair configuration. The twist typically has 33-50 twists per meter, one every 20-30 mm (.75-1.2 inches). It is important to maintain the twist configuration during repair. The twisted pair typically terminates in adjacent cavities at the connector. The twisted pair prevents electrical interference that can interfere with the communication signals. It is always best to route communication wires away from ignition coils, injectors, starter cables, solenoids, or any devices that can create electromagnetic interference (EMI).

You can buy twisted pair wire from this site, but it comes at a five-meter minimum.
https://www.dsgarage.it/store/en/wire/166-can-bus-twisted-pair-cable-035mm-22awg.html

Twist Wire
Wire repair may require twisting up a set of wires. Place one end of the two wires in a holding fixture and place the other end in a drill. Twist the wires to obtain a twist every .75-1.2 inches.

- A drill and a holding device
- Place wire ends into holding device
- Place other ends into drill
- Twist wire until desired twist rate is achieved



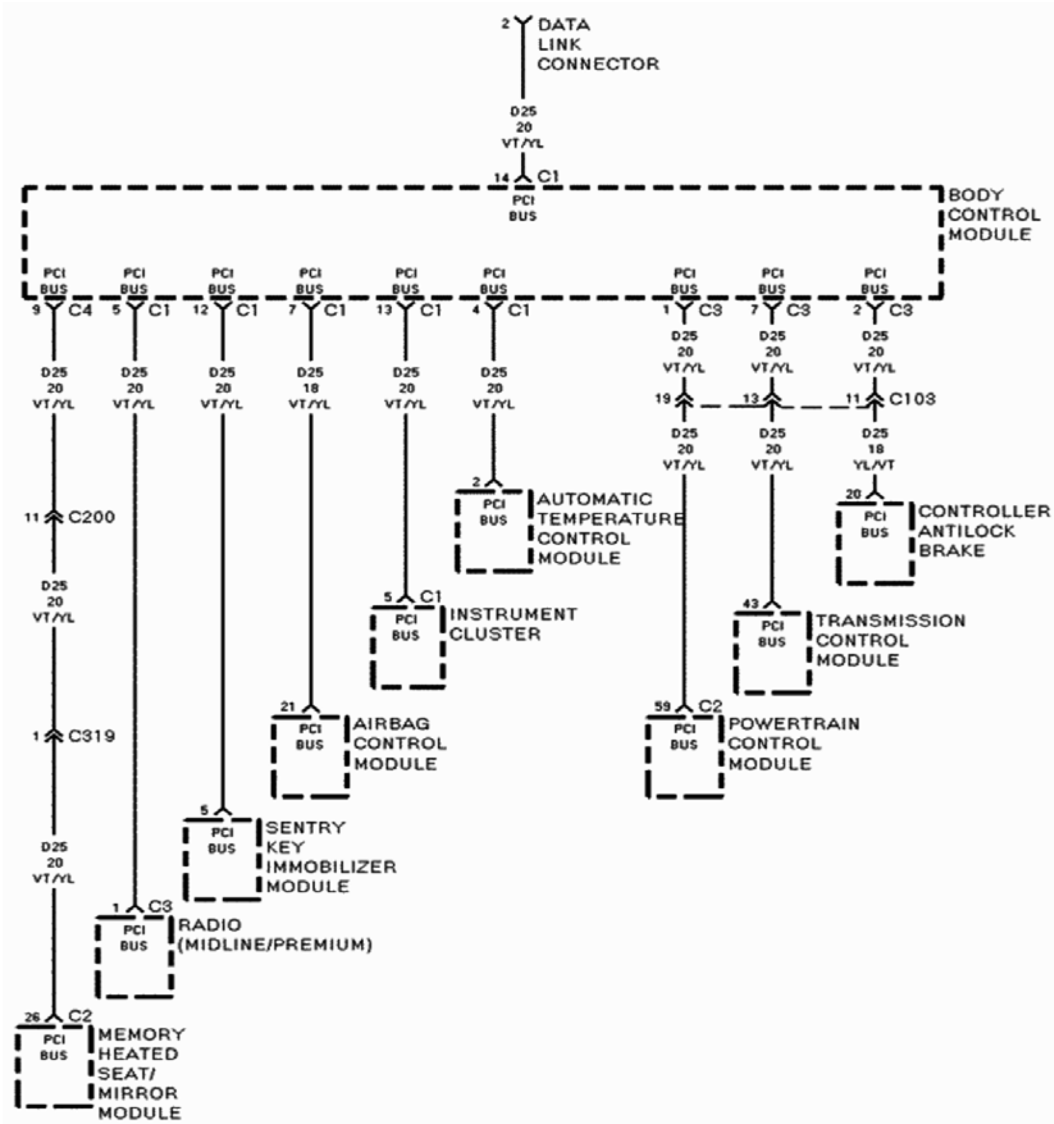


Note: Twisted wires provide an effective shield that helps protect sensitive electronic components from electrical interference.

In order to prevent electrical interference from degrading the performance of the connected components, you must maintain the proper specification when making any repairs to the twisted wires shown :

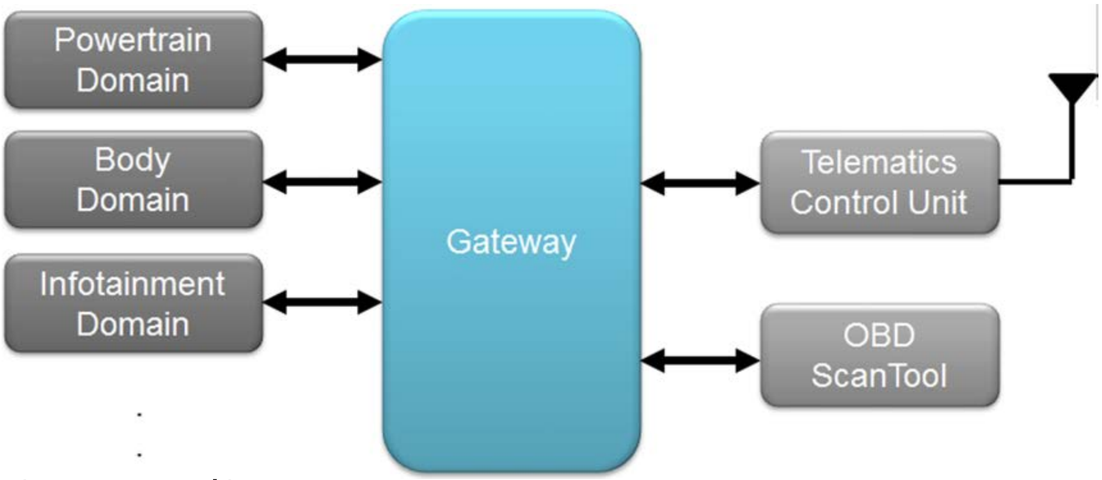
- The wires must be twisted a minimum of 10 turns per 31 cm (12 in) as measured anywhere along the length of the wires.
- The outside diameter of the twisted wires must not exceed 6.0 mm (0.25 in).

Example of Twisted Wires
2017 Cadillac Escalade used this to show the twists per foot shown in the diagram.



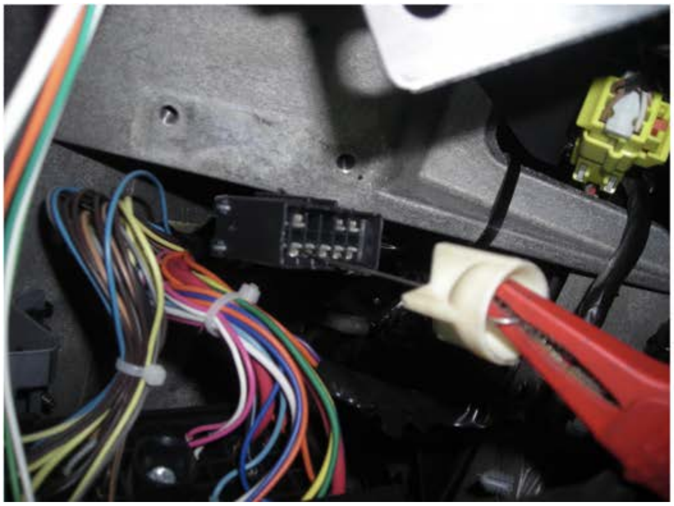
Network Components/Module

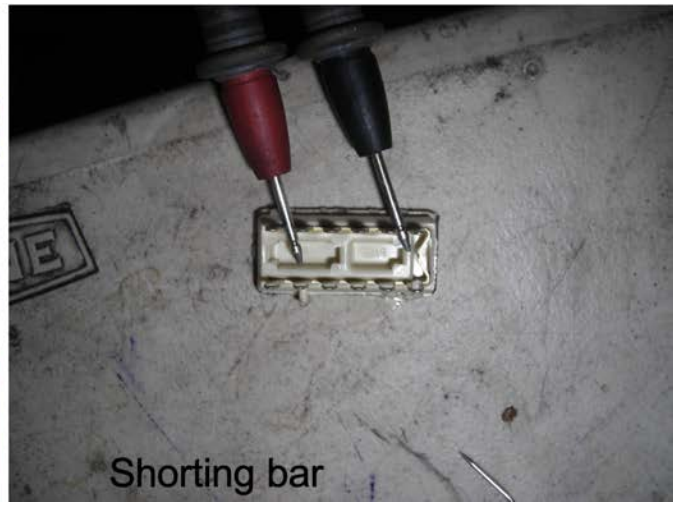
ECUs, modules, and Nodes are essentially the same devices. The names for this manual will be interchangeable. Each network has many modules connected to the network. Most networks have a limit to the number of modules it can support. The diagram above is from Chrysler's PCI bus. The PCI bus was phased into Chrysler vehicles in 1998 and was used for many years. The PCI bus is a one-wire network following the OBD J1850 variable pulse width requirements. The PCI bus operated at 10,400 bits per second. This network could only support up to 32 modules.



Network Components/Gateway

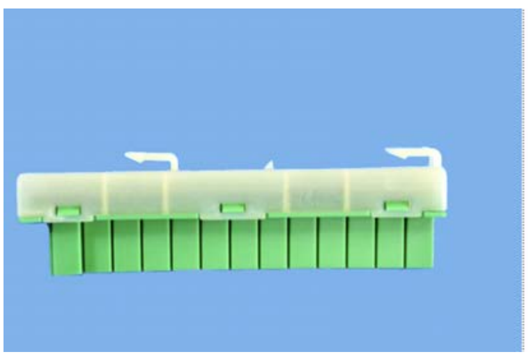
There are multiple different networks on modern vehicles. These networks may operate at different voltage levels and speeds and may be a single or a dual pair of wires. Due to the differences in the networks, they cannot communicate directly. A central gateway is needed to allow different networks to exchange information. The gateway also provides access to the various networks from the scan tool. The gateway provides the VIN to the various networks to ensure all modules share the exact VIN. The gateway houses the vehicle build configuration, so when replacing the gateway, the build configuration must be saved in an existing module or programmed into the new gateway.





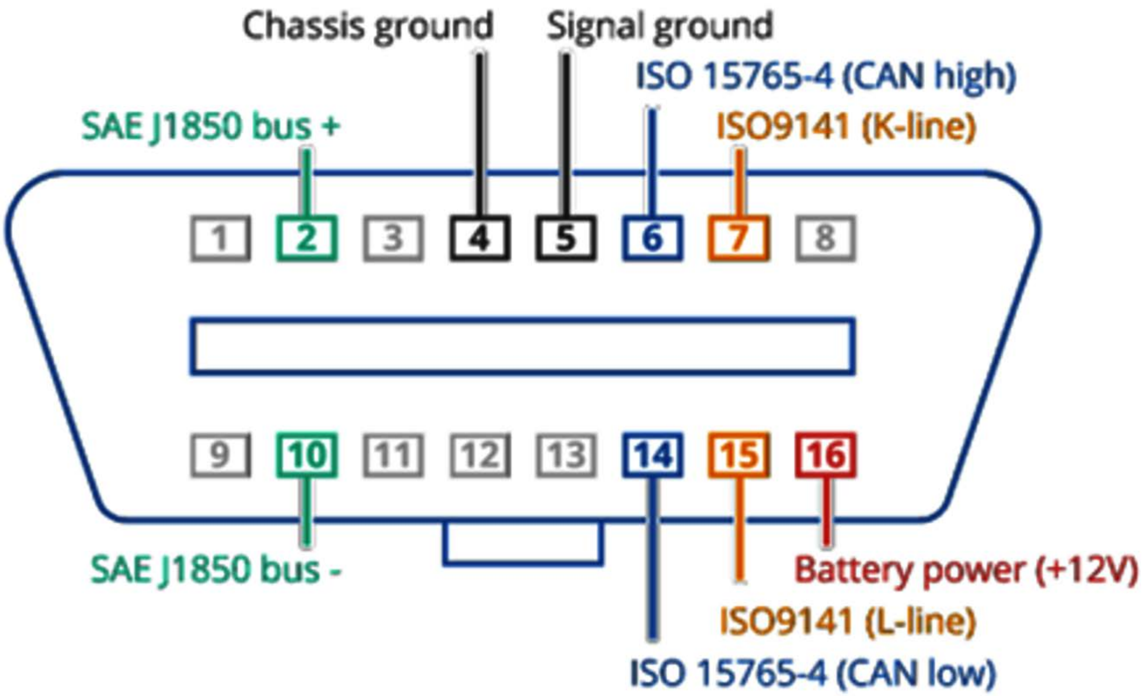
Network Components/Splice Block

A splice pack may connect multiple modules on a single wire network. A splice pack is typically a connector, as shown in the top picture, with a shorting bar in the lower picture. When tested with an ohm meter, the shorting bar should carry zero resistance between the terminals. The shorting bar is inserted into the connector, connecting all modules on the network. The shorting bar can be removed to test each network separately. The top picture shows the shorting bar removed, and one network is being tested with a lab scope.



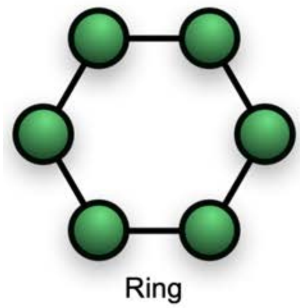
Chrysler Star Connectors

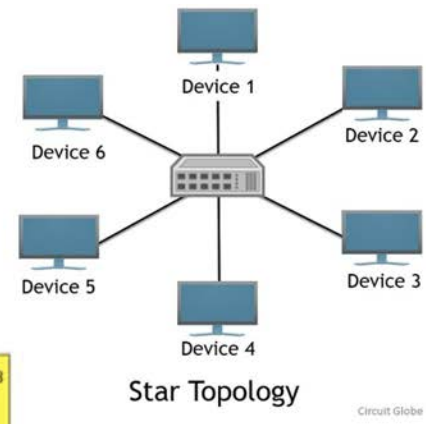
These are used for CAN and are a great diagnostic starting/ending point. The 120-ohm resistor is often hidden in these.

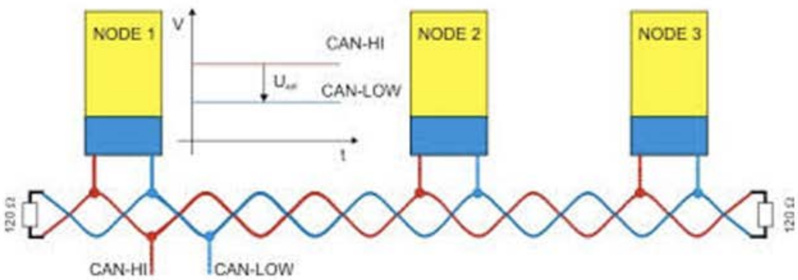


Network Components/DLC

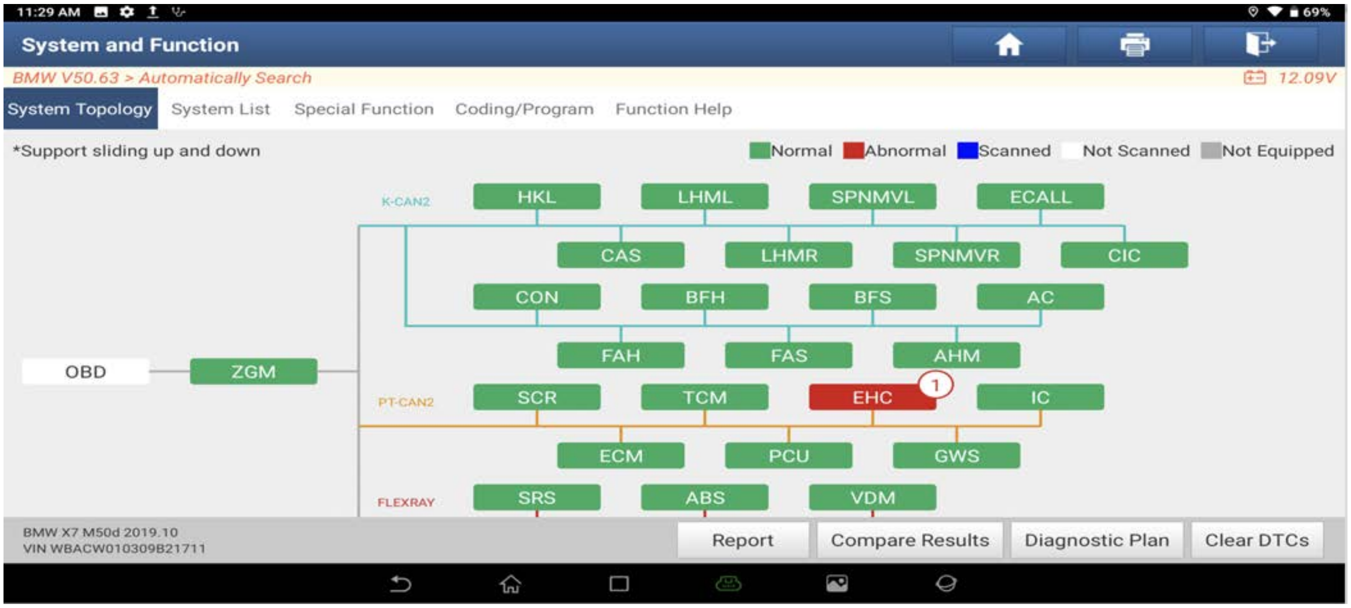
The data link connector (DLC) has standardized scanner connector locations and some pin assignments. The DLC allows scan tool access to various networks. The DLC also provides a power source and ground for a scan tool. In the example above, pins 2 and 10 give access to the J1850 network for a Ford. Chrysler and GM would use pin 2 for a single-wire J1850 VPW network. Pin 4 is chassis ground, and pin 5 is signal ground. Pins 6 and 14 allow access to the CAN network on most vehicles, but in late-model Chryslers, it would only give access to the secure gateway module and not the complete CAN network. Pin 7 and 15 would give access to the ISO9141 protocol, typically where a generic scanner would gain access. Pin 16 is battery power.







Topology
Topology refers to the way the modules are laid out and wired together. The above is the most common, but this is not a complete list. There are variations of this topology. The ring topology of all modules is arranged in a ring or circle. The information must go through each module to get to the next module. The star topology of the modules is wired to a central point called a master; if one module fails, it should not affect another module's communication. The problem with the star configuration is that if the master fails, every module will be out of communication. The bus topology's modules are wired directly to the bus. In the bus configuration, if a wire is broken to a module, it only affects communication with that module, and the rest of the modules may continue to communicate.



Topology Scan Tool

Launch, TopDon, Autel, and OEM scan tools all show topology.

BUS	CAN	LIN	FLEXRAY	MOST	ETHERNET
Speed	Up to 1000 kbps	Up to 19.2 kbps	10 Mbps	Up to 23 Mbps	Up to Gbps
Cable Type	Twisted Pair, 5v	Single wire, 12v	2 or 4 wires	Fiber Optic / Coax	One or more twisted pair
Cost	$$	$	$$$	$$$$	$$
Applications	ABS, Powertrain, Engine Control	Electric Seats, Mirrors, Tailgate	Steering, Traction control, Active suspension	Media Players, Infotainment	IP Cameras, Radar, Infotainment

Network Type & Speed

Manufacturers have used many different networks over the years. Some early networks included class 2 for GM, Standard Corporate Protocol for Ford, and PCI bus for Chrysler. These networks have been phased out in favor of faster networks like CAN. Some standard networks we see on modern vehicles are listed above. A few standard networks are controller area network (CAN), CAN flexible data rate (FD), local interconnect network (LIN), FlexRay, and media-oriented system transport (MOST). We will focus on the CAN network and offer some tips on diagnosing network issues.

Bits and Bytes

Bit: A bit is a binary digit, the smallest increment of data on a computer. A bit can hold only one of two values: 0 or 1, corresponding to the electrical values of off or on, respectively. Because bits are so small, you rarely work with information one bit at a time.

Byte: One byte is equal to 8 bits.

Assessment

Questions

Question: T or F It is not necessary to use twisted wire on a CAN communication line.

Question: T or F A computer on the network may be called a node or a module.

Question: T or F A splice pack is typically used on CAN networks.

Question: T or F It is not necessary to use twisted wire on a CAN communication line.
Answer: False, the twisted wire helps reduce noise entering or leaving the communication lines.

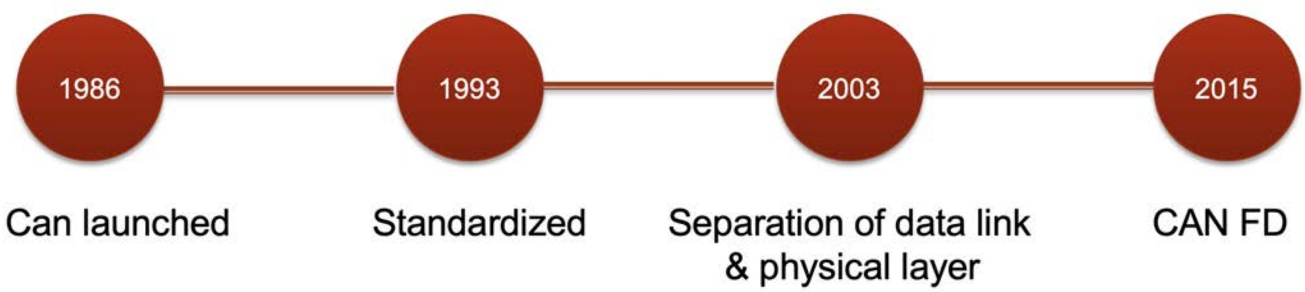
Question: T or F A computer on the network may be called a node or a module.
Answer: True

Question: T or F A splice pack is typically used on CAN networks.
Answer: False, Splice packs are used on one wire networks.

Chapter 2 - CAN C High Speed

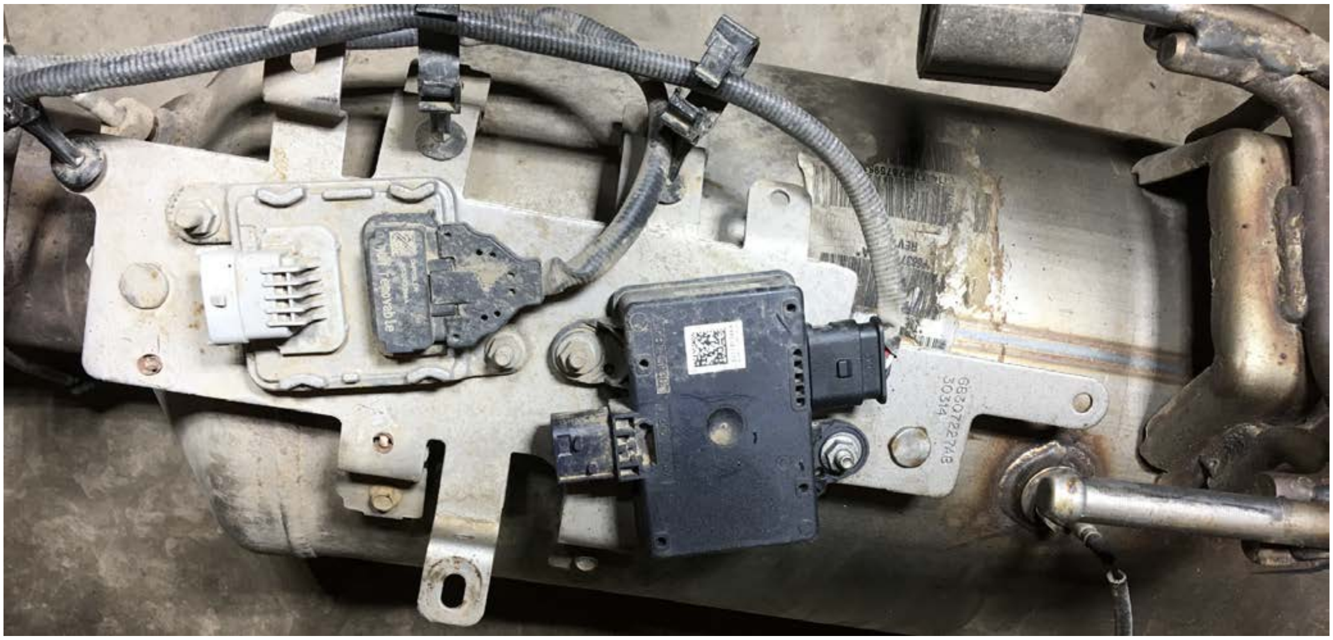
CAN Examples

CAN has many uses throughout many industries. A few of the applications are NMEA 2000, a marine application. CAN is also used in industrial automation, medical equipment, Aerospace, and automotive. Also, the J1939 CAN network is used in heavy trucks and equipment. The focus of this class is the CAN data bus for the automotive industry.



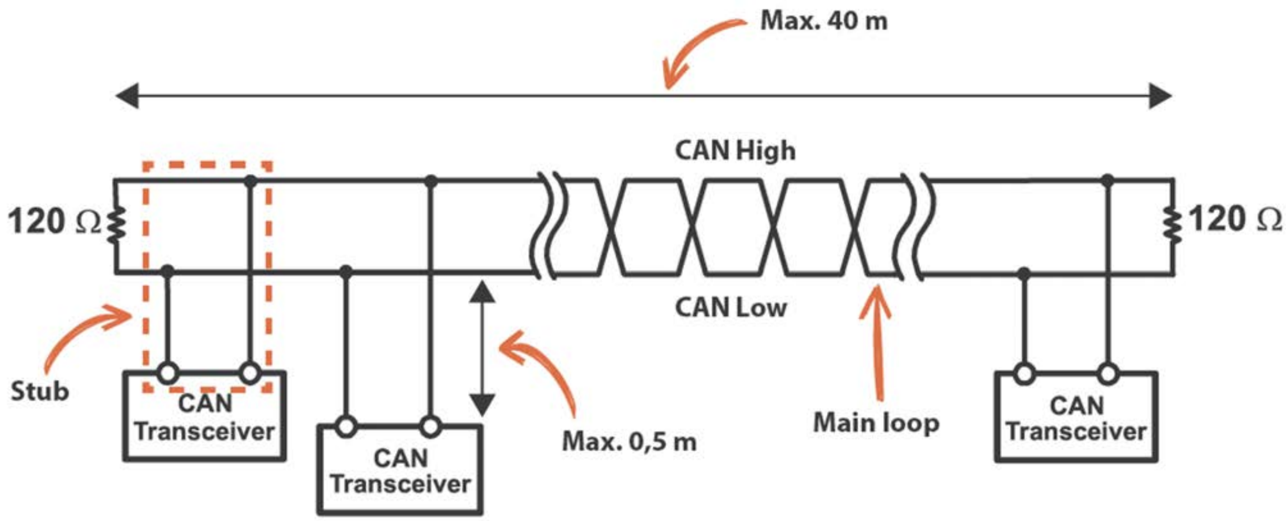
CAN History

SAE and ISO created standards for speed and physical layers of the CAN network. CAN protocol was divided into three distinct categories: CAN A, CAN B, and CAN C. All CAN networks operate at different speeds, but all three networks may reside in the exact same vehicle. The CAN standard is called ISO 15765. Bosch developed CAN in 1983 and launched it in 1986. In 1991, CAN 2.0A 11-bit and 2.0B 29-bit were published. Mercedes S class installed the CAN bus in 1991. In 1993, the CAN high-speed transceiver was adopted as an international standard (ISO 11898). In 2003, ISO 11898 became a standard series. In 2012, the CAN FD protocol was released and was standardized in 2015. In 2016, the physical CAN layer can handle up to 5Mbps, so CAN should stay relevant for quite some time. Federal law requires all 2008 and later passenger vehicles to use the CAN protocol to communicate with emission-related modules.



CAN Modules

There are two basic types of CAN modules: integrated and stand-alone. The stand-alone module only passes data through it with no ability to store data. Examples of this type of module are the O2 sensor, NOX, and window modules. The stand-alone module only passes information through and acknowledges messages on the network. An integrated module is an intelligent module that can store data; an example is the PCM, TCM, and ABS modules.

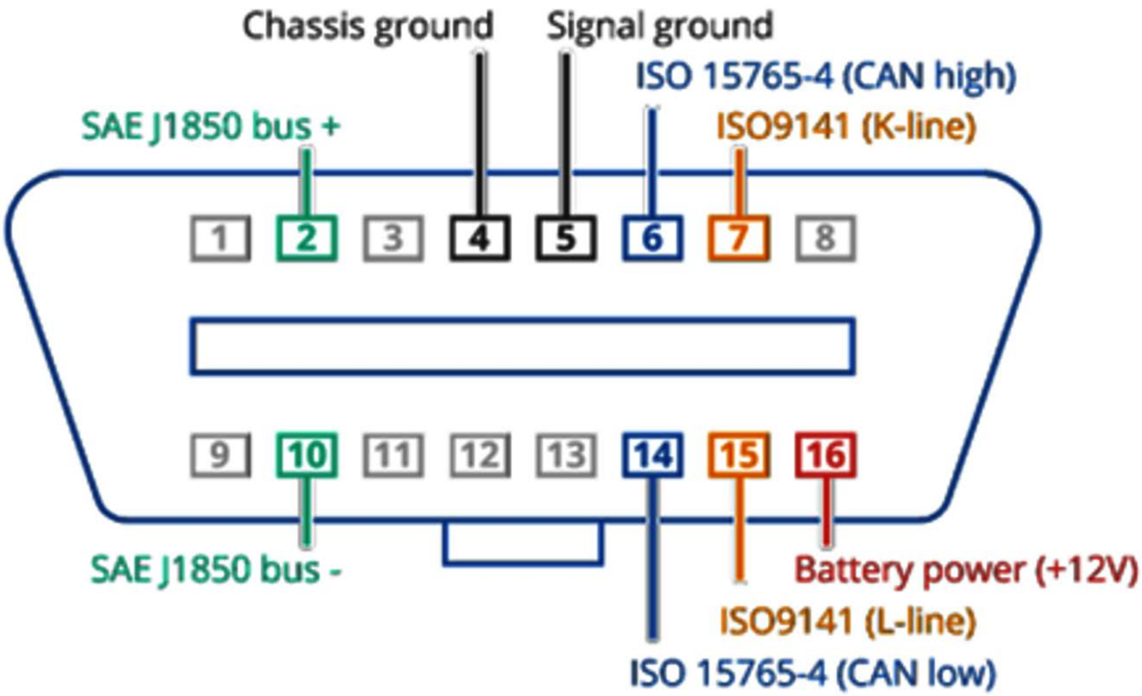


CAN C Bus Construction
The physical layer of the CAN bus is a twisted pair of wires that should not exceed 40 meters in length, and a maximum of 30 modules can exist on the bus. The twisted pair is referred to as the backbone or trunk cable. Each module is wired to the bus through a stub, sometimes called a drop cable. The stub length is recommended not to exceed 30 centimeters, but 50 centimeters is the max. CAN C high speed is not fault-tolerant; all communication stops if one line goes down.

Termination Resistance
There is a 120-ohm resistor at each end of the CAN wires. The resistors may reside inside a module or outside a module. The total resistance of the CAN circuit is 60 ohms. The resistance of a CAN network can be measured when the bus is off, and no bias voltage exists. The termination resistors prevent the signal from reflecting from one line to another, causing a ghost image. The transmitter drives the signal of the 2.5-volt bias in both directions and the termination resistors help return the bus to 2.5 volts.

Measure Bus Resistance

How should we correctly measure CAN bus resistance? Look at a wiring diagram to ensure you access the CAN bus at pins 6 and 14 at the DLC. Use a breakout box (BOB) at the DLC to reduce terminal damage. Some vehicles, like newer Chryslers, use a secure gateway, and you cannot gain access to the can bus at the DLC. Vehicles with secure gateways require bus access at a participating module.



CAN Differential Voltage

The can network is available on pins 6 and 14 on most vehicles. Pin 6 is CAN high (H), and Pin 14 is CAN low (L). Both CAN legs are biased to 2.5 volts. CAN H will increase voltage, and CAN L will decrease when the bus is active. Industry documentation states that CAN H can travel upward to 3.75 volts and CAN L can travel downward to 1.25 volts, but not every vehicle follows these voltage levels strictly. The important thing is that there is a voltage difference between CAN H and CAN L when the bus is active and no voltage difference when the bus is idle. The voltage differential is how the binary numbers are created. When the voltage is the same at 2.5 volts, meaning there is no volt difference between the two can lines, a recessive bit is created, which is a one. A dominant bit is created when the voltage difference is approximately 2.0 volts, which is a zero.

CAN H travels upwards to 3.5 volts commonly, and CAN L travels downwards to 1.5 volts commonly. Each signal is the exact opposite of the other, happening at the same time. If you invert one of the signals on your lab scope when viewing it, it should be an exact overlay of each other.

Controller Area Network (CAN)

CAN has three different speeds. CAN A is a single wire network and communicates at 33.3 Kbps (88.3 Kbps in high-speed mode). CAN B is a two-wire network and communicates at 125 Kbps. CAN C is a two-wire network and communicates up to one Mbps, but typically operates around 500 Kbps. CAN C is not fault tolerant if one wire goes down the whole system goes down. CAN B is fault tolerant if one wire goes down the system can still communicate in this system each module will include a termination resistor.

Max message that bus can handle. Class A-Less than 10 messages a second, Class B-Less than 50 messages a second, andClass C-Less than 200 messages a second. CAN data is transferred using frames.

Frames

Messages are sent on the CAN bus using frames. Three frames are used: the data frame, the remote frame, and the error frame. The data frame is used 99% of the time to transmit messages from module to module. The remote frame is a request from a module for data. The error frame is transmitted when a device detects a fault.

Bidirectional Communication

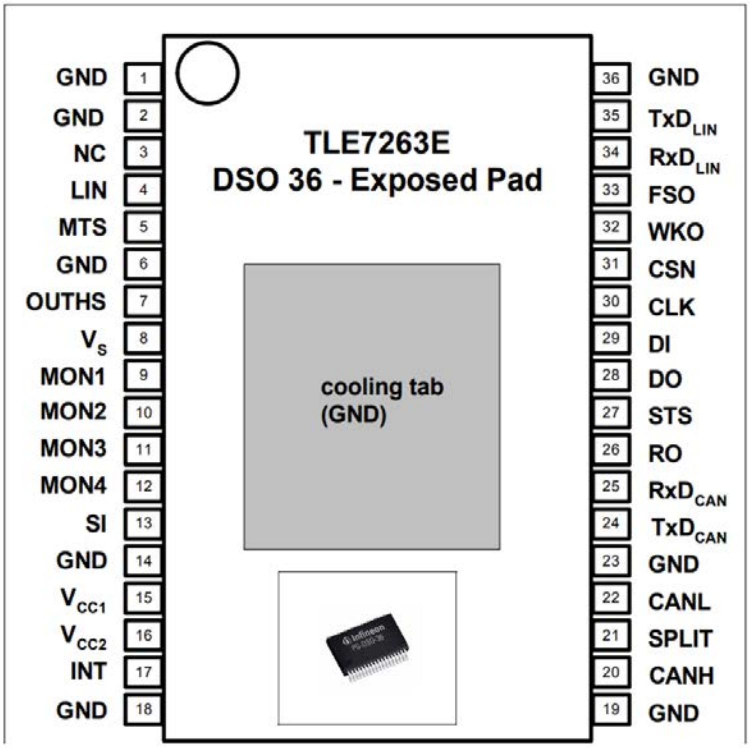
Modules need bidirectional communication; each module must be able to send and receive messages. To operate bidirectionally, each module must take the role of a transmitter and receiver. When a transmitter and a receiver are combined together they are referred to as a transceiver. Each module is connected with wires, which are referred to as a medium.

Parameter	Symbol	Limit Values			Unit	Test Condition
		Min.	Typ.	Max.		
Dynamic CAN-Transceiver Characteristics						
Propagation delay TxD-to-RxD LOW (recessive to dominant)	$t_{d(L),TR}$	–	150	255	ns	C_L = 47 pF; R_L = 60 Ω; $V_{CC1/2}$ = 5 V; C_{RxD} = 20 pF
Propagation delay TxD-to-RxD HIGH (dominant to recessive)	$t_{d(H),TR}$	–	150	255	ns	C_L = 47 pF; R_L = 60 Ω; $V_{CC1/2}$ = 5 V; C_{RxD} = 20 pF
Propagation delay TxD LOW to bus dominant	$t_{d(L),T}$	–	50	120	ns	C_L = 47 pF; R_L = 60 Ω; $V_{CC1/2}$ = 5 V
Propagation delay TxD HIGH to bus recessive	$t_{d(H),T}$	–	50	120	ns	C_L = 47 pF; R_L = 60 Ω; $V_{CC1/2}$ = 5 V
Propagation delay bus dominant to RxD LOW	$t_{d(L),R}$	–	100	135	ns	C_L = 47 pF; R_L = 60 Ω; $V_{CC1/2}$ = 5 V; C_{RxD} = 20 pF
Propagation delay bus recessive to RxD HIGH	$t_{d(H),R}$	–	100	135	ns	C_L = 47 pF; R_L = 60 Ω; $V_{CC1/2}$ = 5 V; C_{RxD} = 20 pF
Min. dominant time for bus wake-up	t_{WU}	1	3	5	µs	–
TxD permanent dominant disable time	t_{TxD}	0.3	–	1.0	ms	–

Bit Propagation Delay

Timing is critical in bit transitions. Bit timing is checked and is referred to as propagation delay. Propagation delay changes depending on the actions being checked. The spec sheet for this transceiver lists all the propagating delays. Depending on the actions, the propagation delays range from 50 to 255 nanoseconds.

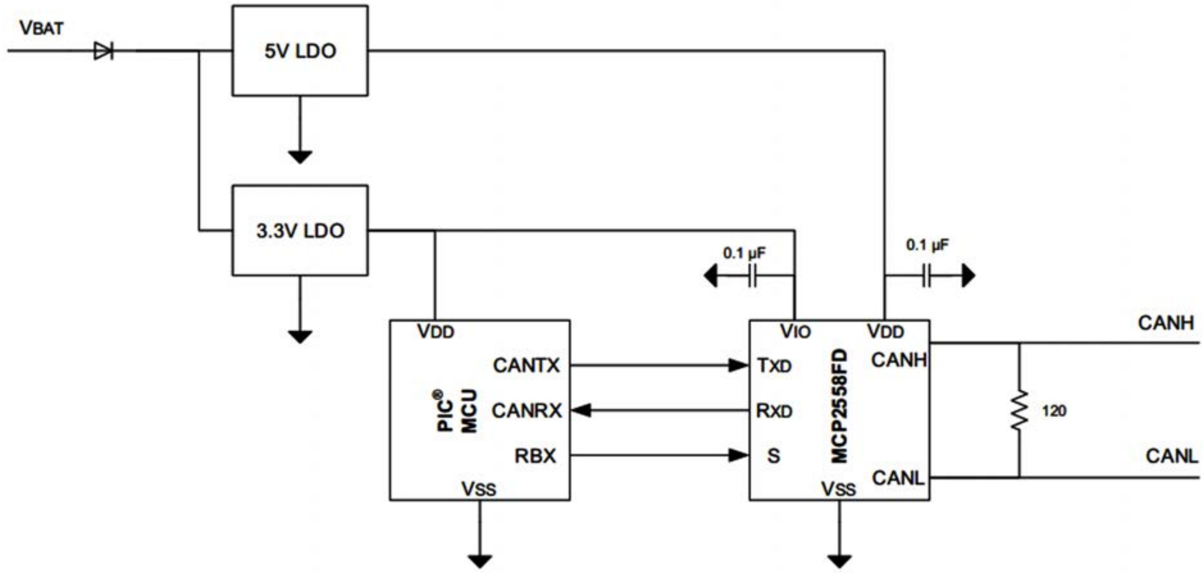
2 Pin Configuration



Transceiver Chip Pin Layout

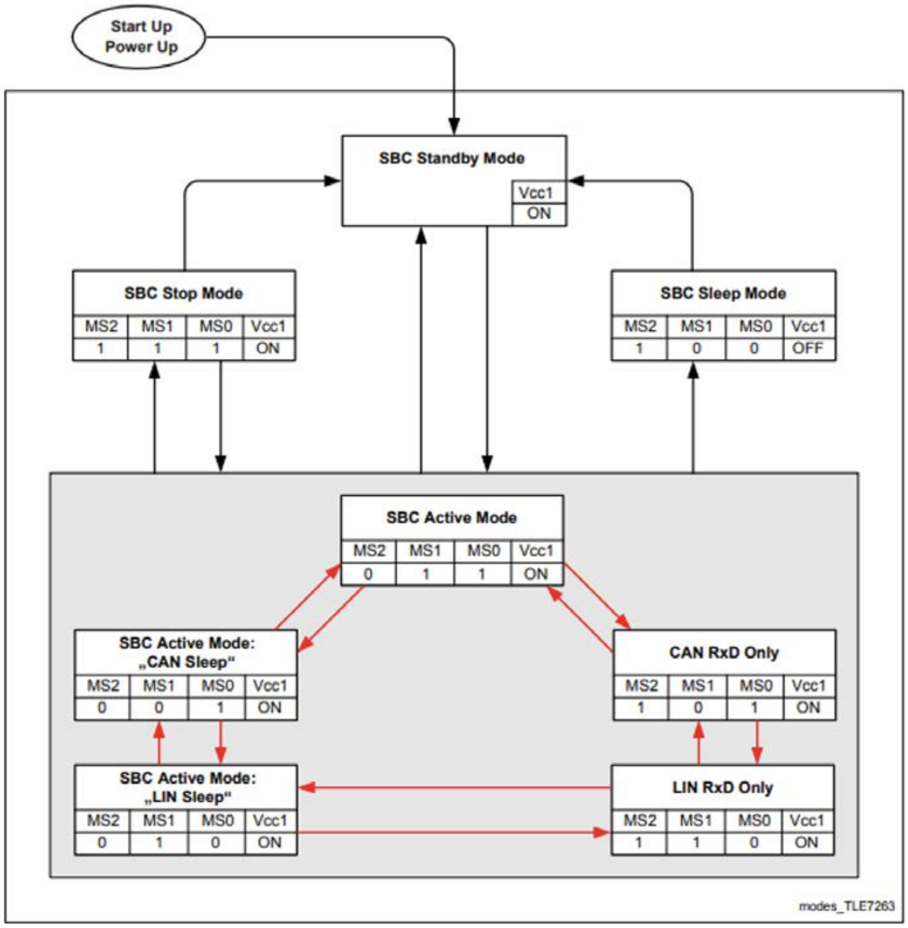
Transceiver example Infineon TLE 7263E with PG-DSO 36 chip. The TLE 7263E is a monolithic integrated circuit (IC). An IC, sometimes called a chip or microchip, consists of a semiconductor wafer on which thousands or millions of tiny resistors, capacitors, diodes, and transistors reside. This transceiver covers smart functions like high-speed CAN and LIN bus communication. The transceiver interfaces between the CAN protocol controller and the physical bus lines. Signals are modulated by the node (1's & 0's) and driven by the transceiver. The transceiver drives the signals high and low to the bus line.

This transceiver supports low power mode and becomes awake from a message on the network or bi-level sensitive wake-up inputs. Pins 9,10,11,12 monitor bi-level wake-up inputs. Pin 8 is the power supply isolated from a short to ground with a ceramic capacitor. Pin 15 & 16 is a 5-volt supply output. Pin 32 is the wake-up event output, indicating wake-up monitoring input via CAN & LIN bus. Pin 34 is the LIN transceiver data output. Pin 35 LIN transceiver data input. Pin 4 is the LIN bus line. Pin 24 CAN transmit data input (integrated pull-up). Pin 25 CAN receive data output. Pin 20 is CAN high output. Pin 22 is CAN low-line input.



Transceiver Operation
The RDX is the received signal, constantly reading what is on the CAN bus and sending it back to the transceiver. The RDX also takes into consideration the signal propagation delay through the transceiver. Using RXD allows the sending transceiver to determine message errors in the message it is sending, and it also allows the transceiver to determine when it has lost arbitration.

TXD is used for transmitting. It is connected to the transceiver output and controls the CAN state on the bus. When TXD is low, the CAN bus is in the dominant state. When TXD is high, the CAN bus is in the recessive state. Typically, the CAN L bus fails first due to the extra duty or load the Low bus is under.

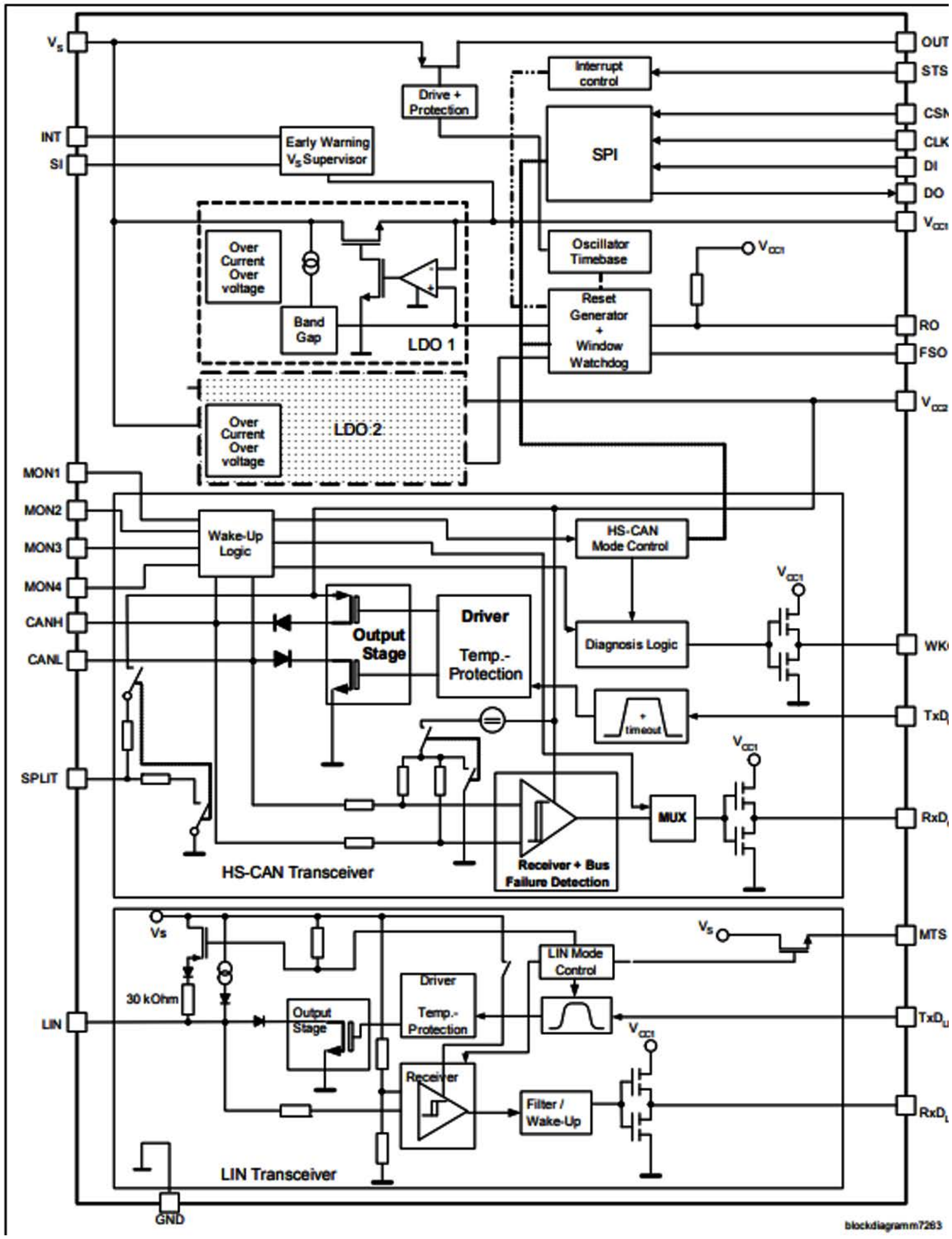


Transceiver Modes of Operation

The transceiver system basic chip (SBC) has five main operating modes. They are active mode, standby mode, sleep mode, stop mode, and CAN receive-only mode. After startup, the SBC enters standby mode, waiting for the microcontroller to finish the initialization sequence. The SBC can be switched from standby mode to the desired operation mode. Active mode is when the SBC sends and receives messages. The active mode has two sub-modes: CAN and LIN sleep. SBC sleep mode offers the lowest power consumption and can wake up anytime. SBC stop mode for both CAN & LIN bus lines is turned off.

CAN receive only mode, and the SBC is simply listening. A time-out feature is incorporated to prevent the bus from being blocked by a permanent dominant TXD input signal. If a dominant signal lasts 1175-3700 microseconds, the SBC will take a dominant time out. Both CAN L CAN H output stages are automatically turned off after the delay time, and the bus will return to its recessive state. The time-out feature allows the offending module to take itself off the bus so the other modules can continue communicating.

To check for this event, look at the last signal and measure the time it spent dominant. Current limiting circuits are integrated to protect the CAN H & CAN L lines when CAN lines are shorted. A temperature sensor will also turn off the CAN output stage when the junction temperature exceeds the maximum temperature.



Transceiver Block Diagram

The RXD output reflects the differential bus voltage between CAN H & CAN L. The differential voltage will either be approximately two volts dominant or zero volts recessive.

Character	Binary Code	Character	Binary Code	Character	Binary Code	Character	Binary Code	Character	Binary Code
A	01000001	Q	01010001	g	01100111	w	01110111	-	00101101
B	01000010	R	01010010	h	01101000	x	01111000	.	00101110
C	01000011	S	01010011	i	01101001	y	01111001	/	00101111
D	01000100	T	01010100	j	01101010	z	01111010	0	00110000
E	01000101	U	01010101	k	01101011	!	00100001	1	00110001
F	01000110	V	01010110	l	01101100	"	00100010	2	00110010
G	01000111	W	01010111	m	01101101	#	00100011	3	00110011
H	01001000	X	01011000	n	01101110	$	00100100	4	00110100
I	01001001	Y	01011001	o	01101111	%	00100101	5	00110101
J	01001010	Z	01011010	p	01110000	&	00100110	6	00110110
K	01001011	a	01100001	q	01110001	'	00100111	7	00110111
L	01001100	b	01100010	r	01110010	(	00101000	8	00111000
M	01001101	c	01100011	s	01110011	)	00101001	9	00111001
N	01001110	d	01100100	t	01110100	*	00101010	?	00111111
O	01001111	e	01100101	u	01110101	+	00101011	@	01000000
P	01010000	f	01100110	v	01110110	,	00101100	_	01011111

Message Transmitted

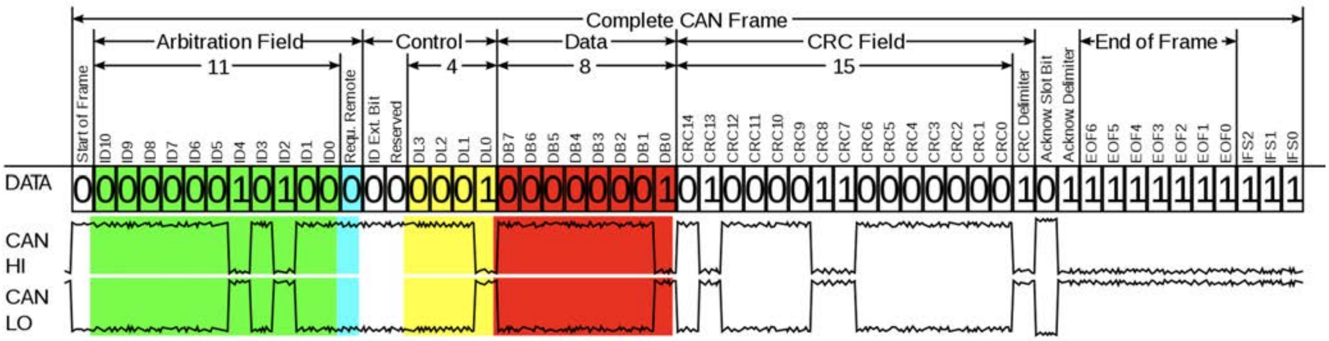
Early messages were created using Morse code, a series of long and short pulses to transmit information. Binary code is used in today's automobiles to transmit information. Messages are transmitted via a binary code. A binary code consists of 1's and 0's. 1's and 0's are formed by voltage pulses. The 1's and 0's can form letters and numbers, as the chart displays above. Arranging and reading bits in ordered groups allows large amounts of data to be transmitted quickly.

Binary code is converted to HEX by the module and then converted to decimal to represent a data PID sent to the scan tool. Modules can receive and transmit binary code. A module on the network is exposed to all the messages on the bus, but each module can sort the messages meant for them and disregard the other messages.

Message ID

Frames of data handle communication on the bus. The frames are structured in a particular way. The CAN module's topology is in a bus configuration. Each module on the bus has access to the data; however, the messages are coded to specific modules on the bus. When a module receives a message, it is required to acknowledge that the message was received. CAN 2.0A has an 11-bit identifier format, and CAN 2.0B is a 29-bit frame identifier. The 11-bit identifier contains the message's priority and identification; 29-bit adds information and coding.

Again, a bit is a one or a zero. If we string eight bits together, it becomes a byte of data.



Bit Creation

The voltage fluctuations of the can bus create the binary code of 0's and 1's. CAN H has a baseline of 2.5 volts and CAN L also has a baseline of 2.5 volts. However, CAN H and CAN L mirror each other during communication as CAN H goes high and CAN L goes low at precisely the same time. CAN H voltage switches high to approximately 3.5-3.75 volts, and CAN L switches low to approximately 1.5-1.25 volts. Each edge of CAN H and CAN L should match. When CAN H goes high and CAN L goes low, a voltage differential of approximately 2.0 volts is created, and a dominant 0 bit is created.

When CAN H and CAN L return to 2.5 volts, there is no voltage differential between the two channels, and a 1 is created. If CAN H stays high and CAN L stays low, a series of 0's will be created. If CAN H stays at 2.5 volts and CAN L stays at 2.5 volts, a series of ones are created during the message. When no messages are being transmitted the bus remains at idle, which again, is when both lines are at 2.5 volts. An easy way of determining a zero is simply stating that when a 2.0-2.5-volt differential exists between CAN H and CAN L a dominant 0 is created. When the voltage differential between CAN H and CAN L is zero during the message, a recessive 1 is created. Again, a 0 is a dominant bit, and a 1 is a recessive bit. The bit transmission is also referred to as the data link layer.

Frame Structure
The start of frame (SOF) dominant zero tells the other modules that a module intends to talk. The ID is the frame identifier, the lower the value, the higher the priority.

Identifier
The identifier portion is either an 11-bit or an extended 29-bit. The identifier field contains the message priority and the module for which the messages are earmarked. 11 bits contain 2047 decimal possibilities, and 29 bits can contain over 536 million decimal possibilities. You can use a binary-to-decimal calculator to convert binary to decimal. Input 11 ones into the binary calculator, and it will convert to 2047 in decimal.

Frame Structure
Remote Transmission Request (RTR) indicates if a module sends data or requests dedicated data from another module.

The control contains the identifier extension bit (IDE), a dominant zero on 11-bit, and the four-bit data length code (DLC). The data length code indicates how many bytes will be transmitted up to eight bytes on 11-bit. The identifier field indicates which module the message was meant for.

Data contains the payload, which is the message. The data can contain up to 8 bytes, which is 64 bits. The data field can be dissected; each signal can be extracted and interpreted into the database file. A database file would contain detailed information to identify the message. The database information is OE specific, and without having access to the database file, an individual could not extract the message.

The binary numbers in the frame can be converted to a hex number and analyzed by a Pico scope to check for a corrupt message.

Frame Structure

Cycle Redundancy Check (CRC) ensures the data integrity. The controller transmitting the data performs a CRC, and all the receiving modules perform a CRC, and all CRCs should match; if they do not, an error is created. If any device on the network sees a mismatch, it takes that single bit at the end of the frame and makes it a dominant bit. Now that an error flag has been detected all modules take that frame and disregard it, so no modules on the network use that frame. The CRC makes error detection very reliable. If we get clock drift in a module, that module will not pass the CRC check and insert an error flag, so the message, even though it was good, must be destroyed by the sending module.

Clock drift in a module causes significant issues because no other module detects this so that no code will be set for the module with clock drift. The module can keep setting error flags, and the messages will continue to be destroyed until 16 messages have been destroyed. Then, the sending module will remove itself from the bus and go into a passive state. The sending module is not the one with the problem. The problem module is the one with the clock drift. Modules on the bus may flag a code against the sending module. A technician may replace the sending module, but it is not the problem. A technician can reset the modules by unhooking and touching the battery cables together, and the missing module will return to the bus.

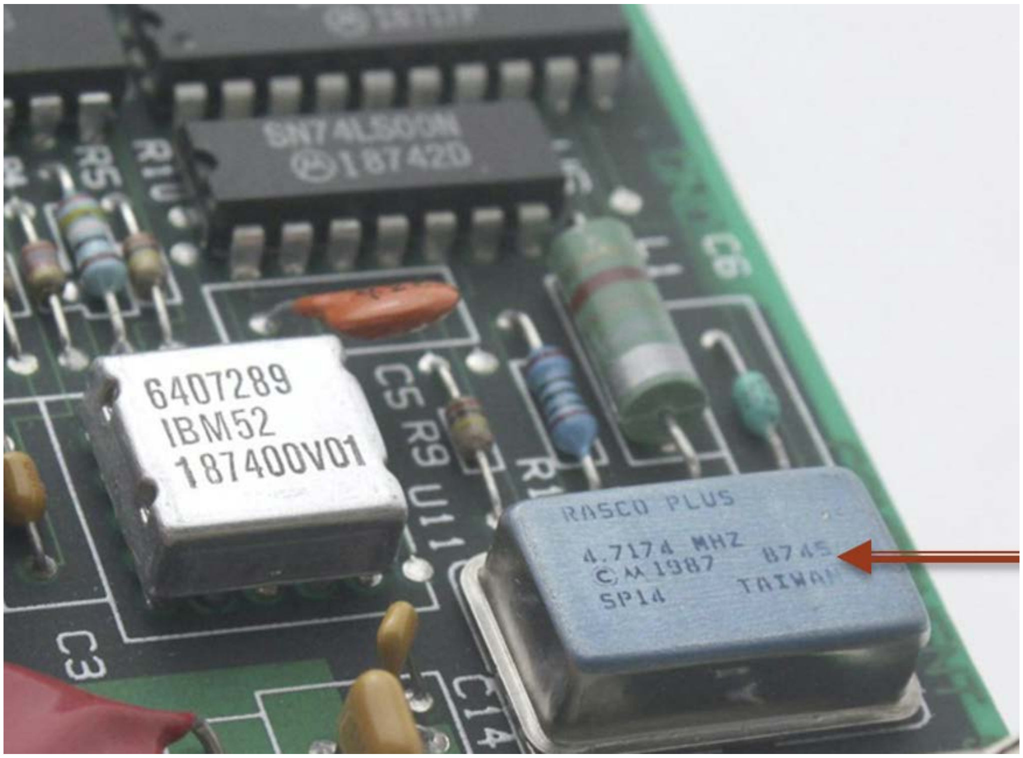
Acknowledgment (ACK): Only the CAN modules receiving the message return an acknowledgment. The sending CAN module checks the bus for an acknowledgment. If one is absent, the sending module creates an error flag, and the message is sent again. The sending module will continue to create errors and resend the message if no acknowledgment is received.

Eventually, the module reaches an error limit and places itself in a bus-off state. If you look at the message with a scope, the data packets will all look the same repeatedly, with no acknowledgment bit at the end. If no module ever sends an acknowledgment bit, it usually means the bus wires are broken going to that module so that no one can hear the message. CAN A & B do not use the acknowledgment bit. Only high-speed CAN C uses the acknowledgment.

The End of Frame (EOF) Indicates the end of the CAN frame. If the CRC strings do not match between all the modules, the end-of-frame message will break the five-like bit rule. If the message has five like bits in a row, the message is good; if the message has six or seven like bits in a row, the message is no good. When the message is poorly tagged, the sending module must destroy the message. The module that sent the message gets the bus again, and there is no arbitration on the bus for the following message. If the module cannot put together a proper message and keeps outputting wrong messages, that module can tie up the bus for a time. After 16 wrong messages, another module will insert a bit at the end of the frame, and the sending module must take itself off the bus. The offending module can still listen but not broadcast. After a specific number of messages have been transmitted on the bus, the offending module can try to broadcast a message again.

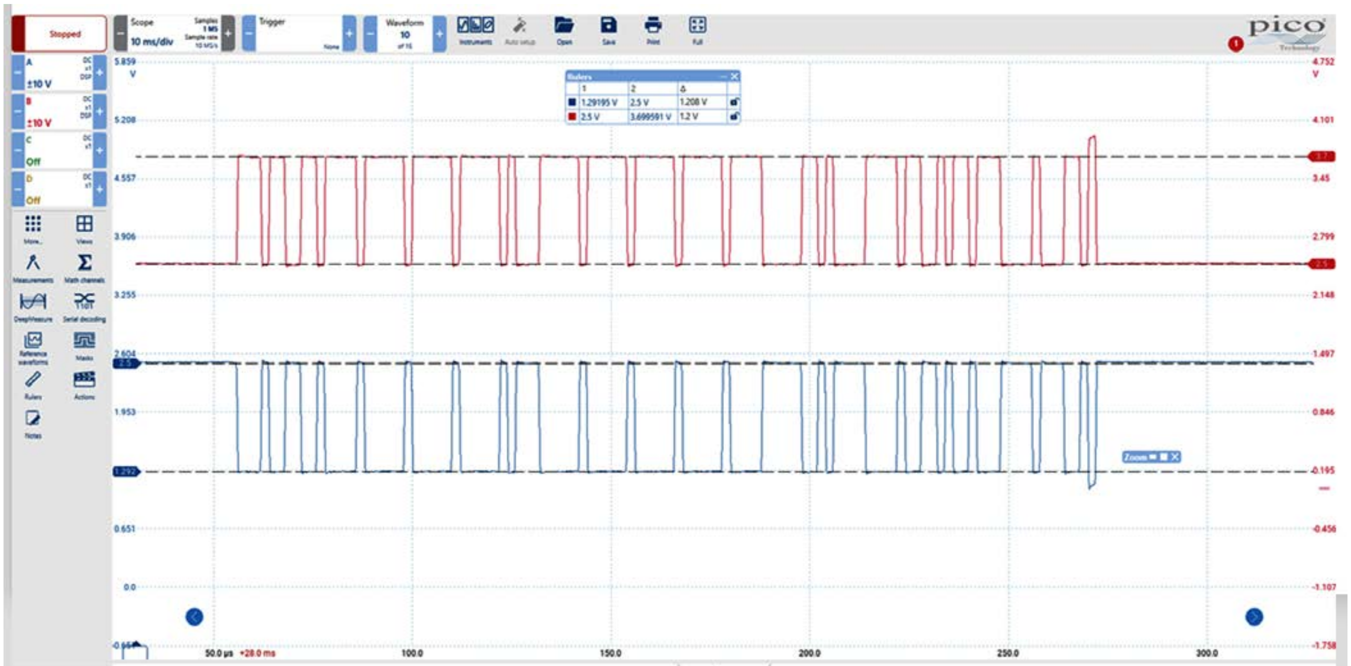
Message Access to Bus

Messages need access to the bus. In some communication systems like the Chrysler PCI bus, the module with the least resistance had the most authority to gain bus access. In some systems, a master clock was used to synchronize all the modules with the master clock to keep the messages on time. CAN does not use either of these methods. The messages are event-driven, and the message with the most authority gains access to the bus. CAN is a multi-master network, and any module on the network can be the master. The high-priority messages are broadcast first, and then lower-priority messages can access the bus. The scan tool has the least priority on the bus.



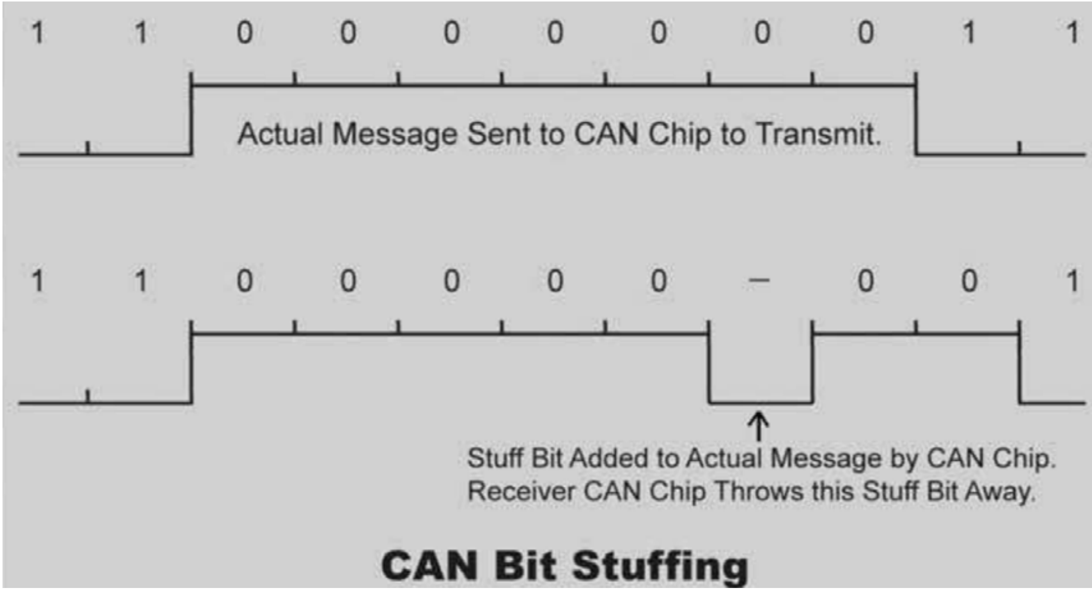
Asynchronous Method
CAN uses an Asynchronous clock method. Each module has a quartz crystal oscillator inside to keep the time of the messages. There is no master clock, and each module has its own master. The modules are all synchronized by the edges of the CAN waveform. That is why it is essential to have a clear signal without much noise. A quartz crystal oscillator is a device that has been used since the first computers. The quartz crystal contains piezoelectric properties, changing its electrical properties as it vibrates.

A quartz crystal can be cut to vibrate at a given frequency. The quartz crystal clock shown in the computer above is cut to vibrate at 4.7174 megahertz. If the clock starts to fail in the module, it can disrupt the entire message structure and cause errors and codes. Synchronous clock systems would use a master clock, and all modules would synchronize. Flexray is an example of this setup.



Non-Return to Zero

Non-return to zero means the CAN signals can continue creating a one or a zero without changing voltage during the message when there is no voltage difference between the two CAN lines. Ones are created in a string when there is a two-volt difference between the two CAN lines, and zeros are created in a string when there is no difference in voltage between the two CAN lines.



Bit Stuffing

Due to all modules having clocks and the clocks being synchronized by edges of the CAN waveform, only five like bits can be sent at a time until we resynchronize all the clocks. If we go more than five like bits at a time, we will have clock discrepancies across the board. In the example above, five 0s are being sent; after the fifth bit, the sending module performs a bit stuffing action, which means the bus changes states. The change of state on the bus creates another edge for all the modules to synchronize. After the bit stuffing action, the module can finish sending the 0's. The receiving modules disregard the bit stuff as part of the message, take it out of the message, and reconstruct the message without the bit stuff. A module with a clock discrepancy will destroy the message and disrupt the bus traffic. The technician is tasked with finding the module with the clock discrepancy.

	SOF	11 bits Identifier											...
Node A	0	0	0	1	0	0	0	1	1	0	0	1	...
Node B	0	0	0	1	0	1	Stop transmitting						...
CAN bus	0	0	0	1	0	0	0	1	1	0	0	1	...

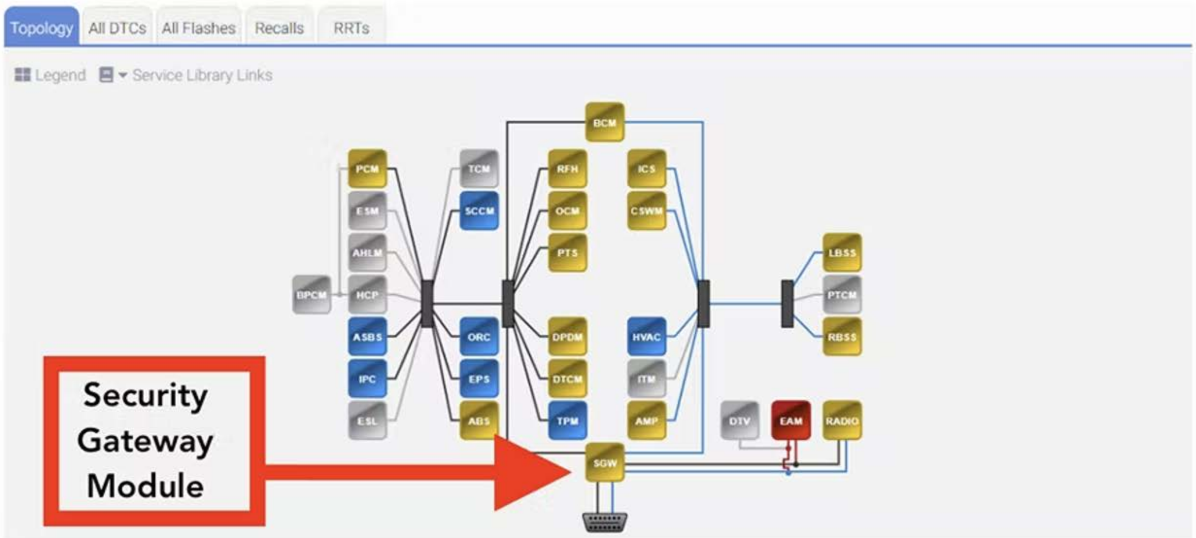
Node B loses arbitration

BUS Access CSMA/CD
Only one module can communicate at a time on the bus, which means we have to set rules for bus access. CAN networks use Carrier Sense Multiple Access Collision Detection (CSMA/CD) to govern bus access. Carrier sense means all modules sense the bus. If no module is talking, any module can access the bus at that time. If two modules start talking simultaneously, collision detection uses bitwise arbitration to determine which modules get the bus. Each bit in the ID field is analyzed for a one or a zero.

Zero is the highest priority when two modules are broadcasting simultaneously; whichever module sends out a one in the ID field first loses access to the bus; that module recedes and now starts listening. The module that sent out the last zero takes control of the bus and continues to send out its message. The first three bits in the ID field usually determine who gets bus access, but not always. The message would be highly prioritized if the first three bits were zero.

CSMA/CD
In the example above, Node A and Node B send out the same binary numbers until frame five of the identifier. In frame five, Node B sends out a recessive bit, a one, and Node A sends out a zero, meaning Node B stops transmitting. Node A wins the arbitration, taking control of the bus, and continues sending its message.

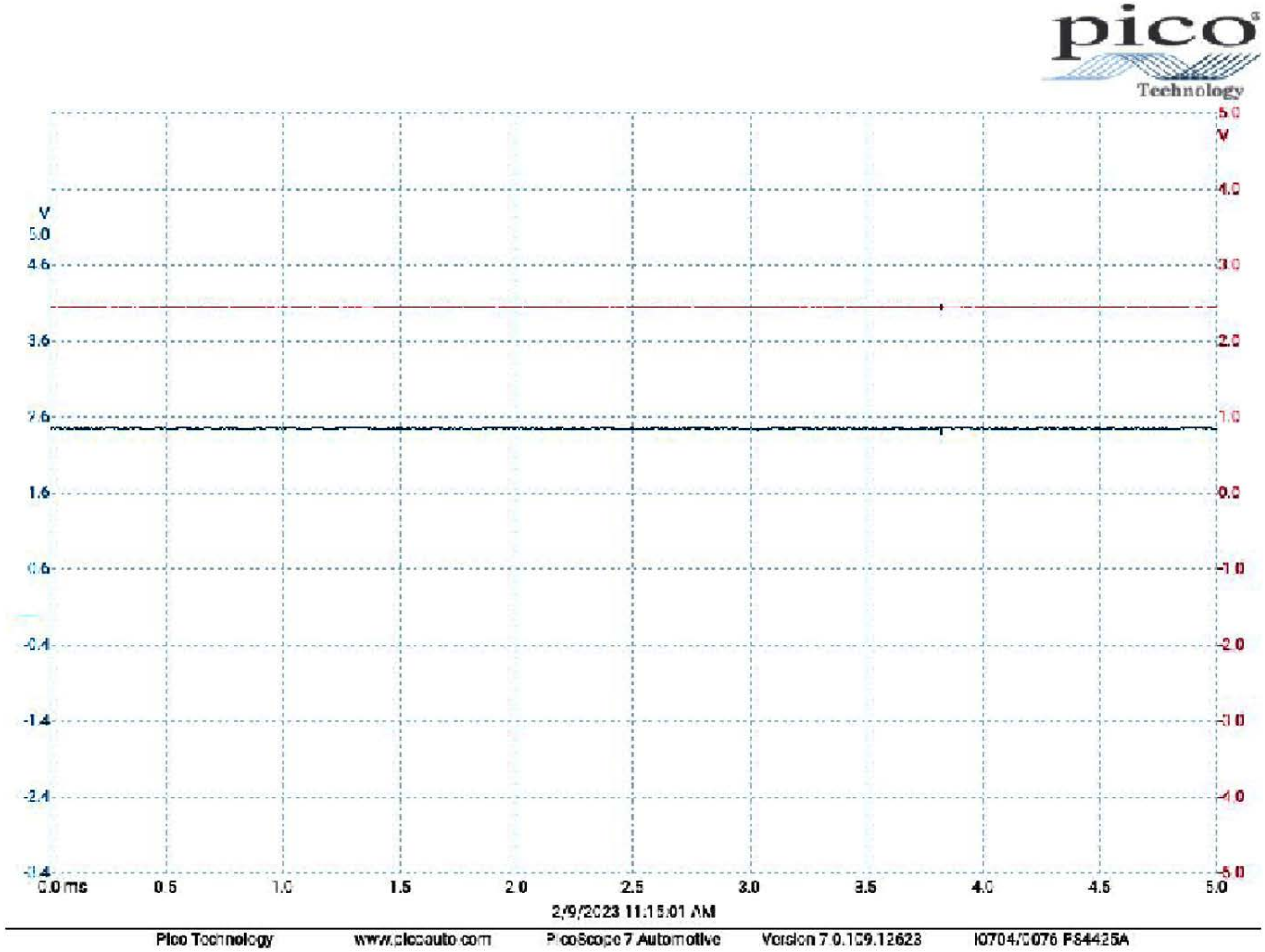
Register with Auto-Auth



Security gateway between scan tool and network

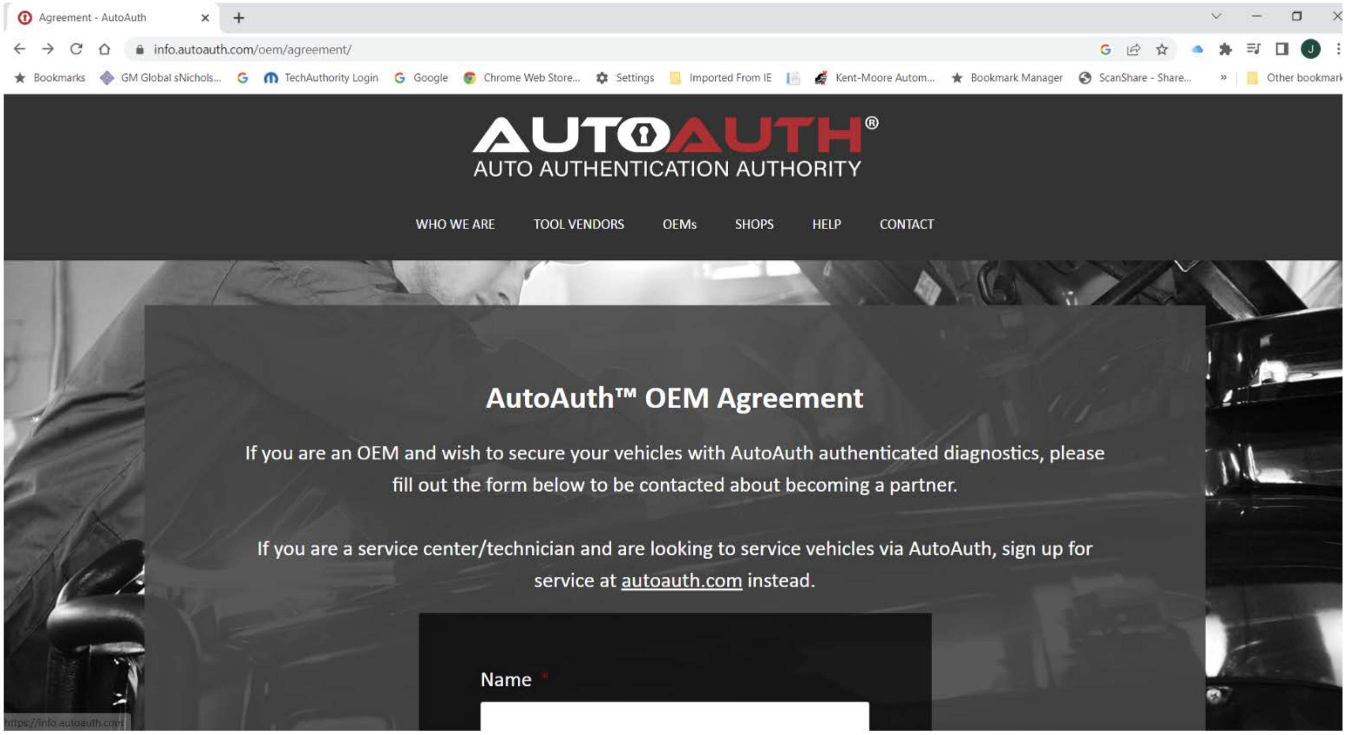
Secure Gateway Chrysler
The secure gateway was added to some Chrysler vehicles in 2018 and all 2019 models. This gateway isolates the scan tool from the network and increases network security. Also, the secure gateway acts as a firewall to reduce the possibility of hackers getting in and exploiting the system. The gateway is split into two sectors: a private sector and a public sector. The public sector includes the telematics unit and the DLC.

Everything else is considered private. The private network has limited access without the proper authorization. In the early days, no scan tool other than an authenticated scan tool could access all the controls, like erasing codes and bidirectional commands, as well as programming. Some tuner companies that needed to install their programs developed a bypass allowing the tuner to program the PCM. A website called Auto-Auth can now be used to gain full access to the system. They allow access once you have signed up for their subscription service. This grants access to the gateway to many different aftermarket scan tool brands.



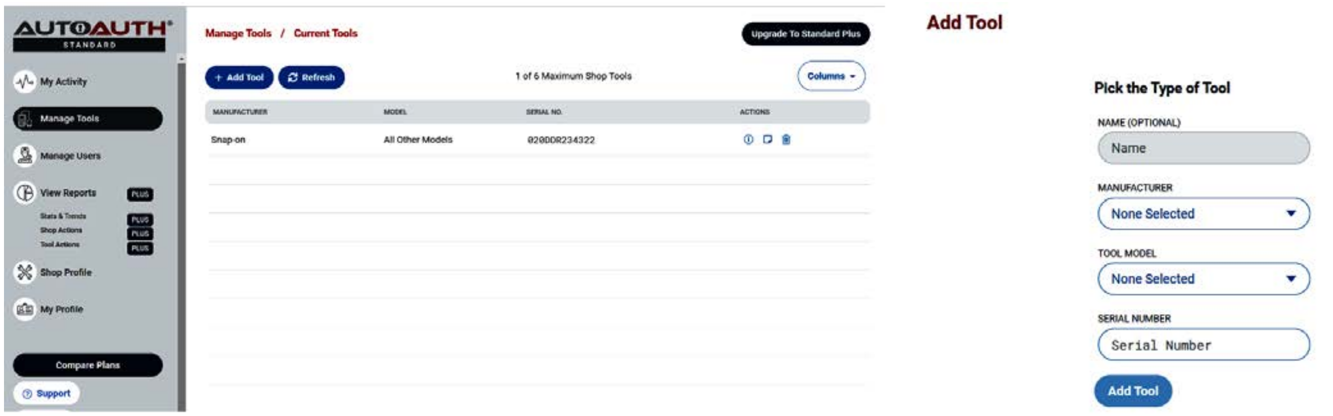
Secure Gateway

The security gateway sits between the scan tool and all the networks. Unless the scan tool is connected and talking, there is no activity on pins 6 and 14 or 3 and 11. In our example above, we are testing pins 6 and 14 without bus activity. If you want to analyze the CAN bus, you must back probe the bus circuit at another location because it is unavailable at the DLC.



Secure Gateway Access

Starting in 2018 and up, many Chrysler vehicles use a secure gateway module, preventing scan tool access to all functions. Scan tools can gain access by performing three simple steps. First, go to the AutoAuthweb site(info.autoauth.com) and sign up. The second step is to confirm a compatible scan tool with approved credentials stored in the scan tool. The scan tool serial number must be entered. The third step is to authorize secure gateway access immediately. The cost is $50.00 per year currently and allows for up to 5 tools to be registered on one account. The scan tool must be connected to the internet during sign-up and activation.



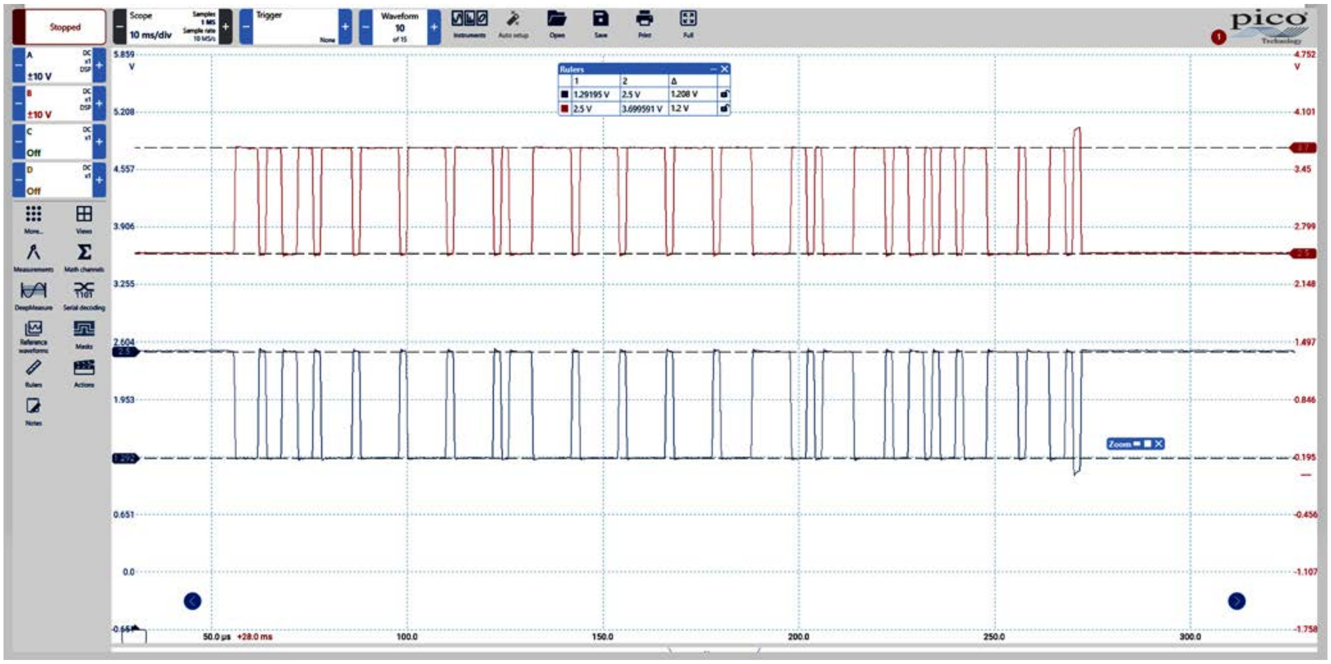
Confirm Platform
Select shop on the menu and add your tool. It will require tool make and serial number to be entered. You can add up to five tools per shop and five users per shop and one admin per shop. There is an additional two-dollar charge for each employee you add.

Bypassing Secure Gateway
In order to install aftermarket programs, the secure gateway must be bypassed. One method is to install the OBD Genie, bypassing the secure gateway and allowing programs to be installed via the OBD 16-pin connector. Installation requires locating the secure gateway module, removing two plugs at the secure gateway module, plugging the Genie into the secure gateway module, and then plugging the two connectors into the Genie.

The Genie performs the bypass function, so the 16-pin OBD 2 connector can install aftermarket programs and access scan data. Several other companies offer a bypass, but we just listed the Genie as an example.
https://www.obdgenie.com/pages/obd-genie-chrysler-dodge-jeep-ram-security-gateway-module-bypass-c-modbyp

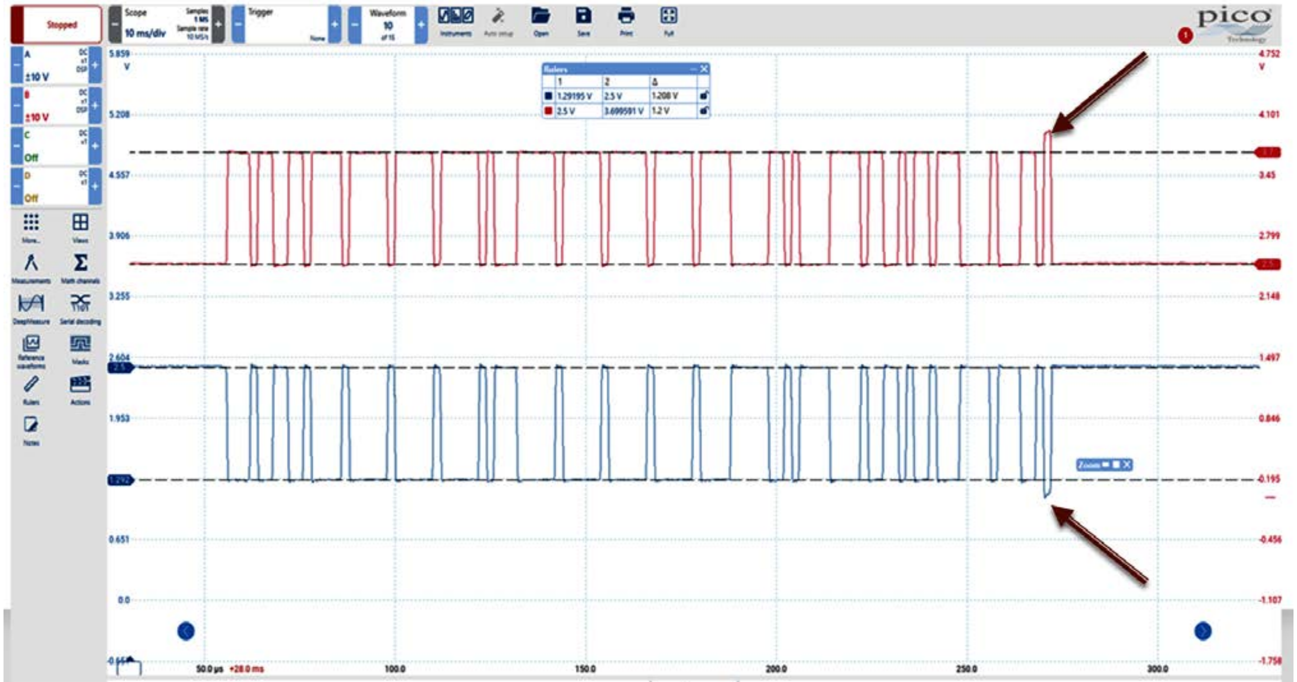
Bypassing Secure Gateway
Another method of bypassing the secure gateway module is to find a place to access the CAN lines other than the OBD port. There are several cables available on the market. Some cables simply plug into an existing module like the instrument cluster and gain access to the CAN bus. The cable shown is designed to piggyback the CAN H and CAN L lines at any connector. The red wire on the cable connects to CAN H and the black wire connects to CAN L. The cable shown plugs into the OBD port to obtain power and then the other end plugs into your scan tool.

Cable above-Smart box SB-SBP006 bypass cable



Normal Waveform

Ford Powerstroke example. CAN H starts at 2.5 volts and rises to 3.7 volts. CAN L starts at 2.5 volts and decreases to 1.2 volts. A low pass filter was applied to the lab scope. The filter has been set at 700kHz. The end of the frame is a slightly higher voltage, showing acknowledgment of what has happened.



Acknowledgement Bit

The acknowledgment bit at the end of the frame has a slightly higher voltage than the rest of the CAN bus signal. The arbitration field and the acknowledge bit are the only times multiple modules can speak at the same time during the message. The acknowledged bit voltage will change depending on how many modules acknowledge the message. The more modules acknowledge the message, the higher this voltage goes.

Assessment

Questions

Question: T or F CAN bus is limited to 75 modules.

Question: T or F Can bus resistance equals 120 ohms.

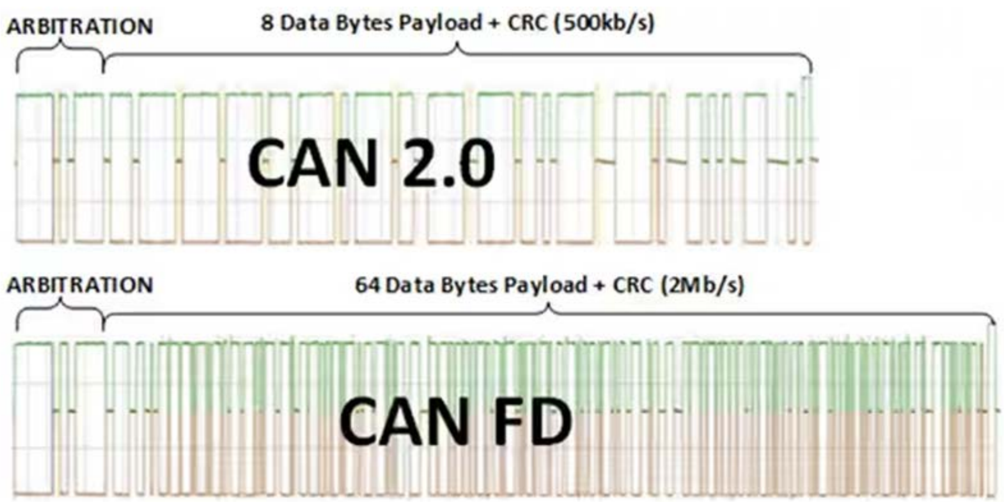
Question: T or F Modules on the CAN bus arbitrate for bus access.

Questions with Answers
Question: T or F CAN bus is limited to 75 modules.
Answer: False, CAN bus is limited to 30 modules.

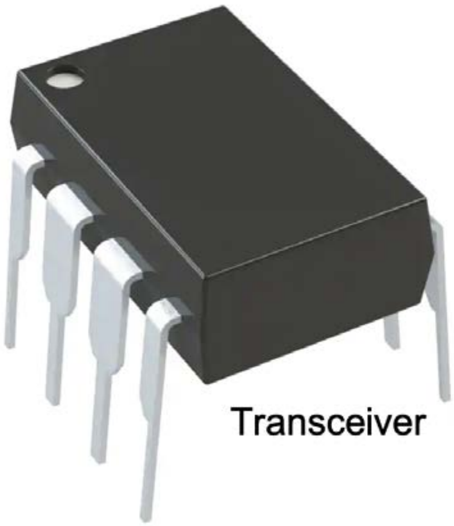
Question: T or F Can bus resistance equals 120 ohms.
Answer: False, CAN bus resistance equals 60 ohms.

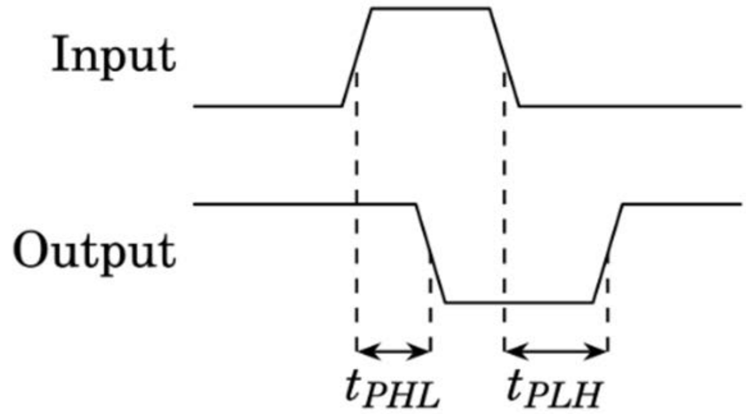
Question: T or F Modules on the CAN bus arbitrate for bus access.
Answer: True, Modules must arbitrate to gain access to the bus.

Chapter 3 - Network Definition and Type



CAN FD
CAN FD is the next-generation CAN bus. It was introduced in 2012 and standardized in 2015 (ISO11898-1). Data rates are available from 2Mbps TO 8Mbps compared to CAN at max 1Mbps. CAN FD allows a much larger payload than traditional CAN. Traditional CAN payload was 8 bytes, but CAN FD handles payloads up to 64 bytes. CAN FD also improved security. Vehicles are more complex, and data transfer requirements have increased immensely. Manufacturers have invested heavily in CAN and do not want to change network topology due to the massive expense. CAN FD allows the manufacturer to update CAN and not change network topology. Redesigning the transceiver was the most significant change that had to be made to migrate into CAN FD.





Propagation delay timing diagram of a NOT gate

CAN FD Transceivers

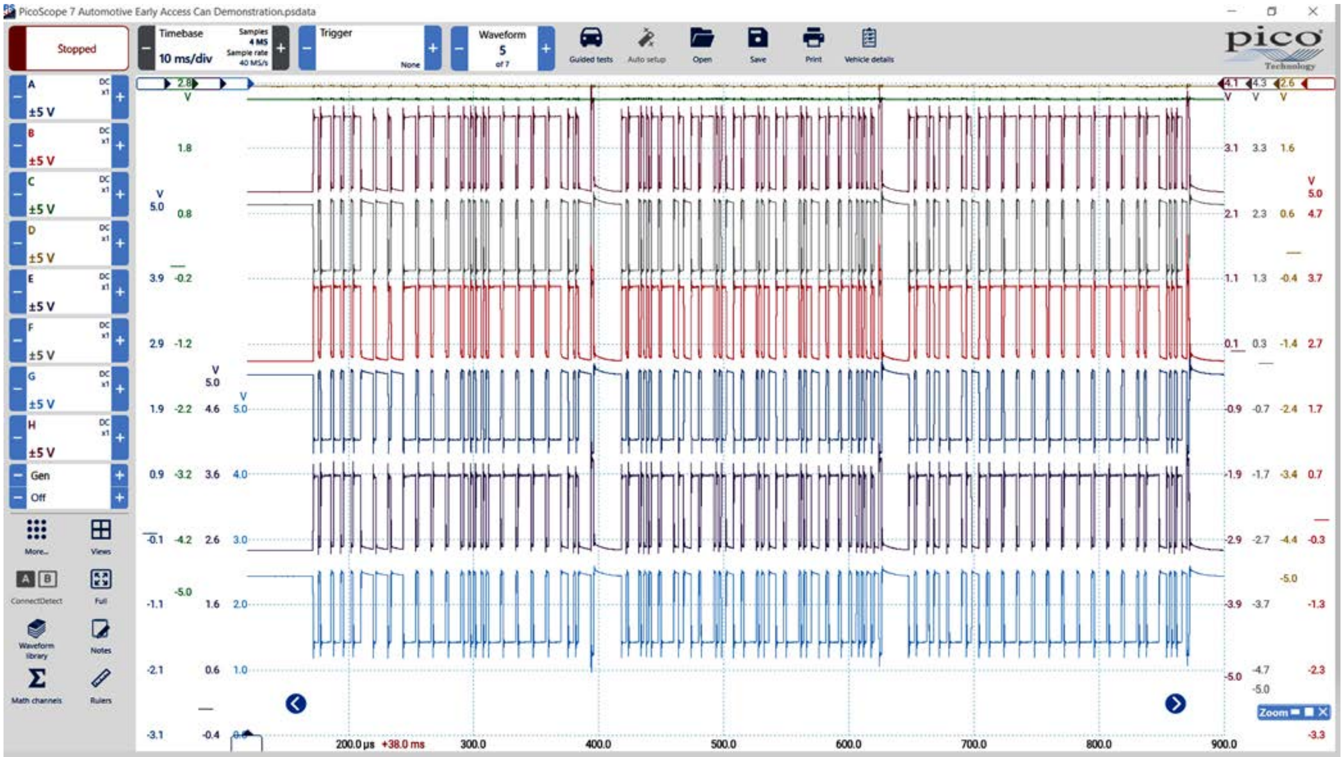
CAN FD networks use transceivers to transmit and receive messages, just like traditional CAN networks. FD transceivers improve latency over traditional CAN. The propagation delay of FD transceivers is in the 120-nanosecond range.

Message Propagation Delay

Propagation delay is the time required for a signal to be received after it has been sent; it is caused by the time it takes to travel through a medium. For example, a lightning bolt can be seen before the thunder is heard. The light from lightning travels at 186,000 miles a second, while sound travels one mile in 4.7 seconds. The propagation delay depends on the size and length of the medium and the logic gate family used. The logic gate family is typically transistor-transistor logic or complementary metal-oxide-semiconductor. Inconsistent propagation delay results in data errors. Transceiver specs usually list the typical and maximum propagation delays, as shown above. Propagation delay can be checked to some degree by checking the CAN signal in multiple locations using additional channels on the lab scope.

In computer networks, propagation delay is the amount of time it takes for the head of the signal to travel from the sender to the receiver. It can be computed as the ratio between the link length and the propagation speed over the specific medium.

Propagation delay is equal to d / s, where d is the distance, and s is the wave propagation speed. In wireless communication, s=c, i.e.the speed of light. In copper wire, the speed generally ranges from .59c to .77c. This delay is the major obstacle in developing high-speed computers and is called the interconnect bottleneck in IC systems.



Propagation Delay Testing

A lab scope can be used to check for signal propagation delay. Test the CAN signal at multiple locations and see if there is a time difference between any locations. We use six channels on the lab scope above to check the CAN signal at three modules. There is no delay in any of the signals.

CAN FD Adapter
Accessing the CAN FD network may require a special adapter. An example of where you may see CAN FD is 2020 and up GM models and 2018 and up Ford models. Your scan tool may exhibit the message unable to communicate with the control unit if the adapter is not used. Most scan tool companies have updated their scan tools to function with CAN FD.

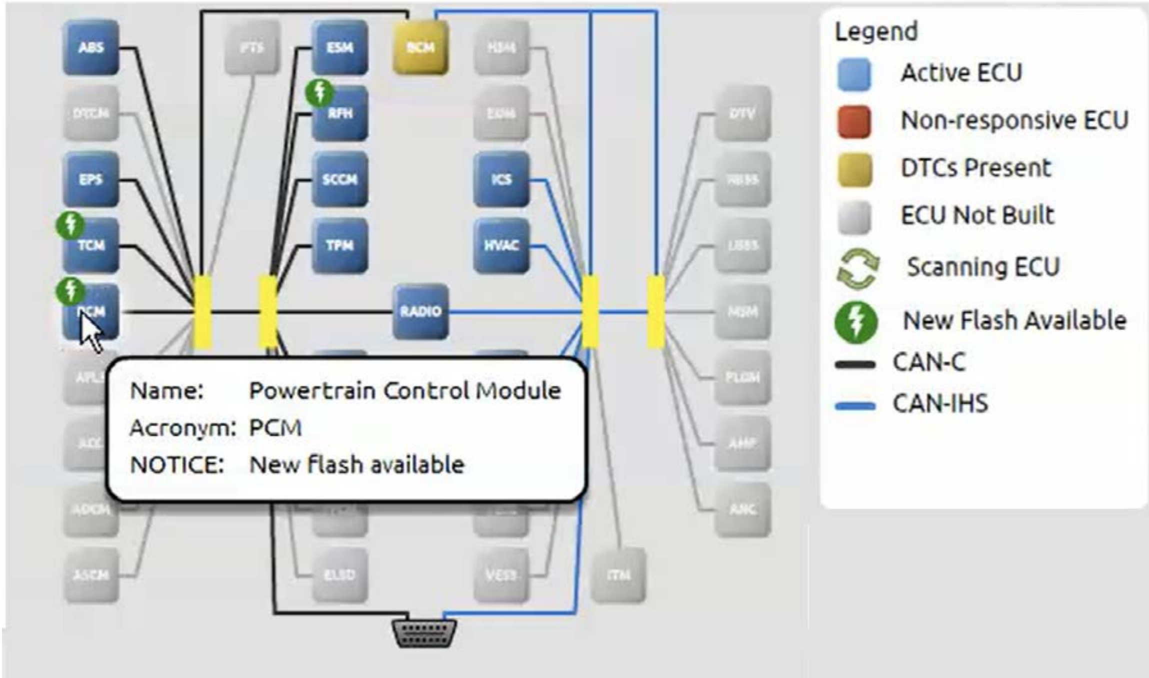
https://www.kvaser.com/e-learning/

Chapter 4 - CAN Data Diagnostics

Chapter CAN Data Diagnostics
Your meter is the first line of defense. Powers, grounds, and data lines need to be confirmed and LOAD tested before condemning a module.

Communication Failures
Communication failures come in three levels. Physical layer failure includes wiring and terminals. The transport layer is a failure in the packets of data. Software layer failure is in the logic and programming.

Scan Tool Issues
All data must be decoded to be read correctly. The scan tool becomes a node or module on the network. The scan tool has the least authority, so it must wait for a break in action before making a request. The scan tool may be unable to decode the data, and it may say there is no communication. The scan tool also may misinterpret the data and give false data. The scanner may give you false codes. If you experience any of these concerns, try another scan tool.



Module Status
A factory-level or equivalent scan tool is a great place to start to check a module's communication status. The example above is from a 2015 Chrysler. The legend indicates that a non-responsive module would be coded red. This scan tool also indicates if the modules retain any DTCs and if the module requires a reflash. The lines from each module are color-coded to alert the technician to what type of communication is being used. A black line indicates CAN C and a blue line represents CAN-IHS. Some scan tools, such as Toyota TIS, will run a CAN bus test.

Decoding CAN Data
Messages may be corrupt and unreadable by other modules, including the scan tool. A technician needs a way to view the messages to determine if they are corrupt.
CAN data can be extracted using a couple of methods. One method is connecting a scope and looking at the CAN bus waveform. We can check the waveform integrity; some scopes have a built-in serial decoder, which we will explain later. Another method of analyzing CAN data is connecting a CAN bus decoder tool. A CAN bus decoder tool can extract each frame of CAN data and convert it from binary to HEX but not into actual values like engine temperature, speed, and other parameters.

To extract actual values like the ones seen on a scan tool, a database container file (DBC)is needed. A DBC file contains essential text files, including information for decoding raw CAN bus data to physical values or human-readable form. The DBC file contains proprietary information that is not readily available. Some car hackers take the CAN data, reverse engineer the rules, and make a DBC file. There are a few sites that have some of these files that you can gain access to. One site is CCS electronics, including the CAN sniffer decoding tool and some DBC files (https://www.csselectronics.com/pages/can-bus-sniffer-reverse-engineering).

Kvaser leaf light tool

Reading CAN Data

Reading the raw data in each CAN frame can be done with a CAN bus decoder tool. The tool plugs into the OBD connector and a USB port on your computer. A program to run the tool must be downloaded to your computer. Plug the tool into the OBD connector and, using the onboard program, record the raw CAN data. The raw data will be displayed in a Hex format. If you cannot access a DBC file to decode the data, you can only view the Hex message. The main concern to the technician is not the physical data, but if the frame had an error, the message was corrupt.

Tool shown-Kvaserleaf light HS V2 OBD2
USB to CAN interface
Kvaser'sLeaf Light V2 series is the workhorse of the CAN interface world. The newly launched KvaserLeaf Light V2 supports high-speed USB for CAN and offers high performance at a low cost. Loss-free transmission and reception of standard and extended CAN messages on the CAN bus is transmitted with a time stamp precision of 100 microseconds.

The KvaserLeaf Light V2 is a high-speed USB interface for CAN that offers loss-free transmission and reception of standard and extended CAN messages on the CAN bus.
8000 messages per second, each time-stamped with 100 microsecond accuracy
Galvanic isolation, previously a more expensive option on Kvaser's original Leaf Light, now comes as standard on the Leaf Light v2, enhancing protection from power surges or electrical shocks.
Low current consumption of 70 mA, reducing power drain from your laptop.

Local buffering and pre-processing result in high performance and reduced time-critical tasks for the PC.

Leaf Light v2 comes with free software, free software updates, and free support. Kvaser's easy-to-use KvaserCANlibapplications programming interface (API) has virtual hardware that simulates our real hardware, so you can start developing around the Leaf Light v2 interface immediately.

To understand what 'raw CAN data' looks like, see the below example CAN frame from a truck:

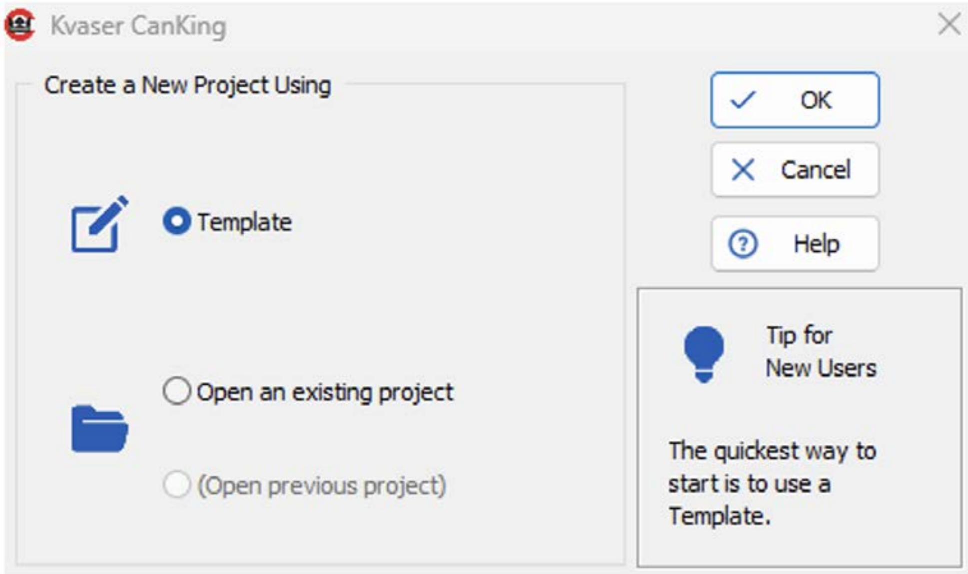
```
CAN ID    Data bytes
0CF00400  FF FF FF 68 13 FF FF FF
```

If you have a CAN DBC that contains decoding rules for the CAN ID, you can 'extract' parameters (signals) from the data bytes. One such signal could be EngineSpeed:

```
Message  Signal       Value  Unit
EEC1     EngineSpeed  621    rpm
```

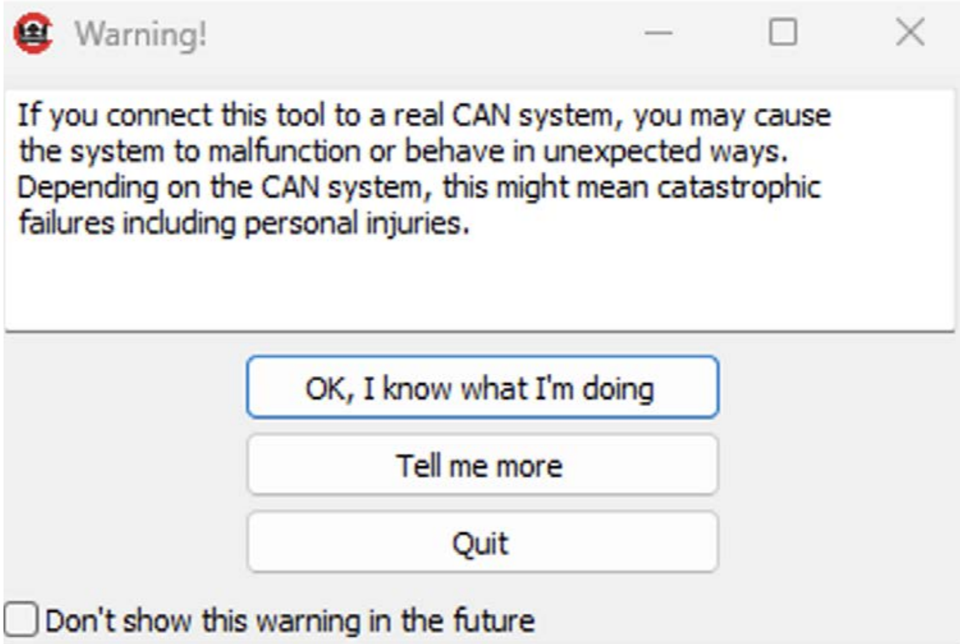
Converting Data

The CAN frame information is in binary code. The decoding tool with the software will convert the binary into Hex. In the example above, we extracted a CAN frame from the J1939 CAN bus and converted it into Hex. The DBC file was available for this application, so the decoding software converted the Hex into physical values. The physical value displayed in this example is engine speed.

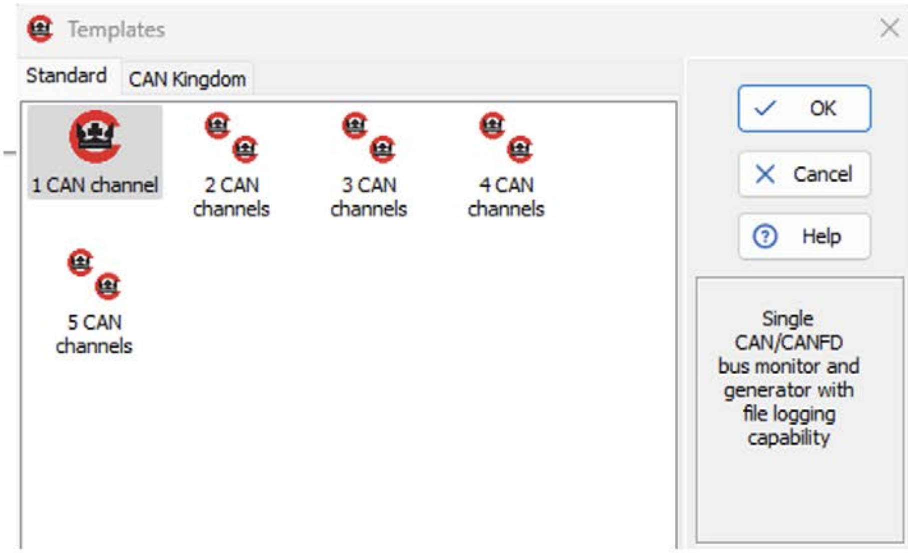


CanKing Example
Example: CanKingis a software program that is downloaded to your computer. The software will interface with a CAN logger (Decoder tool). Plug the tool into the OBD 2 port and turn on the software program. The software will ask if you want to open an existing project or start a new template. If you are connected to the car select template.
https://www.kvaser.com/software_application/bus-analysis/#?applications=bus-analysis

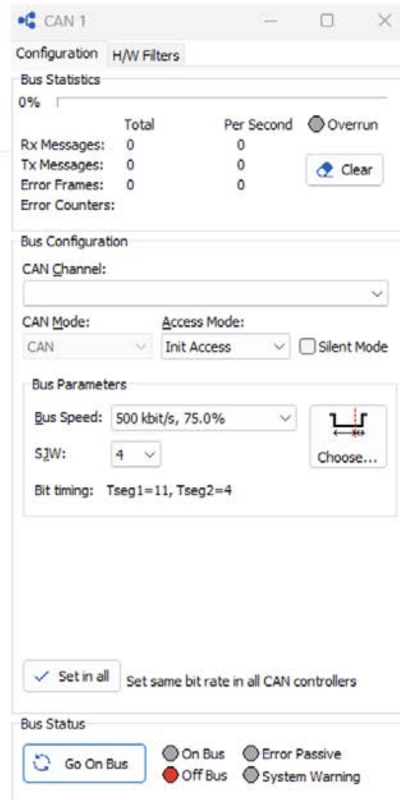
- Send & Receive CAN and Extended CAN messages
- Supports CAN FD, both ISO and non-ISO
- Log to File (Up to 4 Channels)
- Filter Messages
- Generate Error Frames
- Timed Transmission
- History Transmit List
- Traffic Generator
- Virtual CAN Channels
- Choose your own Bus Parameters
- CANkingdomMessage Formatters
- Supports Custom Formatters



Connect To OBD Port
Connect the tool to the OBD port to start logging CAN data. A message warning will appear on your screen. The warning says you can cause the system to malfunction, which could result in catastrophic damage or even personal injury. Data logging does not usually cause problems, but uploading new messages to the CAN bus could cause severe problems if not performed correctly.



SELECT CAN CHANNEL



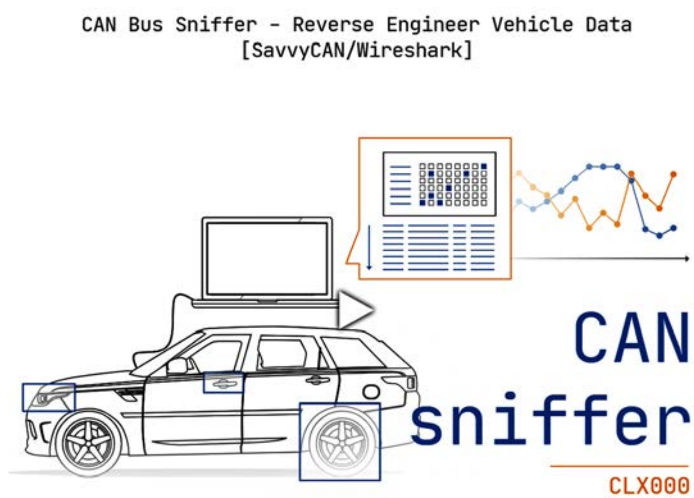
SELECT BUS SPEED

Selections

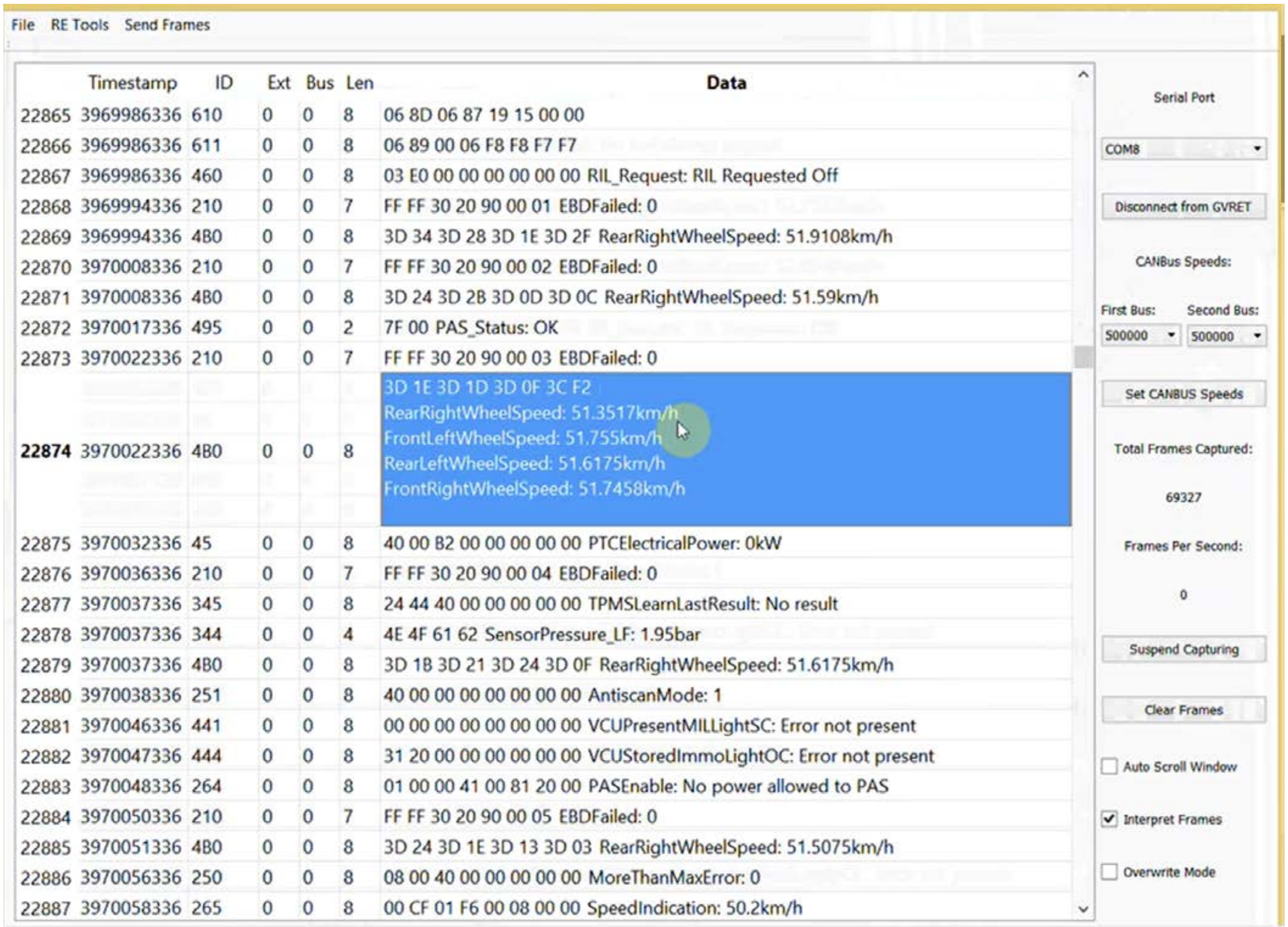
The technician must set up the software before viewing CAN data messages. In our example, we picked one CAN channel. You must select the bus speed, which we have chosen to be 500 kbps. After making your selections, hit the button at the bottom left to board the bus.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Typ	Ch	Pri	PGN	SA/I	DLC	D1	D2	D3	D4	D5	D6	D7	D8	Time	Dir
1695	CAN	1	3	0F004	00->*	8	43	87	8D	66	14	0	F0	87	222.036	R
1699	CAN	1	3	0F004	00->*	8	43	87	8D	66	14	0	F0	87	222.045	R
1702	CAN	1	3	0F004	00->*	8	33	89	8F	66	14	0	F0	89	222.0541	R
1704	CAN	1	3	0F004	00->*	8	33	89	8F	94	14	0	F0	89	222.065	R
1706	CAN	1	3	0F004	00->*	8	73	8B	90	94	14	0	F0	8B	222.075	R
1710	CAN	1	3	0F004	00->*	8	73	8B	90	94	14	0	F0	8B	222.085	R
1714	CAN	1	3	0F004	00->*	8	23	8E	93	D6	14	0	F0	8E	222.094	R
1717	CAN	1	3	0F004	00->*	8	23	8E	93	D6	14	0	F0	8E	222.105	R
1721	CAN	1	3	0F004	00->*	8	23	90	95	D6	14	0	F0	90	222.115	R
1724	CAN	1	3	0F004	00->*	8	23	90	95	42	15	0	F0	90	222.126	R
1729	CAN	1	3	0F004	00->*	8	53	92	95	42	15	0	F0	92	222.1359	R
1731	CAN	1	3	0F004	00->*	8	53	92	95	42	15	0	F0	92	222.145	R
1734	CAN	1	3	0F004	00->*	8	13	96	98	C0	15	0	F0	96	222.154	R
1736	CAN	1	3	0F004	00->*	8	13	96	98	C0	15	0	F0	96	222.166	R

Message example
The data logger has captured some frames. The software has converted the data to a Hex number. The highlighted data is engine speed but is still in Hex format.

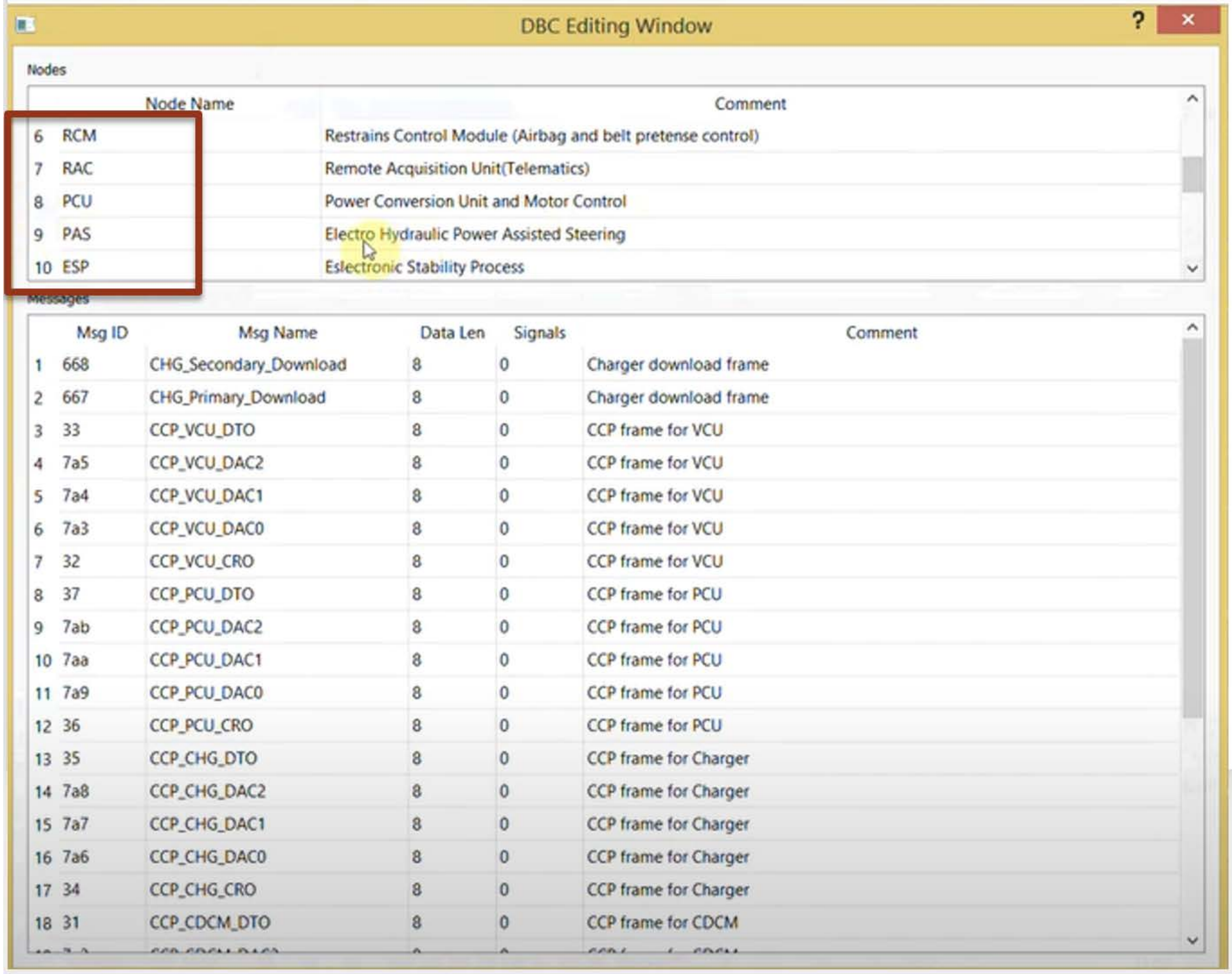


CAN Sniffer Software
You will need a DBC file to convert the data logger files you have collected into physical data. If possible, locate the DBC file that applies to your manufacturer and use the file to convert the Hex data into physical data.
https://www.csselectronics.com/pages/can-bus-sniffer-reverse-engineering



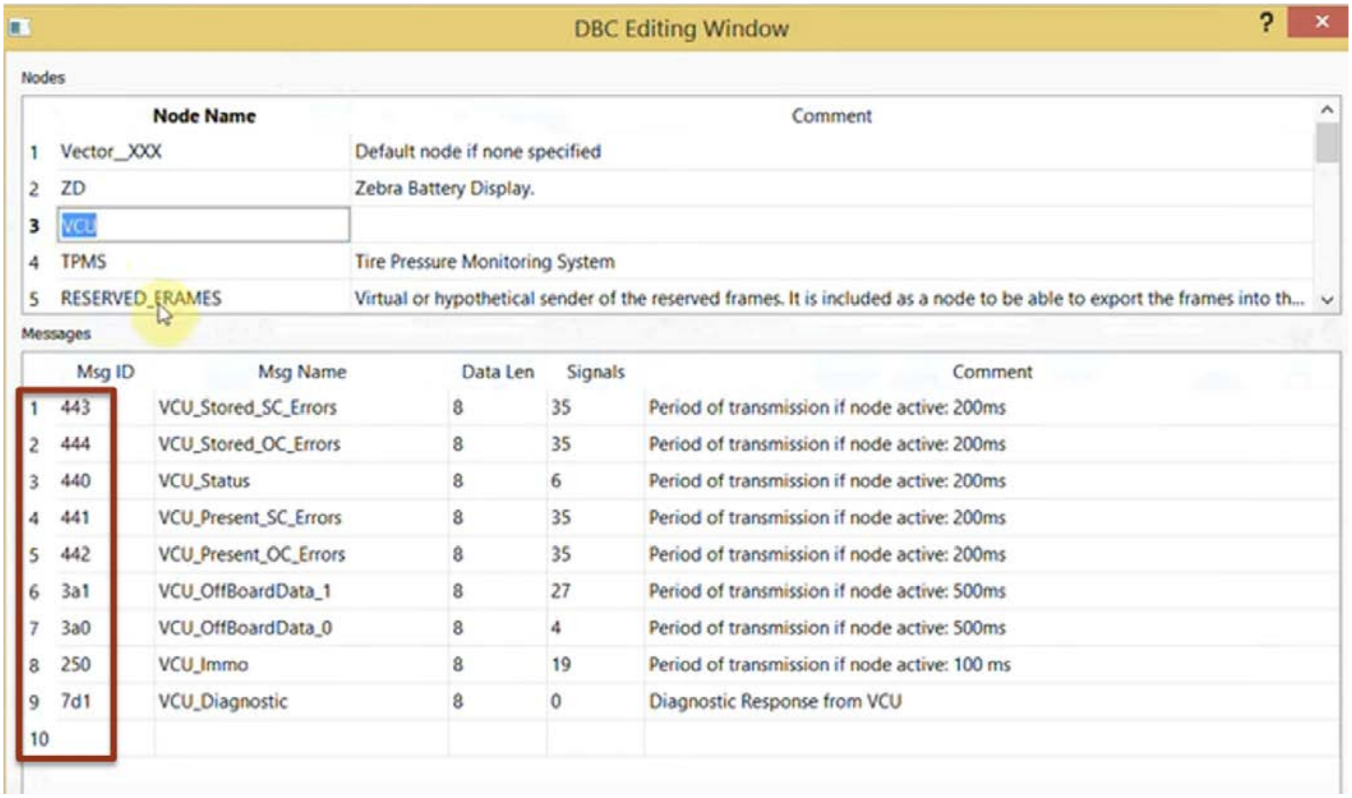
Physical Data

Once the DBC file has been downloaded to the CAN Sniffer software, you can click on a Hex message to extract the physical data. In this example, the wheel speed sensors all report wheel speed. The DBC file converted the Hex format to give you the same data you would have retrieved from a scan tool.



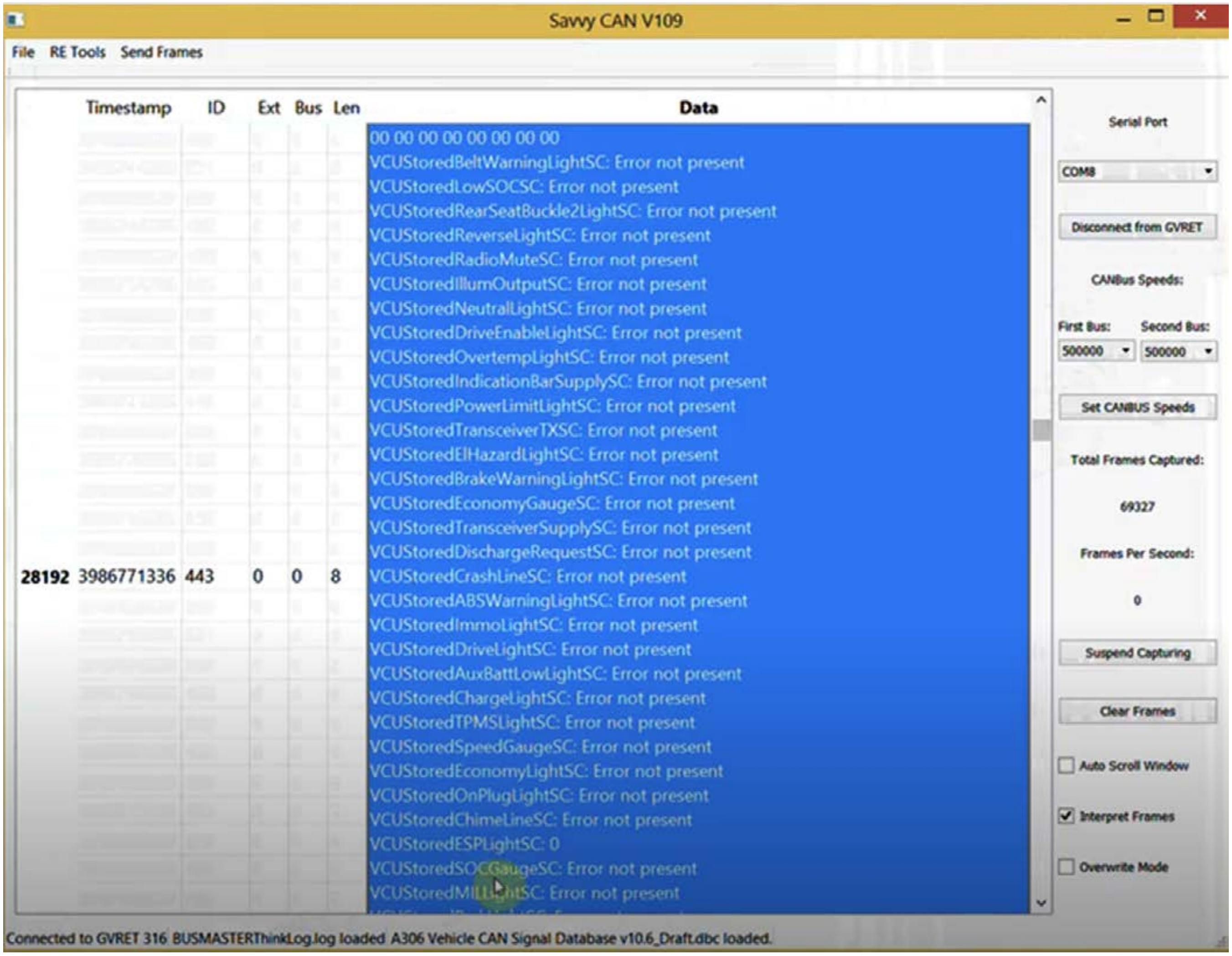
Module List

The CAN sniffer software will list all the modules communicating on the bus, which may be handy during diagnostics. The top left corner of the screen lists the modules; you can scroll down through the list. If you know the vehicle has a specific module, but it is not reporting on the bus, this could be helpful.



Controller Message ID

The CAN data message has an identifier field up front to identify the module and message. Each module uses many different IDs depending on the message it is sending. In the example above, the TPMS module sent nine messages with nine different IDs.



Check For Errors

A module sending corrupt files, basically sending a file with errors, needs to be identified. If you can identify the module corrupting the bus, you can diagnose the module and determine if replacement is necessary.

Capture data

Save to text file with notepad

Open excel and import

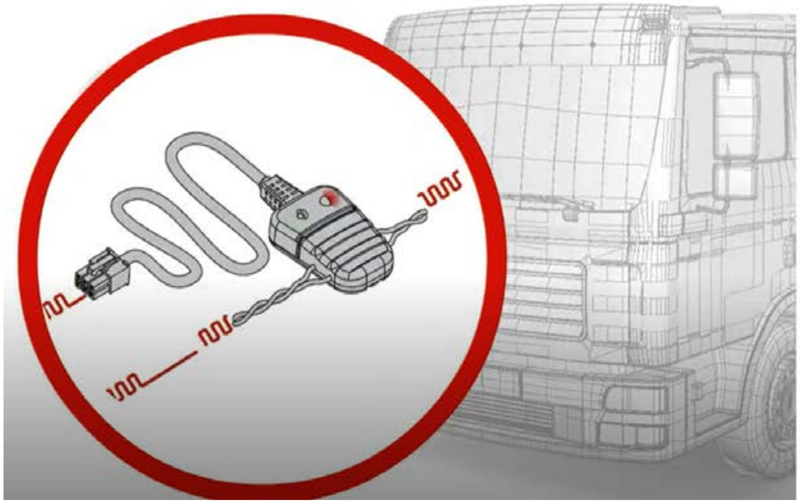
Format data in separate columns

Isolates the data to be analyzed

Analyze Data Without DBC

The steps we will follow are:

- Capture the data with CanKing, and save it to a text file with Notepad
- Open Excel and import the data from the text file
- Format the data to be in separate columns
- Filter the data to find the CAN frames of interest
- Isolate the data bytes to be analyzed
- Convert the data to a graphable format
- Create a graph showing the signal of interest



CAN Crocodile

- CAN Crocodile - Contactless CAN Bus Reader – CSS Electronics
- Open the CANCrocodile housing
- Place the CAN-H/CAN-L wires on the marked surface
- Close the housing, engaging the latch
- Optionally seal the housing (recommended)
- CANCrocodile is designed for safe data reading from vehicle CAN bus. Contactless reader is used within vehicle telematics and GPS tracking systems for receiving various parameters of vehicle operation from CANbus: speed, RPM, oil pressure & temperature, fuel volume & consumption, etc.
- Data reading occurs without electrical connection and without damaging CAN wires. CANCrocodile works in “listen” mode only, i.e. it does not change original J1939 messages and does not send any signals to CANbus.
- Safe connection to CAN bus
- CANCrocodile does no damage to insulation of wires and has no direct electrical contact with CANbus. Data reading occurs in a contactless way – CANCrocodile detects magnetic field around CAN-High and CAN-Low wires.
- Strong fixation of CAN bus wires
- The design of the body provides perpendicular fixing of the wires to the reading surface. This way of fixing will not allow the CAN bus wires to slip from the CANCrocodile reading surface.
- A soft pressure pad in the upper part of the housing tightly presses the wires to the reading surface.
- LED indication of operation modes
- Red indicator – power is on, blinking green indicator – messages received from CAN bus. Indicators allow to test the operability of the reader without dismantling and connecting to monitoring terminal.
- Open CANCrocodile housing.
- Place CAN-High and CAN-Low wires in the grooves of CANCrocodile housing according to the marks on reading surface.
- Close CANCrocodile housing in a way to engage the latch. Put sealing rope through special holes and seal housing, if necessary (recommended).
- CANcrocodile CANbus data reader for vehicle telematics

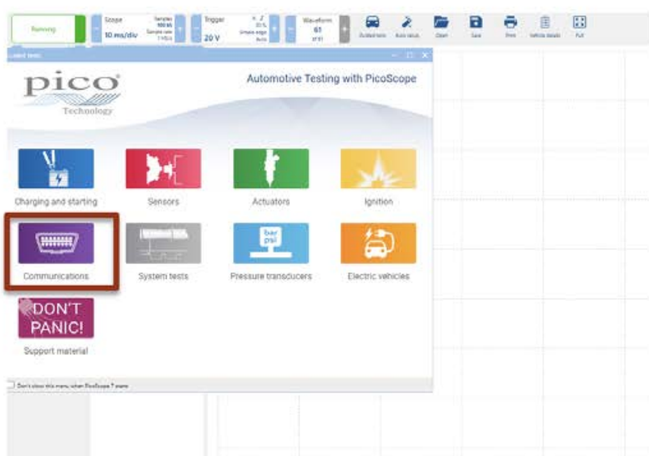
Pico Scope Decoding

Bus H must have a 2.5-volt bias level, and the voltage must travel upwards towards 3.5-3.75 volts approximately. CAN bus L must have a 2.5-volt bias and travel downward to approximately 1.2-1.5 volts. The waveform must not have much noise. These are just a few things that must be analyzed when looking at the CAN bus voltages from a lab perspective. The waveform may look okay voltage-wise, but the message may be corrupted. The message is a series of 1s and 0s created by a voltage differential of CAN H, and CAN L.

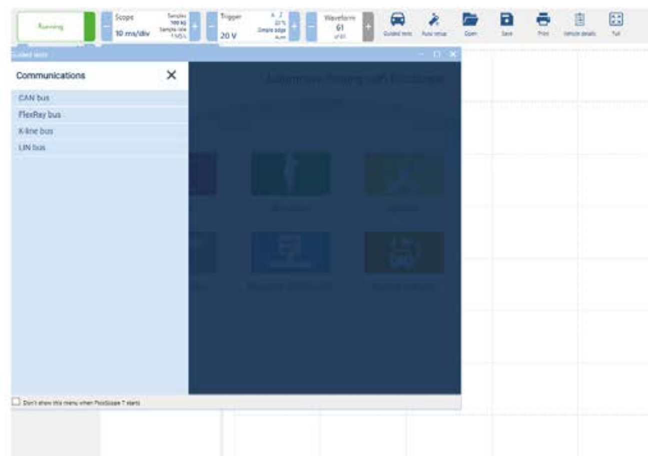
Our human minds cannot look at the waveform to decode the message and verify message integrity. There are a couple of ways of decoding the message. A CAN bus decoder, or the Pico scope, has a serial bus decoder function and can be used. The message is typically decoded into a hex format with no actual data like engine temperature or speed, as seen on a scan tool. In order to extract the actual data from the HEX format, you must apply a database file.

A database file in the module contains the information and math formulas to decode the hex message. The database file is proprietary to each manufacturer, and that information is not readily available.

GUIDED TEST MENU

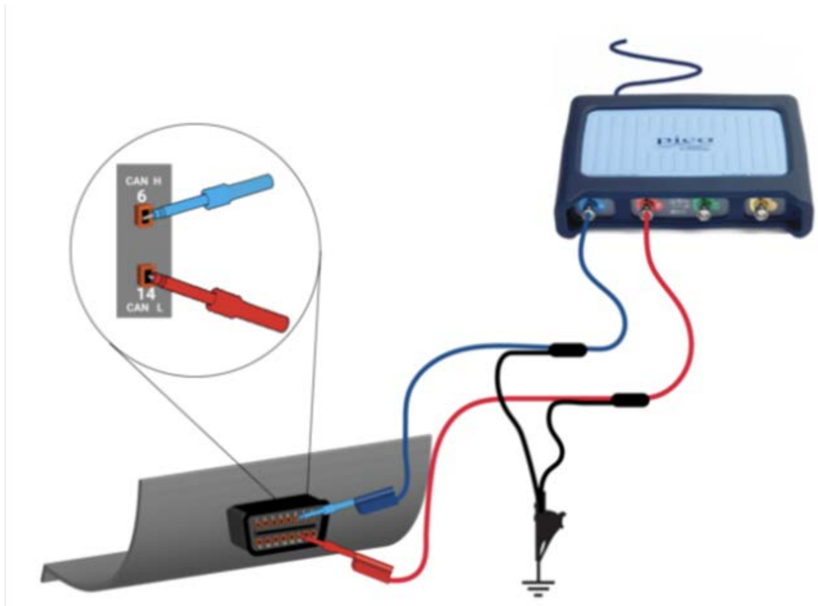


SELECT NETWORK



Guided Test

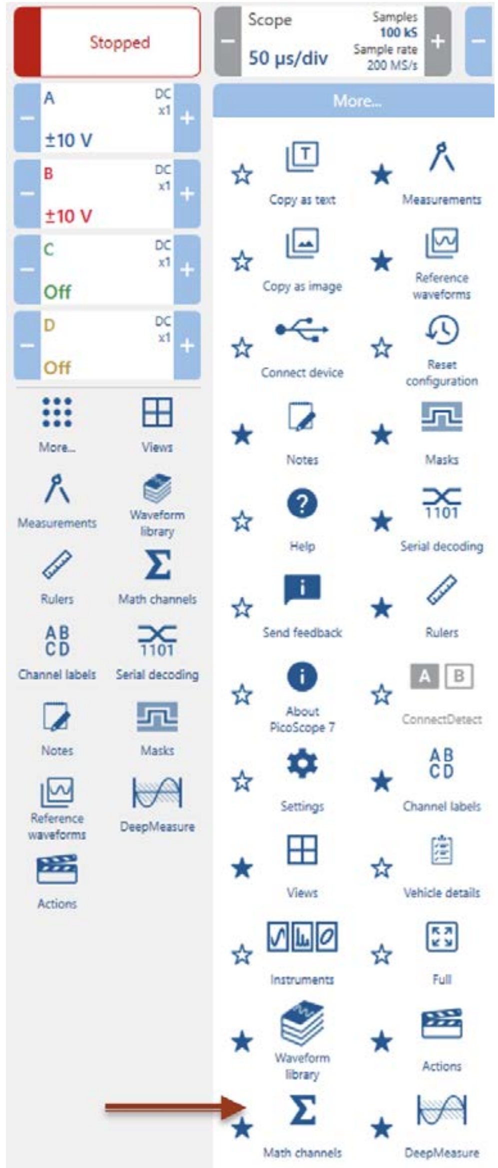
Pico scope offers a guided test menu that will walk the technician through connecting and checking CAN communication. Select guided test menu and then select communications. Once the communication menu opens select CAN bus. Select bus type or example. It is the CAN bus.



Serial Decoding

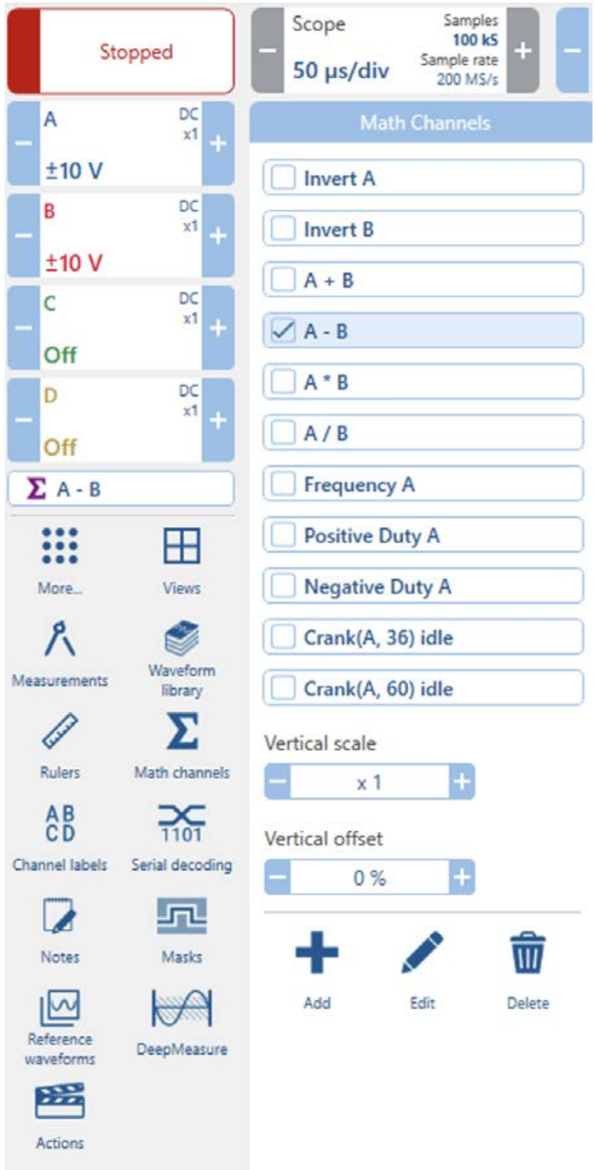
They located the CAN bus access point. If the DLC accesses the CAN bus, connect to pins 6 and 14. If not, find the CAN bus and back probe a connector. Connect channel A to CAN H and channel B to CAN L. Use battery ground for scope. Start the vehicle or turn the key on to activate the CAN bus. Capture some CAN frames and then stop the scope. Use the waveform buffer to move through the frames.

Set the scope ten times faster than the signal to get a suitable waveform that can be decoded. Most CAN waveforms run at 500,000 bps, so the scope should be set to 5 million samples per second. If the scope is not set to the proper speed, the decoding software may not decode properly and give you false results.



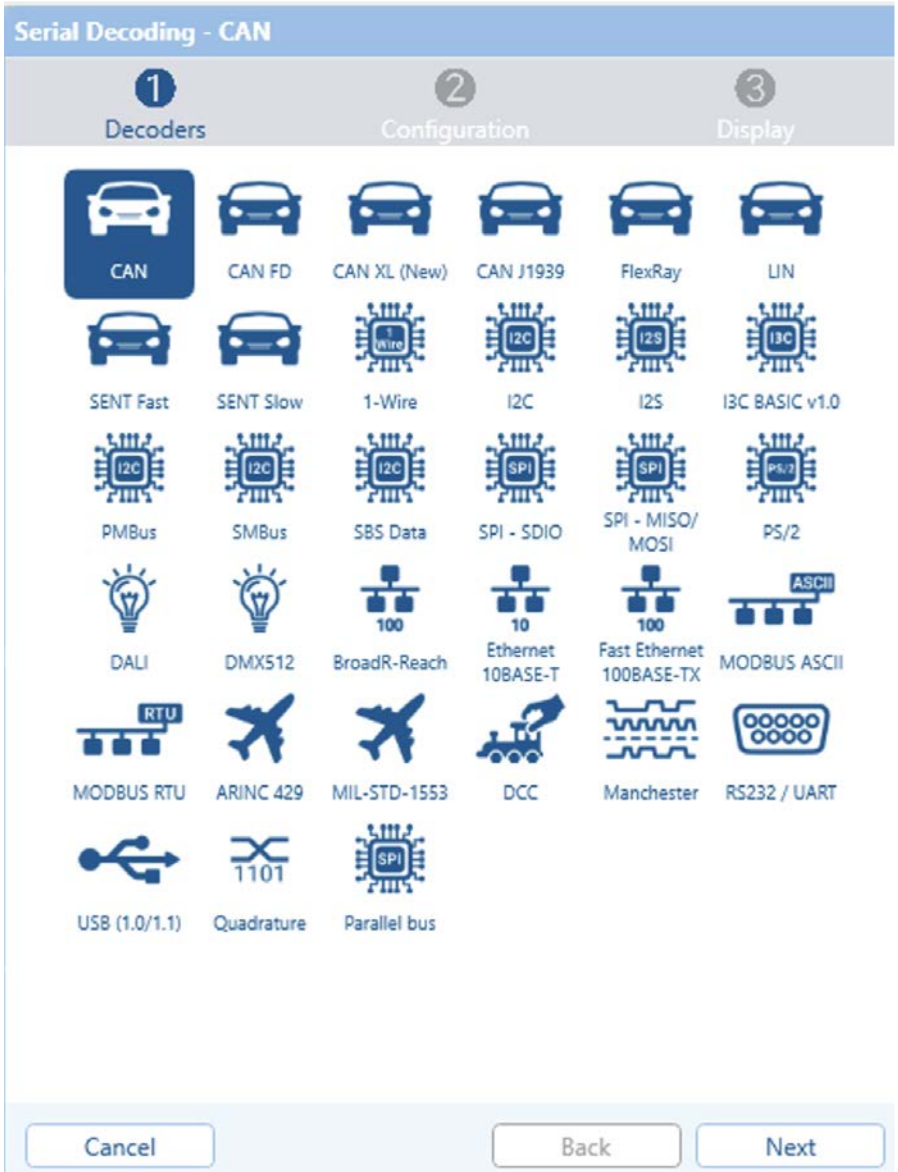
Serial Decoding

Set up a math channel to view the difference between the two channels. The math channel can be found in the menu section.

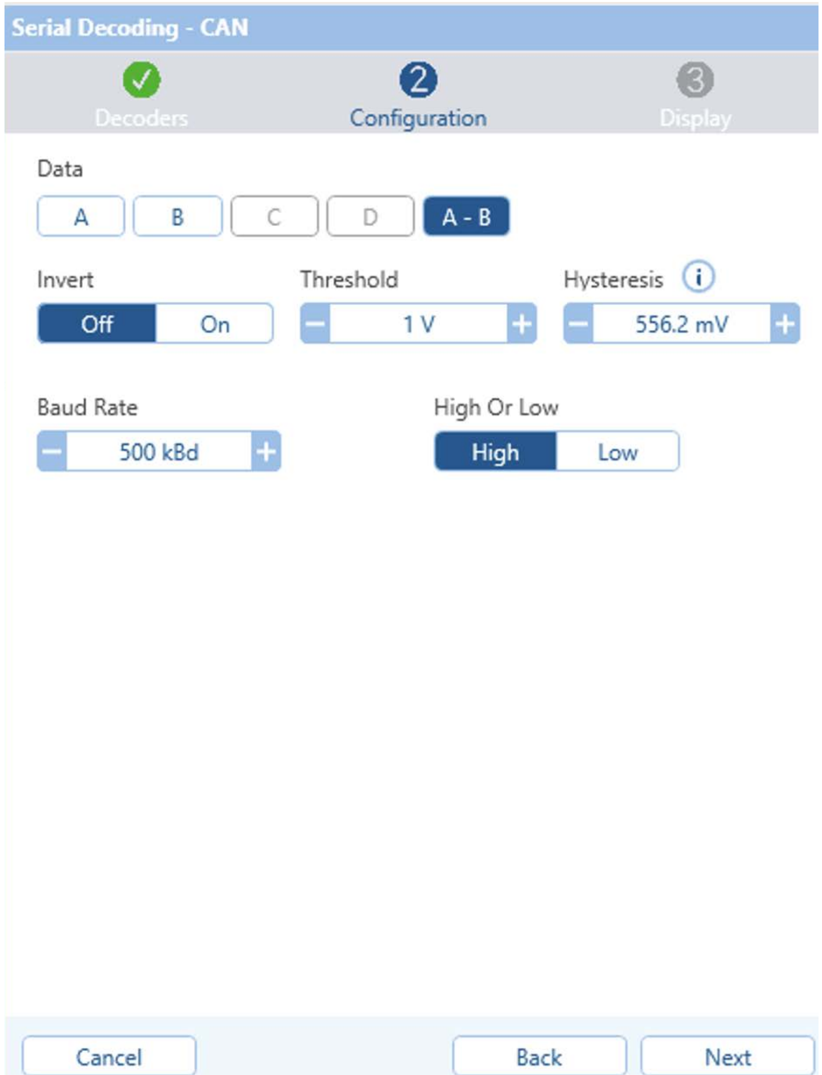


Serial Decoding

The math channel we chose is A-B. This will give us the difference between the two CAN lines.

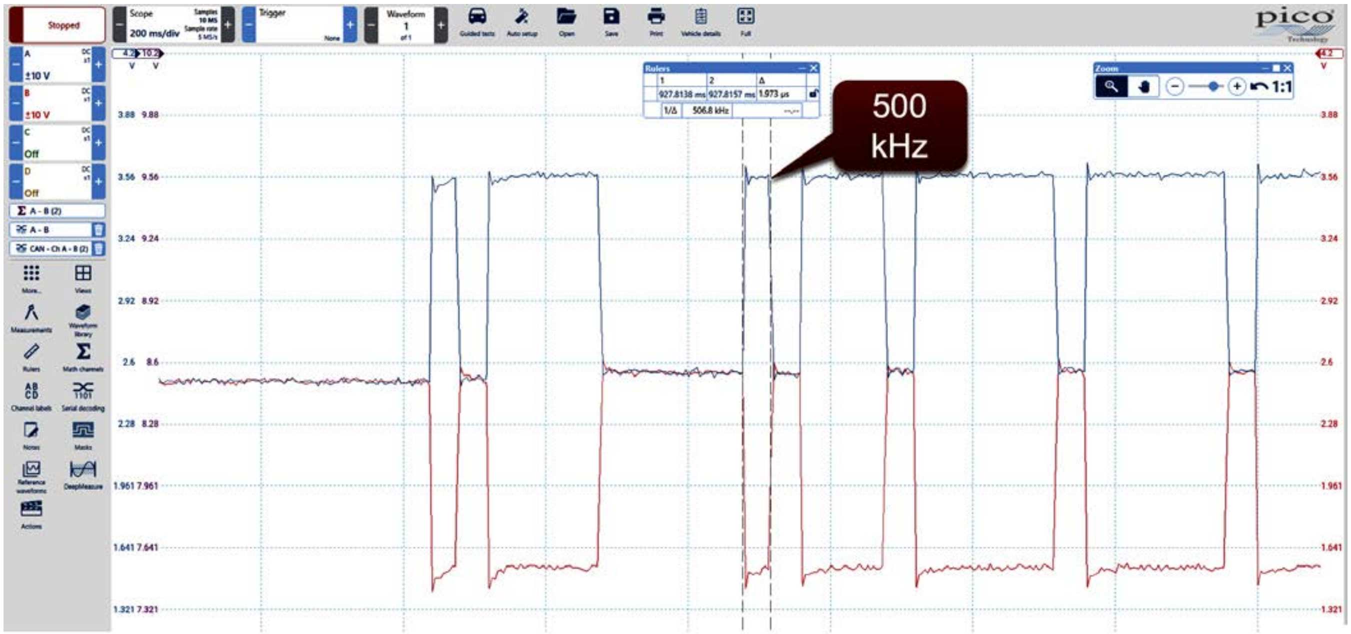


Serial Decoding
Next select serial decoding and then CAN.



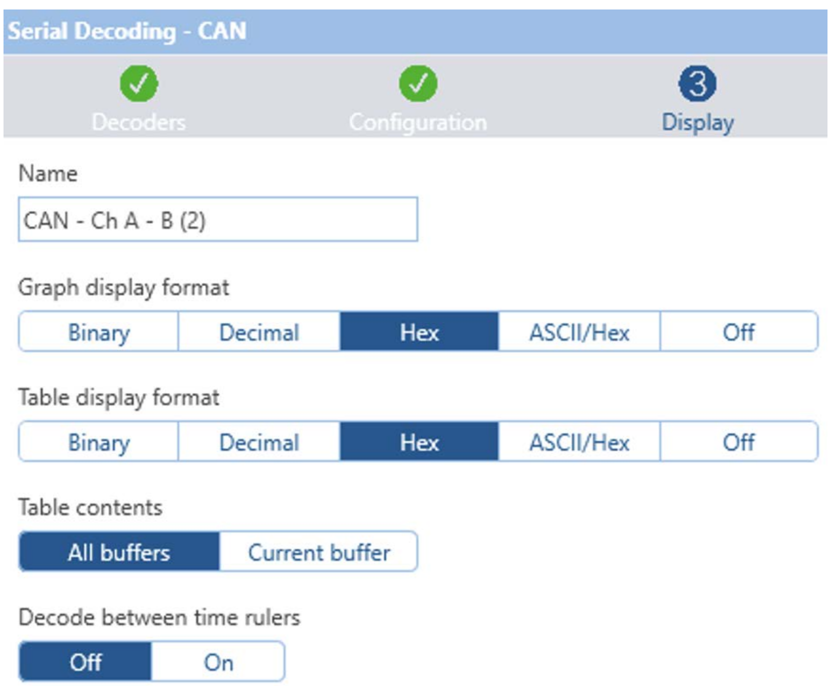
Serial Decoding

Adjust voltage threshold. The voltage is typically 2 volts between the channels, so that you can select 1-2 volts for the threshold. The hysteresis can be set to 400-500mv. Next, select the baud rate. You can measure the baud rate using cursors, which we will show later. For now, we are going to pick 500kbps. The software will not decode the message properly if the baud rate setting is incorrect.



Serial Decoding

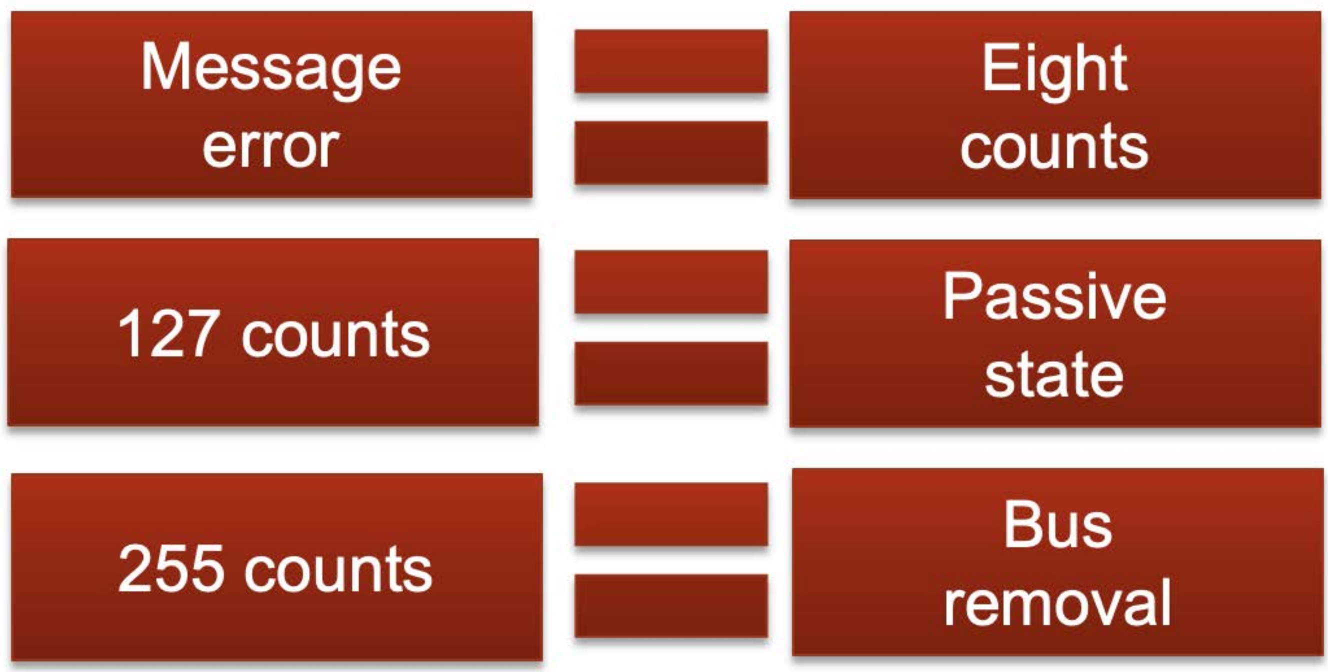
You must insert the correct baud rate to ensure serial decoding is accurate. If the baud rate is unknown, you can measure it by using cursors, as shown, to measure the shortest bit time. In our CAN bus example, the baud rate measures right at 500kHz. This is normal for the CAN data bus.



Serial Decoding

Select HEX for graph display. Select HEX for table display. You can choose to decode all buffers or decode between rulers.

Serial Decoding
CAN H and CAN L waveforms are on top, and the A-B channel is on the bottom in the purple trace. It is necessary to have a purple trace for every signal bit and that the waveform is consistent with voltage. Serial decoding will be shown below the waveform. Error frames will be highlighted in red.



Message Errors

When another module flags a module transmitting a message error, the sending module gets eight error points flagged against it, and the message is destroyed. When the module gets 127 counts of 16 messages, it becomes passive. On every excellent message the module sends, it gets to take away one error count against it. In the passive state, the module can continue to transmit messages; it can only send passive error flags for other module's messages. In passive mode, the other modules do not recognize any error counts created by the passive module, so their messages are not destroyed.

During passive mode, if the module continues to get counts against it for sending corrupt messages, it will be taken off the bus when the counts rise above 255 counts. The module is taken off the bus and may be bad, but it may not be as well. If a module listening on the bus has clock drift, that module will flag error counts against the sending module even if the message is good. This type of failure is common. The technician automatically thinks the module that has been taken off the bus is the bad one, but that is not always the case.

Module Kicked Off Bus

When you cannot connect to a module with a scan tool on the bus, it may be due to that module being kicked off the bus. Just because the module was kicked off the bus does not mean the module is bad. Another module with clock drift may have flagged enough message errors against that module in the CRC field to get the module removed from the bus.

Replacing the module will restore the module to the bus, but it may be kicked off again soon due to the module being on the bus with a clock drift. Before replacing a module, perform a hardware reset by disconnecting battery cables and touching them together; before performing this procedure, ensure the vehicle only has one battery. The hardware reset will return the module to the bus unless the module has failed.

After hardware reset the module returns

Error codes for multiple modules

One module has no codes

After Hardware Reset
After performing a hardware reset, if the module returns, there is a good chance that the module is not bad, especially if there are error codes for all the other modules. The module that does not have any codes charged against it is the offending module. It is hard for technicians to think that the module with no codes is bad, but if that module has a clock drift, it will set codes against all the other modules and typically get one or more of them kicked off the bus.

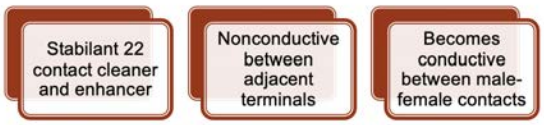
No codes are set against the offending module because the clock drift does not affect transmission, and all the other modules synchronize their clocks to the sending module. The clock drift only affects listening mode. When the module is listening to the clock drift it causes that module to set error codes against other modules.

Moisture Or Water Intrusion

Moisture or water intrusion is a big problem with connectors and board contamination. If the module is under the seat, it may be exposed to moisture or water ingress. Water can cause bad connections and damage to the board inside the module. The vehicle may have been in an accident and had the body repaired, causing issues with body drains or water leaking into the vehicle. Anytime you are dealing with communication issues these are things to consider. The module in the picture came from an Audi Allroad. The module was located under the passenger seat.

Terminal Oxidation-Fit

Terminals must be clean, snug, and unable to be pushed back out of position. Communication terminals are extremely low current and will not tolerate oxidation or contamination. Technicians need to test terminal tension and use a product to clean terminals to restore circuit integrity.



Terminal Enhancer

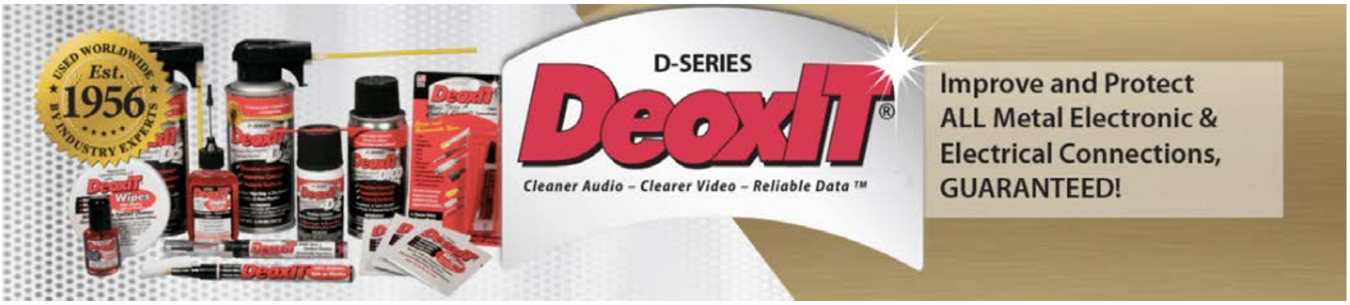
Stabilant22 is a contact cleaner. It is initially non-conductive but switches to a conductive state when placed between two male and female contacts. It will not become conductive outside the connection and will not bridge between an adjacent terminal, causing signal leakage.

One point is of critical importance: Stabilant22 is not “just another contact cleaner.” It is an initially non-conductive long-chain organic block polymer that switches to a conductive state only when used in a thin film within the interstices of a contact pair. It remains non-conductive on the insulating surface between adjacent contacts. Thus, remaining resident within the contacts improves connector reliability within a contact set without causing signal leakage problems between adjacent contacts.

Its advantage lies not in making new contacts better but in keeping contacts operating as if they were new while making old contacts function perfectly despite the ravages of time and adverse environments. In simple terms, Stabilant22 provides the reliability of a soldered joint without producing a mechanical bond. It also exhibits excellent lubricating properties but does not creep; it is exceptionally stable and has a shallow vapor pressure. Its continued regular use will enhance contact reliability for many years.

Stabilant22 and Stabilant22A (its Isopropanol-diluted form) are used in:

- Sound recording studios
- Radio & TV production & broadcasting
- Computers
- CAD/CAM systems
- Multi-pin, XLR & RCA plugs & jacks
- Printed-Circuit board I/O connectors
- Patch bays • Relay contacts
- Battery contacts
- Telephone, cellular, wireless communications
- Speaker terminals
- Screw, spade & ground terminals
- Hospital patient monitors
- Biomedical electronics
- Avionics (it is TSO’d)
- Air marine navigation equipment
- Robotics equipment
- Aircraft and helicopter maintenance
- Satellite communications (including encryption decryption)
- Military electronics (Supplier Code & NATO/Military Stock Numbers available)



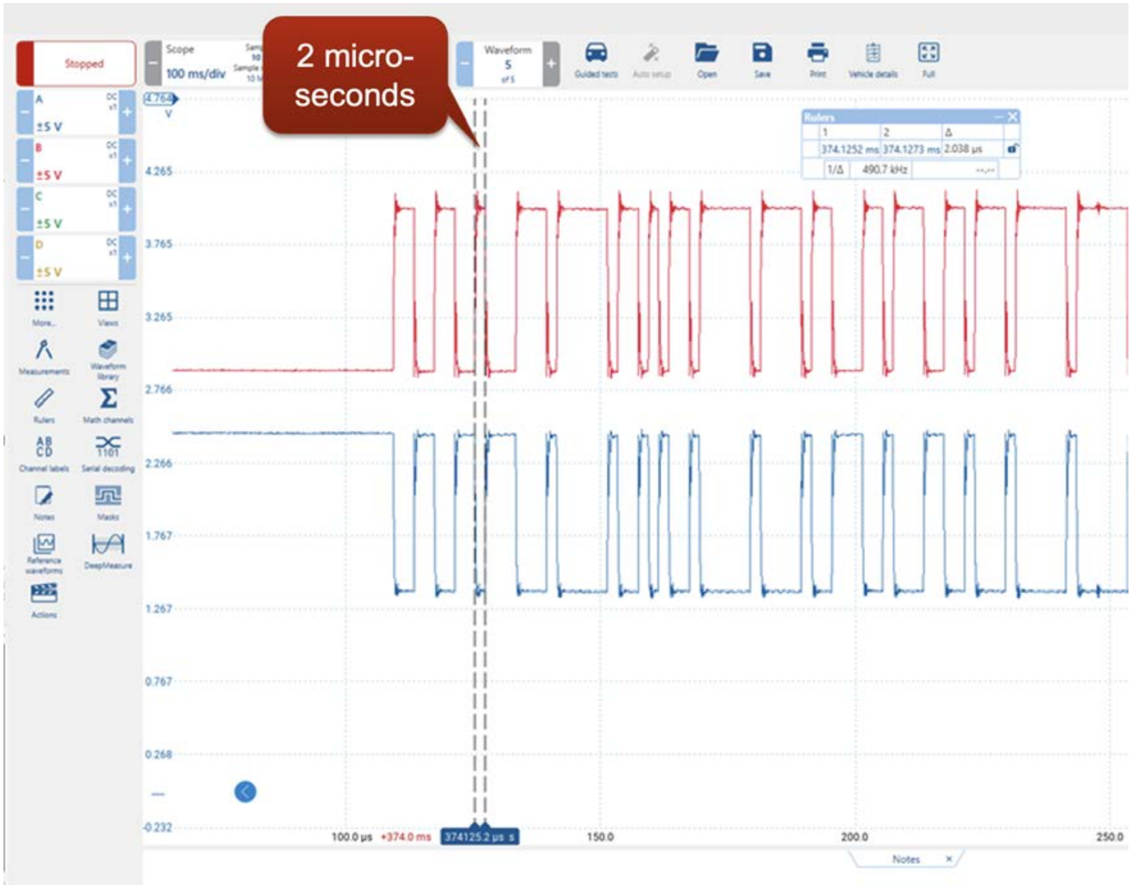
DeoxIT

Another product on the market that can clean and enhance electronic connection is DeoxIT. This product dissolves oxides and sulfides to restore contact integrity. DeoxIT forms a protective layer on the terminals to reduce future terminal connection problems.

DeoxIT® D-Series contact/connector treatment dissolves oxides and sulfides that form on metal surfaces. This restores the contact's integrity and leaves a thin microscopic layer that protects the metal. Special additives improve conductivity and prevent dissolved oxides from re-attaching, keeping them in suspension and allowing them to be easily dispersed by the mechanical action of the contact.

Lab Scope Test

Check your equipment before condemning the CAN bus due to an ugly waveform. Lab scope leads may be bad. Test the leads or swap them out if there is any concern. A lab scope channel can also malfunction. Both problems can cause a CAN waveform to look bad even if the waveform is good. Move to a different channel on lab scope to see if signal improves.



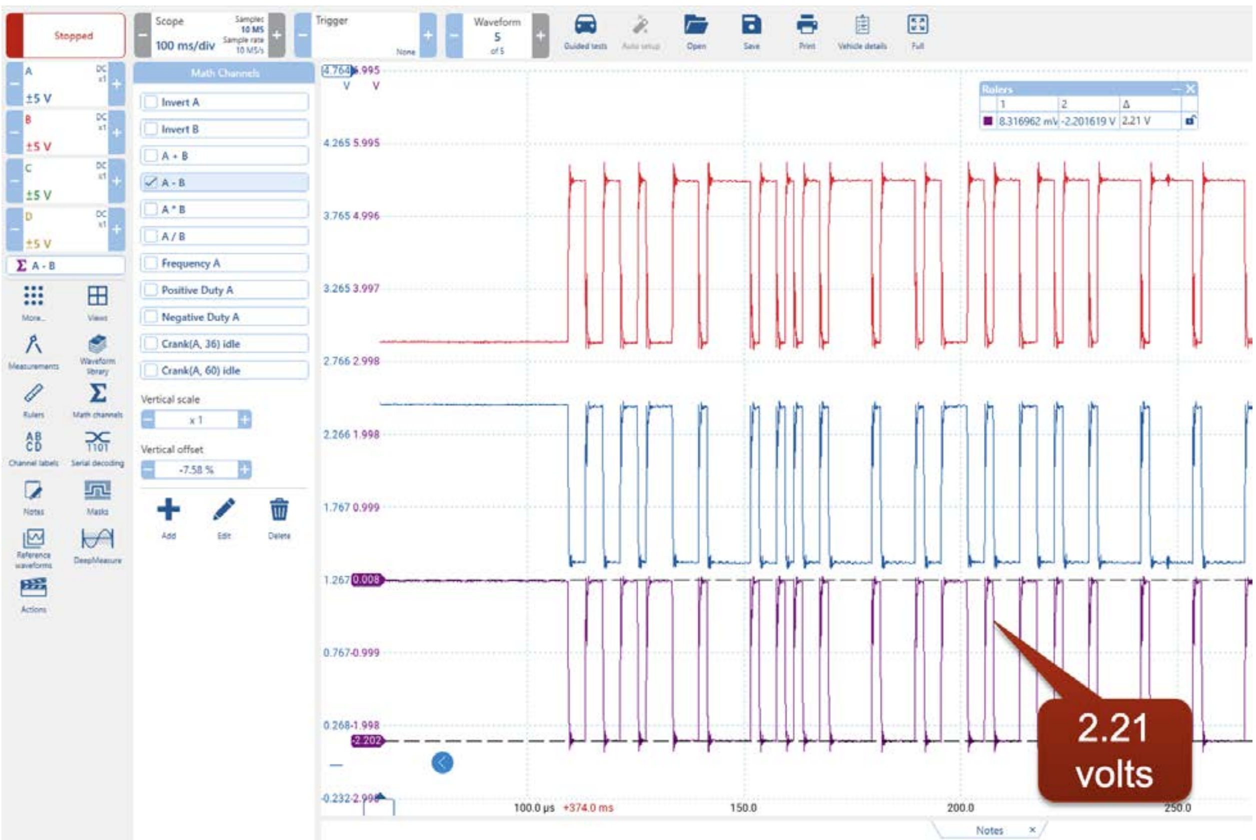
Lab Scope Waveform Rules

Rules of delivery need to be observed in the CAN bus waveform. The baud rate is based on each bit's value's clocking rate or speed. In our example, we measure the fastest bit on the scope screen. The bit rate is measured at two microseconds, indicating a baud rate of 500kbs. If the bit speed is four microseconds, then the speed is reduced to 250kbs. Also, observing the waveform, you can only have five like bits in a row before we get some stuffing.

Suppose we have more than five bits without a stuffed bit. In that case, the other modules will not acknowledge the message, which will lead to the original module to continue broadcasting over and over, hogging the bus so no other module can gain access.

Measure bias voltage: it should be close to 2.5 volts on both CAN lines. If the bias voltage is high, a module may have lost ground. If bias voltage is low around .9 to 1.7 volts modules may have lost ignition-on power or the modules are in sleep mode. The termination resistors may be lost if the modules are ringing and do not stay at 2.5 volts.

Each module that is still on the bus but is not powered on removes approximately .83 volts from the bias. Both bias voltages on CAN H and CAN L will be affected the same when modules are not powered on. If enough modules are not powered on it can drive the bias down to zero. The CAN bus may still differentiate the voltages at this time, but the bias will be incorrect. If only one CAN line is affected, it points to a single module failure internal to the module.



Lab Scope Waveform Rules

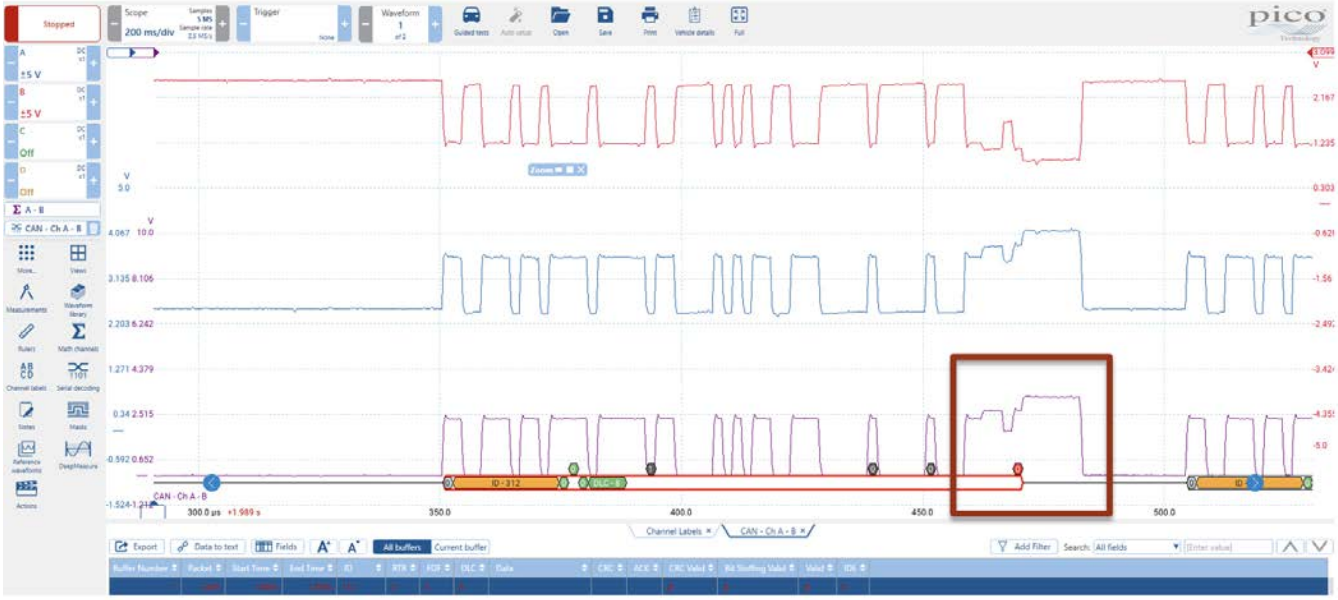
Communication depends on the voltage level difference between CAN H and CAN L. To create a binary zero, a voltage difference must occur. The voltage difference varies between brands, but the 1.5- and 3.0-volts difference is normal. The Pico scope makes reading the voltage difference easier by opening a math channel on the scope. Our waveform example above CAN H is channel B in the red trace, and channel A is CAN L in the blue trace.

The math channel is set to subtract A minus B. The voltage difference on the math channel is measured with the cursors and displays 2.21 volts. The voltage difference falls between the 1.5-volt and 3.0-volt expectations. If the voltage difference expectation is not met, communication will be broken. The math channel only indicates binary numbers are being created. It does not tell you if the message is corrupt.

Rules Broken

The rules have been broken in this communication string. The string has been highlighted in the serial decoded field. The waveform does not follow the SAE rules at this point. The communication has broken down and the message is invalid.

The time cursor shows the blue channel CAN H goes high and stays there for over 24 microseconds. This should be 2-4 microseconds. The voltage also goes too high. It travels over 4 volts. This indicates that a module is starting to malfunction intermittently.

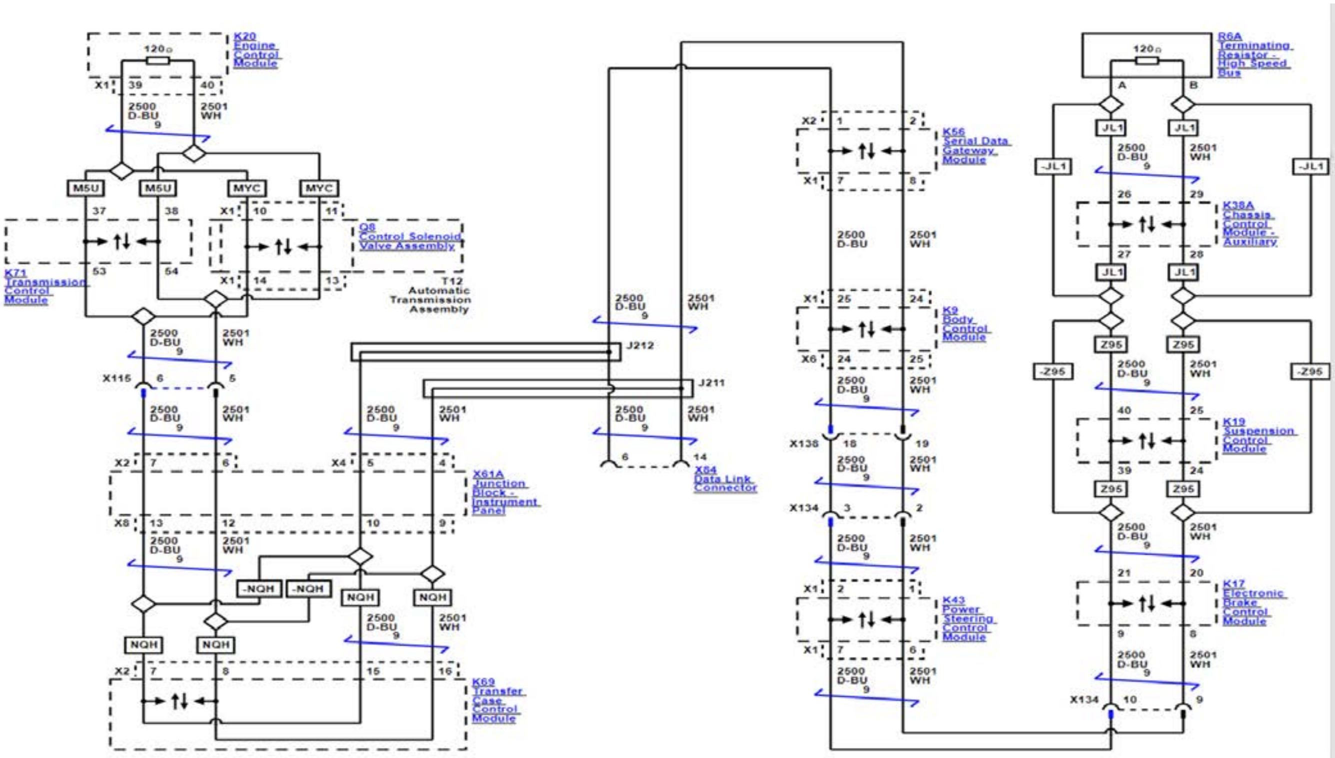


Rules Broken

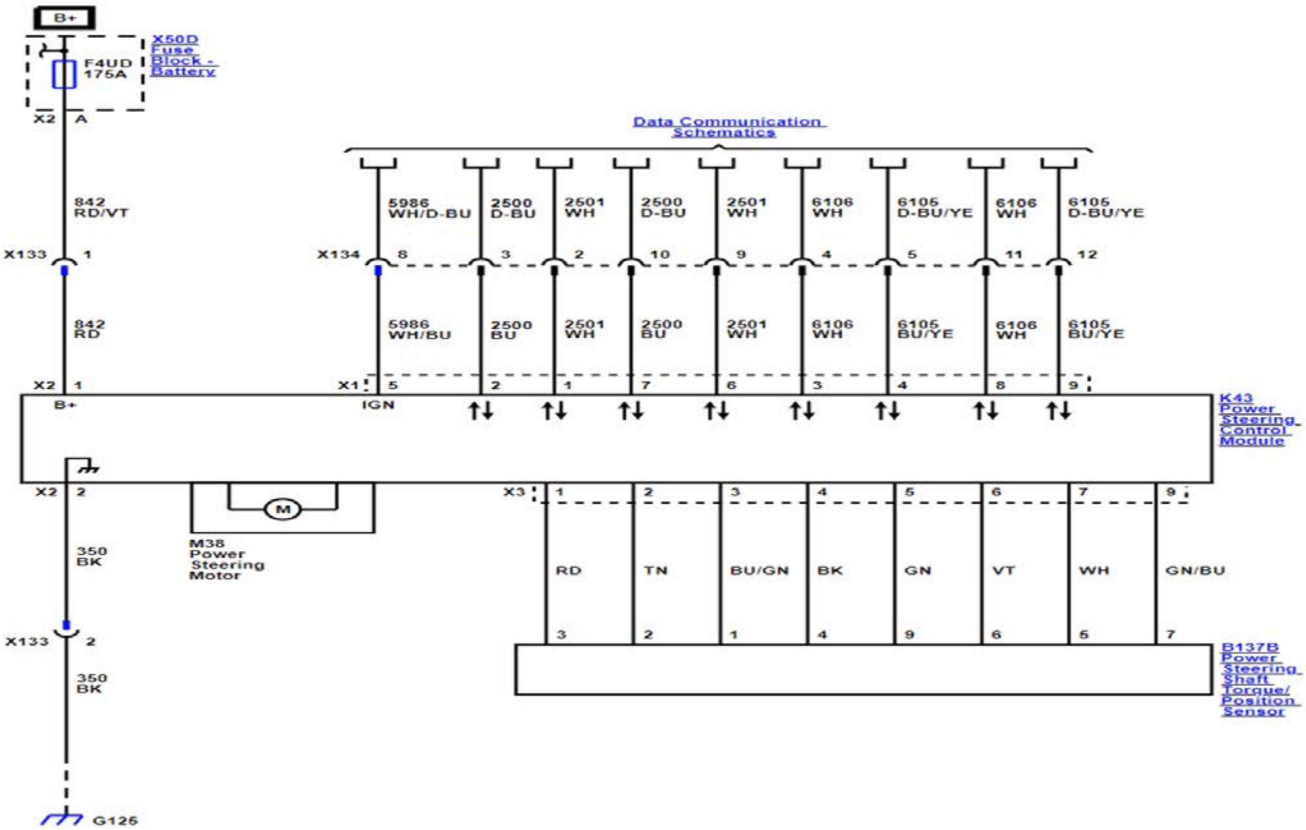
This is another waveform that is breaking down. We have added a math channel to check the difference between the two signals. You can see the math channel breaking down in purple. The message was decoded at the bottom of the screen, but the decoding did not pass. ID 312 message fails CRC, bit stuffing, and valid message check. The data field is blank. The packet is no good.

No Comm Prep

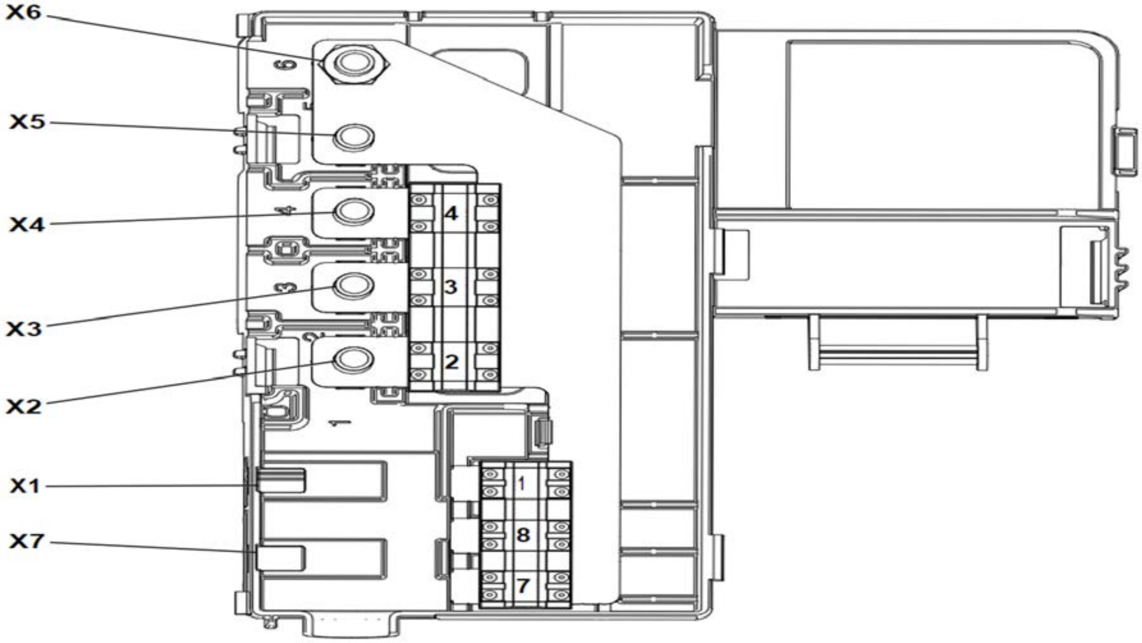
A 2017 Chevrolet Silverado has no communication with the PSCM. Codes in every module, steering is hard to turn. Our scan tool backs this up as there is no communication with the SCM. Let's dig in and figure out how to address this concern quickly.



Pins 6 and 14 show there is 60 ohms resistance. What does this tell us? What would be the next checks in this system?



We need to load test the power and ground circuits.
How would we load test these in your shop?
Can we rule out F4 quickly?
Can the wiring diagram be used very quickly to divide and conquer?



X4 is only for the power steering rack.
If this was controlling other things how could we test this out quickly?

Chapter 5 - Chevrolet Equinox Diagnostic Example

2019 Chevrolet Equinox
Vin: 3GNAXWEU9KL173902
Date: 2/15/24
Mileage: 99458
Complaint: Check engine lamp is illuminated
Observations: Vehicle was at the local dealership; they were quoted over $6k in repairs.

2019 GM Vehicle Diagnostic Report

Vehicle Information
2019/GM/Chevrolet/
VIN: 3GNAXWEU9KL173902
Odometer Reading: 99458.51miles
License Plate: TWX127

Customer Information
Name: --
Tel: --

Device information
Scanner: MaxiSys Ultra
Version: V13.00
Serial Number: V1DGP4V02135
Repair Order Number: V1DGP4V0213520240206131837541

System Scanned(19)

System	Status/DTC
Engine control module	15
Transmission control module	0
Electronic brake control module	0
Inflatable restraint sensing and diagnostic module	0
Instrument cluster	0
Power steering control module	0

DTC (22)

Engine control module (15 DTCs)

DTC	Description	Status
P0627:00	Fuel pump relay control circuit open	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Not run MIL status : Requested
P1015:00	Reductant control module sensor serial communication circuit low voltage	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Not requested
P1016:00	Reductant control module sensor serial communication circuit high voltage	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Not requested
P203C:00	Reductant level sensor circuit low voltage	DTC history status : History Since DTC clear : Passed and failed Last test : Passed

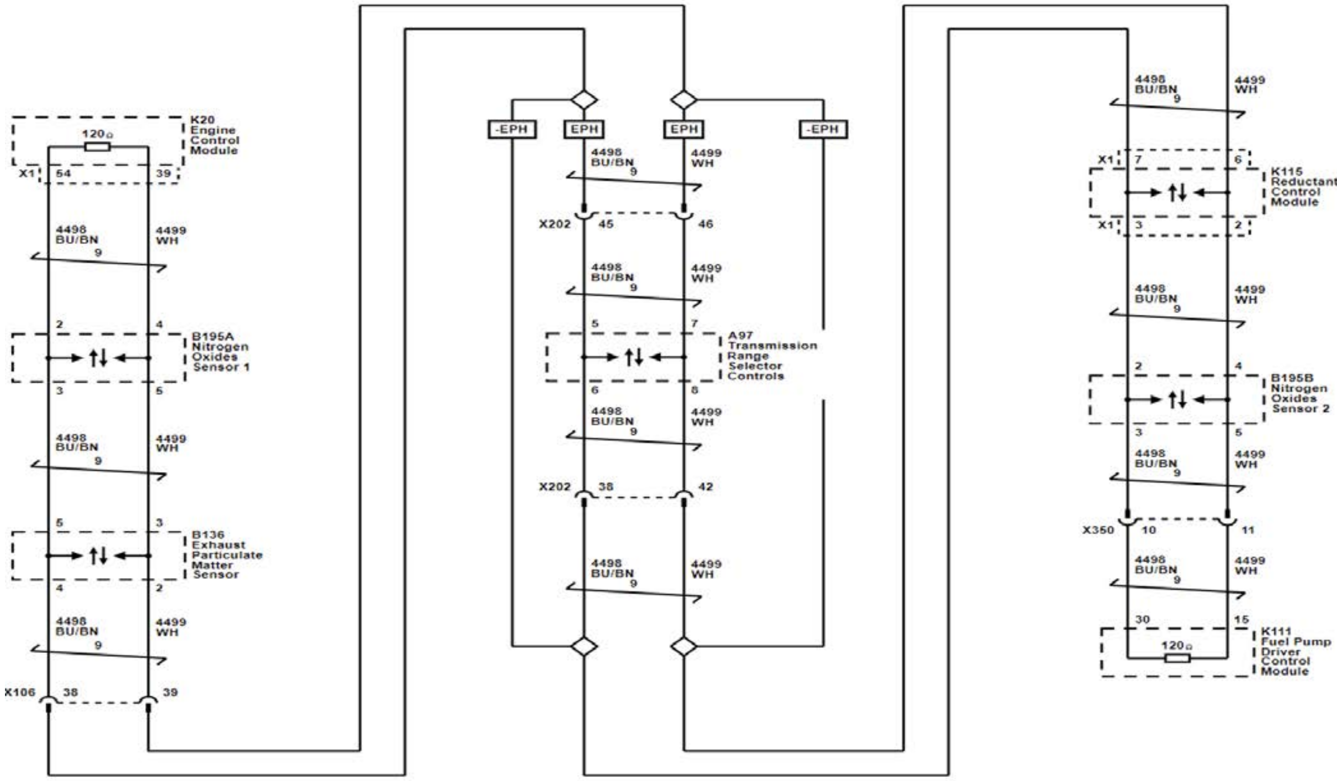
P204C:00	Reductant pressure sensor circuit low voltage	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Not requested
P205C:00	Reductant tank temperature sensor circuit low voltage	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Not requested
P21CB:00	Reductant control module supply voltage low voltage	DTC history status : Not history Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Not requested
P220A:00	Nox sensor 1 supply voltage circuit	DTC history status : Not history Since DTC clear : Failed Last test : Failed current DTC This ignition cycle : Not run MIL status : Not requested
P2457:00	Exhaust gas recirculation (EGR) cooler low efficiency	DTC history status : History Since DTC clear : Failed Last test : Failed current DTC This ignition cycle : Not run MIL status : Requested

P257E:00	Engine hood switch circuit low voltage	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Requested
U010E:00	Lost communications with reductant sensor module	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed MIL status : Not requested
U029D:00	Lost communication with nox sensor 1	DTC history status : History Since DTC clear : Failed Last test : Failed current DTC This ignition cycle : Failed MIL status : Requested
U029E:00	Lost communication with nox sensor 2 module	DTC history status : History Since DTC clear : Failed Last test : Failed current DTC This ignition cycle : Failed MIL status : Requested
U02A3:00	Lost communications with particulate matter sensor	DTC history status : History Since DTC clear : Failed Last test : Failed current DTC This ignition cycle : Failed MIL status : Requested
U18A2:00	Lost communications with fuel pump driver control module	DTC history status : History Since DTC clear : Passed and failed Last test : Passed This ignition cycle : Passed and failed MIL status : Requested

What is noticed about all of the codes that were present?
With this many codes and knowing it just came from another facility what would you do?
We are going to clear and see what returns.

And just as luck would have it…..
All of the communication codes were returned almost instantly.
U029D:00 Lost communication with nox sensor 1 module
U029E:00 Lost communication with nox sensor 2 module

U02A3:00 Lost communications with particulate matter sensor
U18A2:00 Lost communications with fuel pump driver control module
U010E:00 Lost communications with reductant sensor module

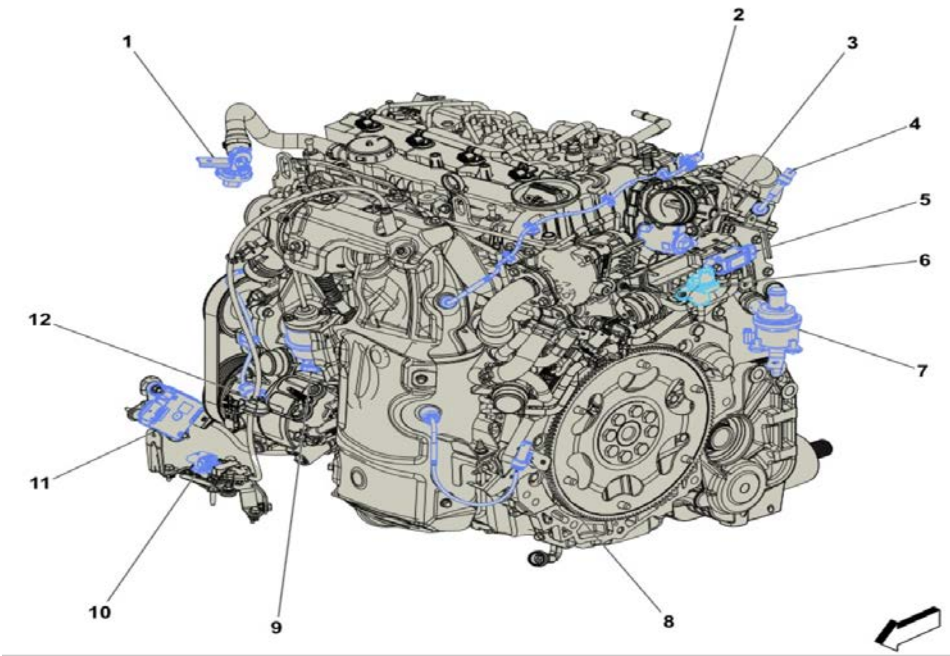


We need a game plan.

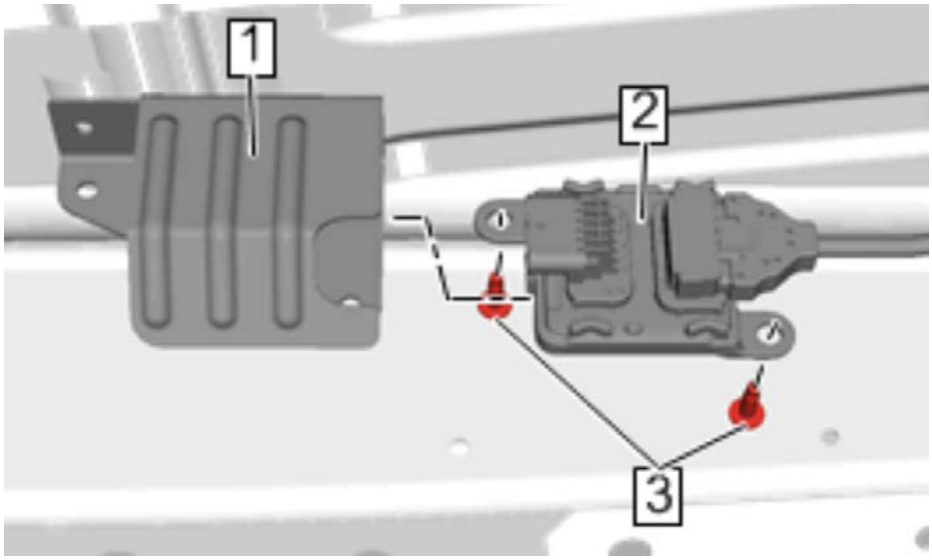
We know this is setup like any other high-speed network. There are 2 120Ω resistors in parallel we should read;

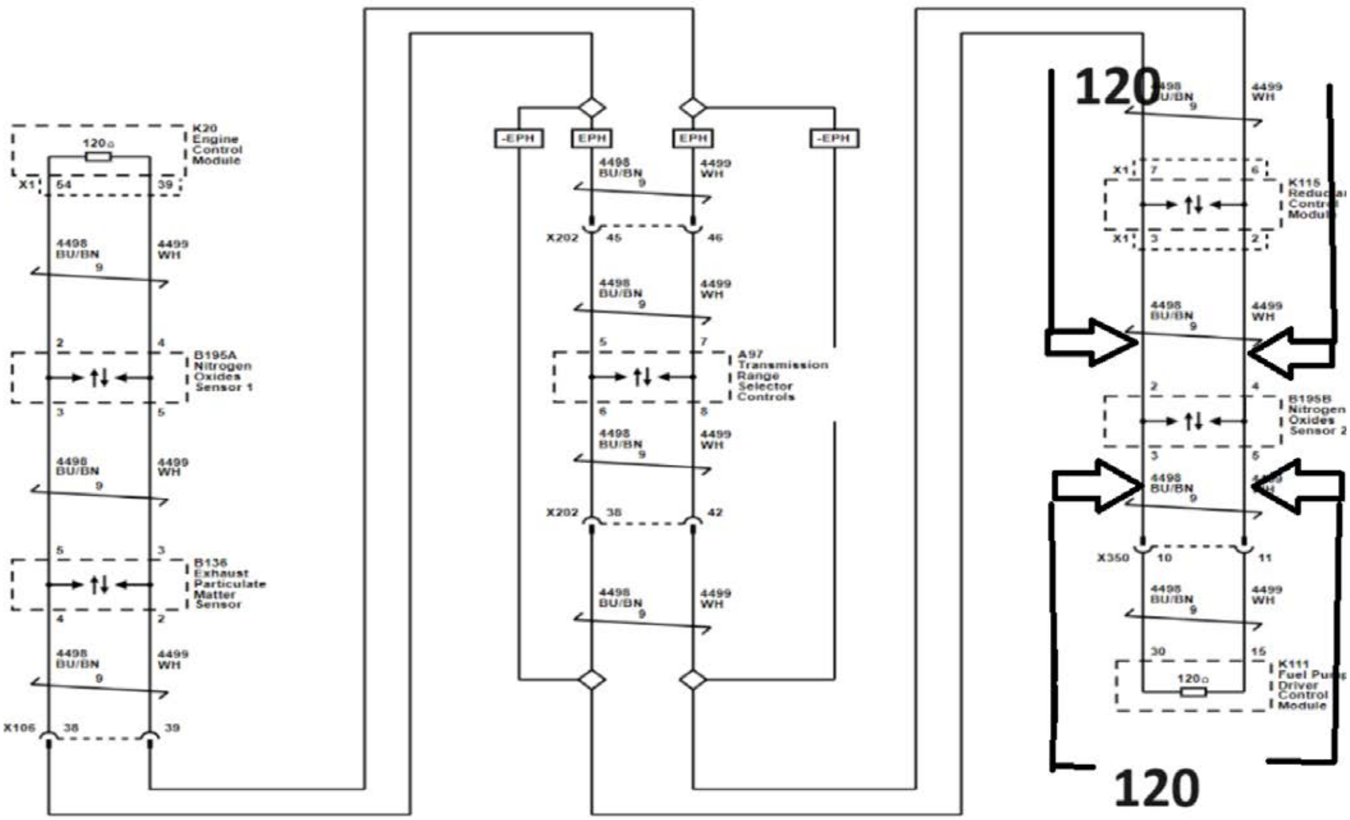
- 60Ω.
- This makes the idea of dividing and conquering easy.
- Divide the system in half or at the easiest test point since there is not a DLC connection.
- The divide and conquer method works great on any bus system.

B193B	Charge Air Cooler Outlet Temperature Sensor	LH7	In the engine compartment, on the outlet from the turbocharger housing	Cooling Fan	B193B Charge Air Cooler Outlet Temperature Sensor
B194	Reductant Pressure Sensor	LH7	Under the left rear of the vehicle, in the reductant tank	–	–
B195A	Nitrogen Oxides Sensor 1	LH7	In the engine compartment, in turbocharger outlet	Front of the Engine 2 of 2 (LH7)	B195A Nitrogen Oxides Sensor 1 (LH7)
B195B	Nitrogen Oxides Sensor 2	LH7	In the exhaust pipe downstream of the selective catalytic converter	–	B195B Nitrogen Oxides Sensor 2
B203	Radiator Coolant Temperature Sensor	LH7	In the engine compartment, back of engine, near the starter	Right Rear of the Engine (LH7)	B203 Radiator Coolant Temperature Sensor (LH7)
B213A	Reductant Level Sensor 1	LH7	Under the vehicle, left rear, inside reductant tank	–	–
B214	Reductant Temperature Sensor	LH7	Under the vehicle, left rear, inside reductant tank	–	–

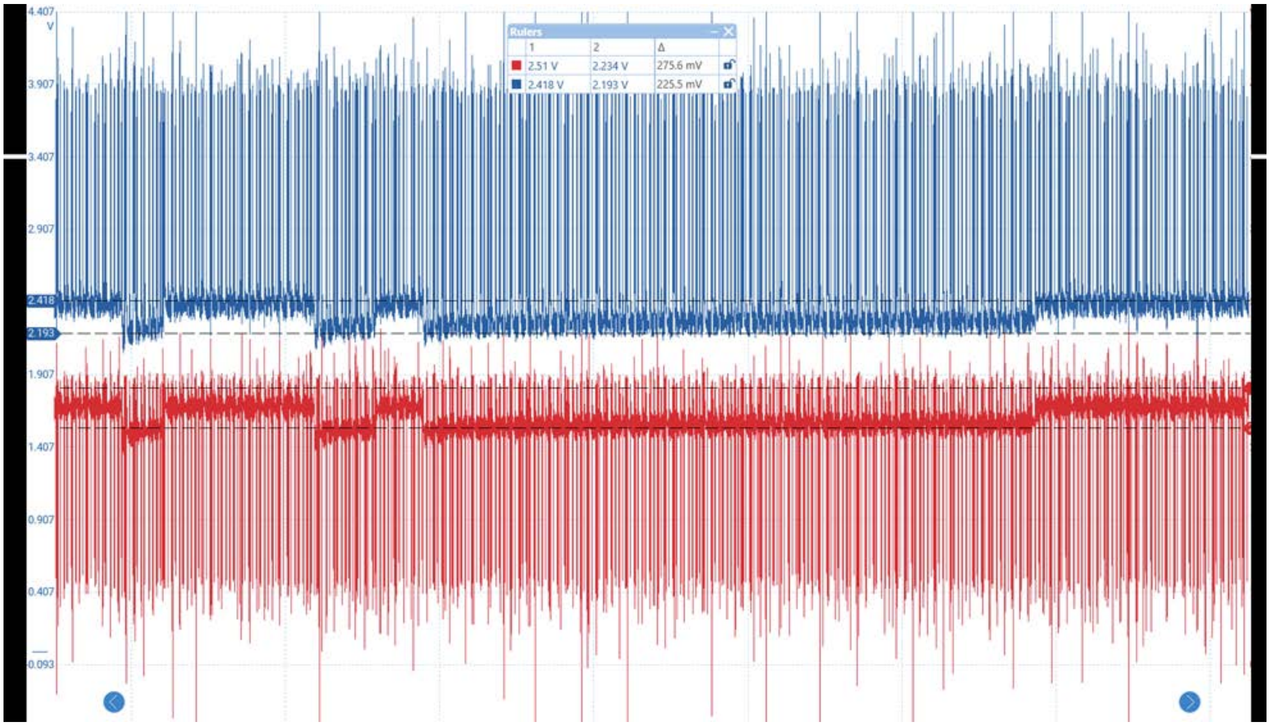


As luck would have it the easiest component to access on this system was Nox2….
The undershield has to be removed to access nox 1 and particulate matter sensor Nox 2 in in the middle of the underside of the vehicle

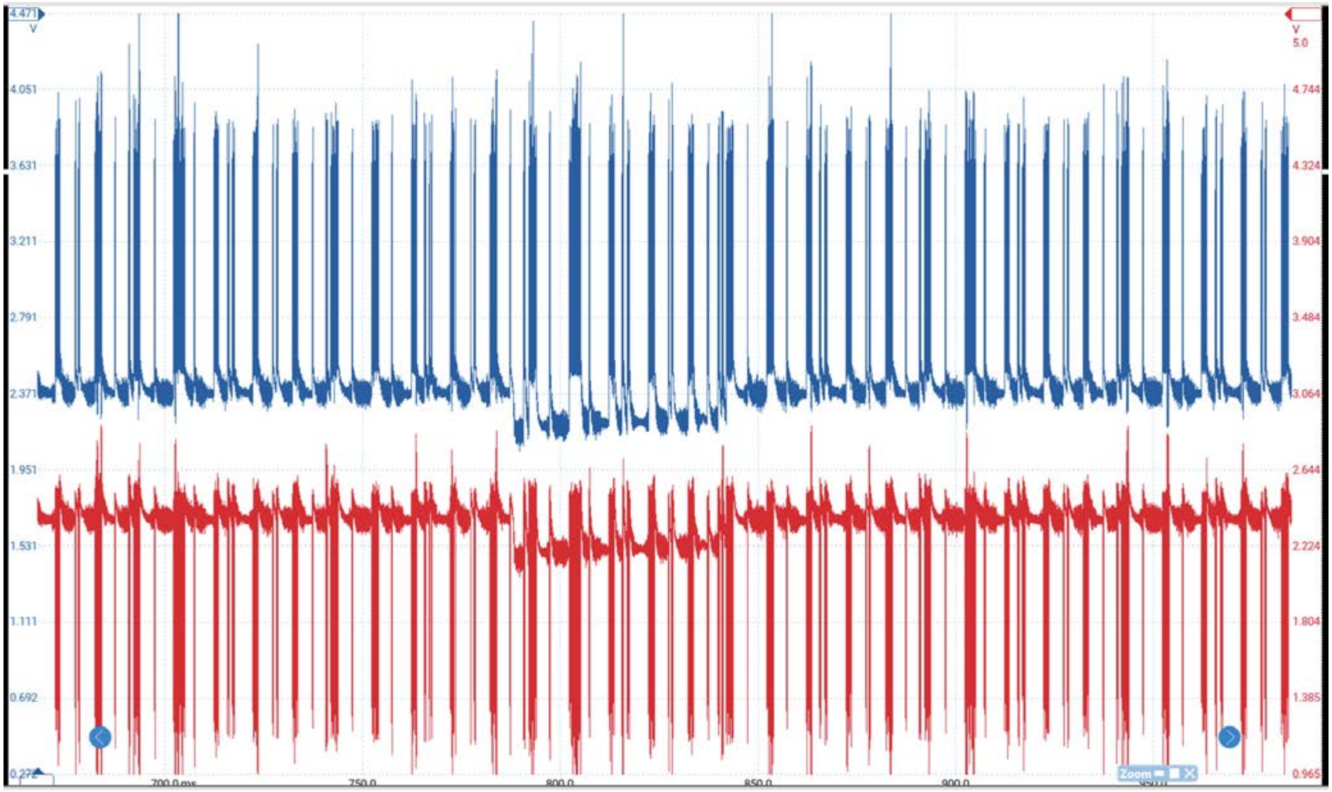

1
2
3


K20 Engine Control Module
120Ω
X1
54
39
4498 BU/BN
4499 WH
B195A Nitrogen Oxides Sensor 1
B136 Exhaust Particulate Matter Sensor
X106
38
39
EPH
-EPH
X202
45
46
A97 Transmission Range Selector Controls
42
120
X350
10
11
B195B Nitrogen Oxides Sensor 2
K111
Driver
Control
Module
30
15

We split the system.
Divide and conquered?
Maybe?
We know we are at 120Ω on each leg.
An open is ruled out.
Now what?
Vehicle is already in the air, let’s grab some scope captures.



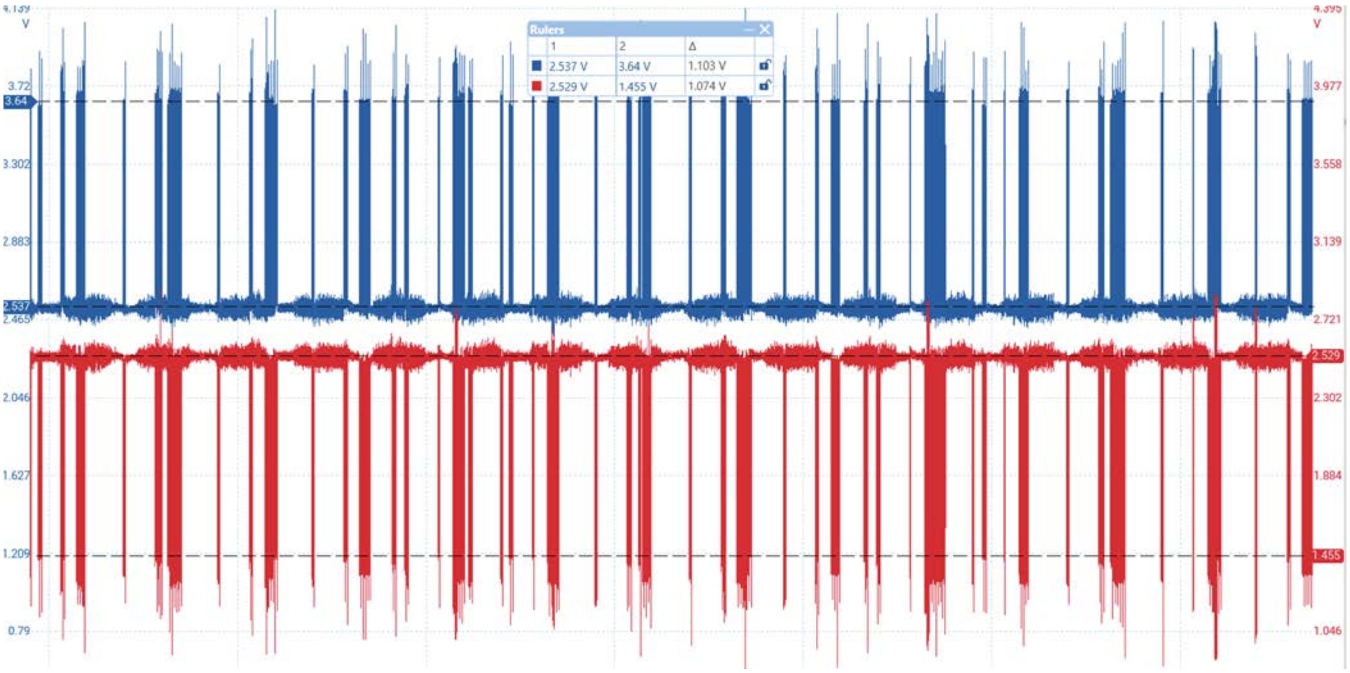
Waveform
This is tested at the reductant control module.
Elevated voltage on the ground side shows a resistance issue.



High Speed GMLAN Circuit Description
When the two-wire bus is at rest the GMLAN-High and GMLAN-Low signal circuits are not being driven and this represents a logic “1”. In this state both signal circuits are at the same voltage of 2.5 V. The differential voltage is approximately 0 V.

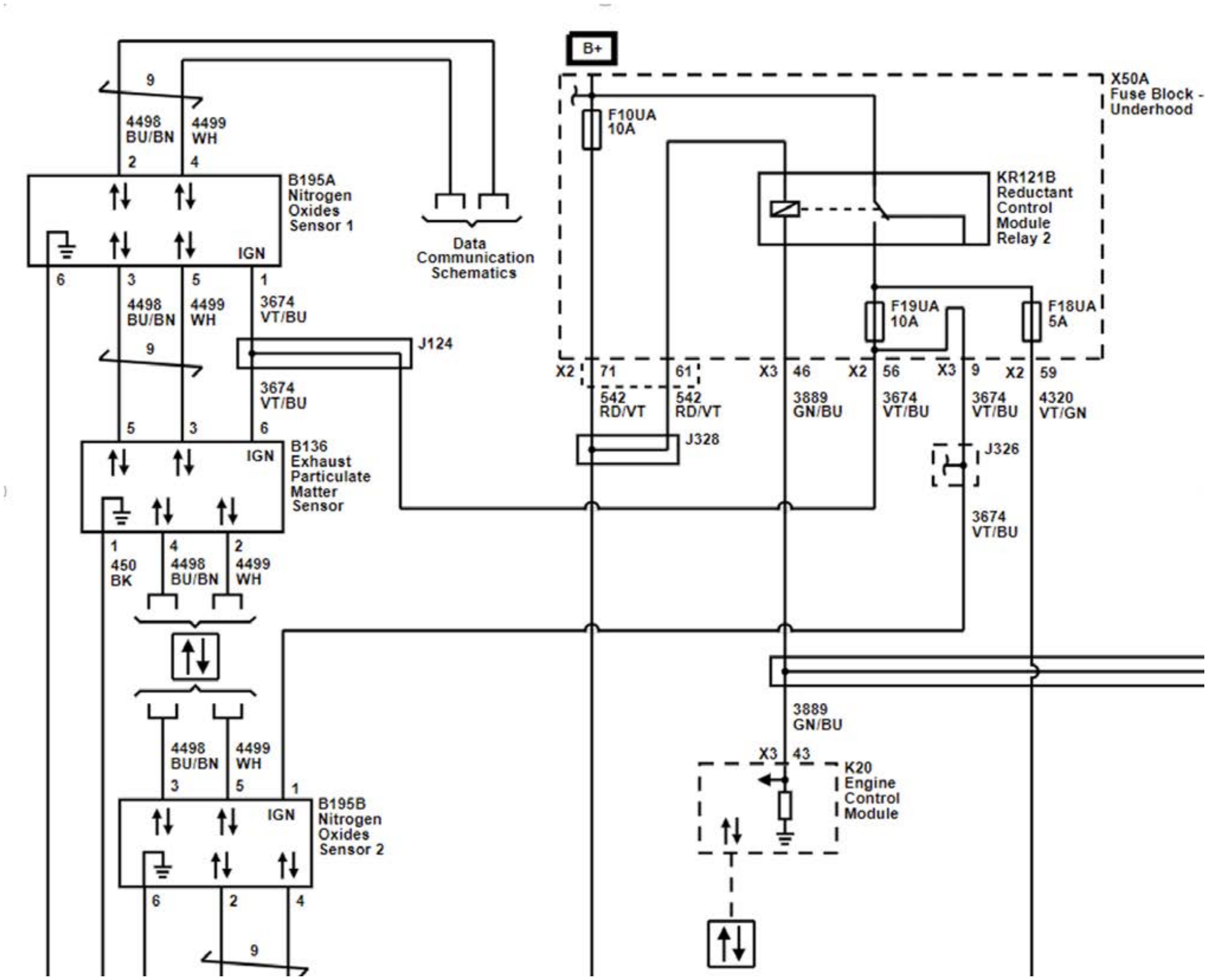
When a logic “0” is to be transmitted, the GMLAN-High signal circuit is driven higher to about 3.5 V and the GMLAN-Low circuit is driven lower to about 1.5 V. The differential voltage becomes approximately 2.0 (+/- 0.5) V. Let’s focus on this. Are we reading this at all?

The data bus is ugly!!!
We will jump each module going in and out.
We start at Nox2 because it is so easy to get to.
And as luck would have it!



145. 2019 Chevrolet Equinox
A new Nox2 was installed.
A system scan was performed.
AND…………..
WE STILL HAVE NO COMMUNICATION

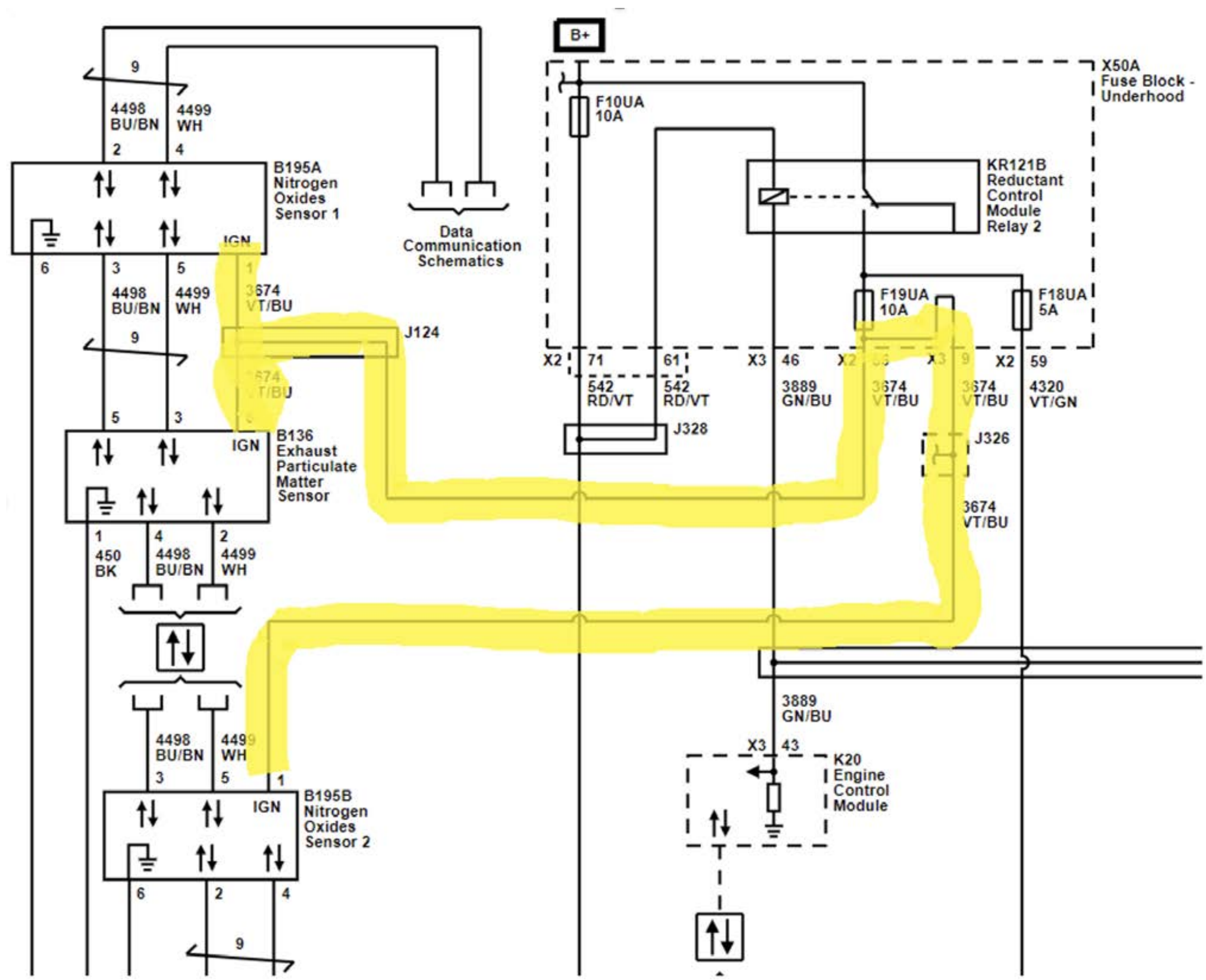
146. 2019 Chevrolet Equinox
The only codes that returned are:
U029D:00 Lost communication with nox sensor 1 module
U029E:00 Lost communication with nox sensor 2 module
U02A3:00 Lost communications with particulate matter sensor



Wiring Diagram
149. 2019 Chevrolet Equinox
The rule with any module before you condemn.

- Power
- Ground
- Communication

We got in a hurry and bypassed the easiest thing we could have checked for while scoping or using our meter.



Conclusion:
A bad Nox2 was pulling down the bus.
When Nox2 failed it shorted internally and blew fuse F19 in the underhood fuse box.
This was not a $6000 repair.

Assessment

Questions

Question: T or F Scan tools may misinterpret the CAN data.

Question: T or F Tools are available to decode the binary data into hexadecimal format.

Question: T or F If a module has been removed from the bus due to corrupt messages it must be replaced.

Questions with Answers

Question: T or F Scan tools may misinterpret the CAN data.
Answer: True. In some instances the scan tool may not decode the data correctly.

Question: T or F Tools are available to decode the binary data into hexadecimal format.
Answer: True. CAN analyzers and some scopes can decode the data.

Question: T or F If a module has been removed from the bus due to corrupt messages it must be replaced.
Answer: False. A battery reset will allow modules to return to the bus.

Summary

- Topology should be observed when diagnosing network problems.
- CAN C must have both termination resistors to function correctly.
- CAN communication is based on voltage differentials.
- CAN network must follow certain rules.
- CAN data can be analyzed in various ways.

Glossary

ACK (Acknowledgment): Signal indicating successful receipt of a CAN message.

Arbitration: Process by which CAN nodes resolve bus access conflicts.

Baud Rate: Speed at which data is transmitted over the CAN bus, measured in bits per second.

Bit Rate: The rate at which bits are transmitted over the CAN bus.

Bit Stuffing: Technique to ensure CAN bus synchronization by adding extra bits.

BUS: The physical communication pathway in a CAN network.

CAN FD (Flexible Data-rate): An extension of CAN with increased data capacity.

CAN High (CANH): The high-level signal in differential CAN signaling.

CAN Low (CANL): The low-level signal in differential CAN signaling.

CAN Message: A data packet transmitted over the CAN bus.

CAN Protocol: Set of rules governing communication over the CAN bus.

CAN Transceiver: Interface converting CAN signals between the controller and the physical bus.

CANalyzer: A tool used to analyze CAN network traffic.

CANoe: A software tool for CAN network simulation and testing.

CRC (Cyclic Redundancy Check): Error-detection method used in CAN messages.

Data Frame: The main type of message in CAN, carrying data.

Dominant State: The logical ‘0’ in CAN signaling.

Error Active: State indicating that a node can actively transmit error frames.

Error Frame: A special frame used to signal errors on the CAN bus.

Error Passive: State indicating a node can only passively detect errors.

Extended Frame Format: CAN frame format with a 29-bit identifier.

Identifier: A unique identifier for each CAN message.

ISO 11898: The international standard defining the CAN protocol.

J1939: A CAN-based protocol used in heavy-duty vehicles.

Loopback: A mode for testing where the CAN node transmits and receives its own data.

MCU (Microcontroller Unit): The processing unit in CAN nodes.

Message ID: A unique identifier within a CAN frame.

Node: A device or ECU connected to the CAN network.

Nominal Bit Rate: The standard speed of data transmission in a CAN network.

Non-Return to Zero (NRZ): Line coding method used in CAN signaling.

OSI Model: A framework for understanding CAN’s role in network communication.

Overload Frame: A frame used to delay the next data transmission.

Passive State: The logical ‘1’ in CAN signaling.

Payload: The data carried by a CAN frame.

Priority: Determines which CAN message gets transmitted first during arbitration.

Remote Frame: A CAN frame requesting data from another node.

RTR (Remote Transmission Request): A field indicating a Remote Frame.

SOF (Start of Frame): The first bit in a CAN message, indicating the start of transmission.

SRR (Substitute Remote Request): Used in Extended Frames, replacing RTR.

Stuff Bit: An extra bit added to prevent synchronization loss.

Synchronization: The process of aligning nodes' timing in a CAN network.

Termination Resistor: A resistor used to prevent signal reflection on the CAN bus.

Timestamp: A record of when a CAN message was transmitted or received.

Transceiver: Hardware that interfaces between the CAN controller and the physical bus.

Tx Buffer: Memory area where outgoing CAN messages are stored.

TxD (Transmit Data): Signal indicating data being sent on the CAN bus.

Unit: Another term for a node or ECU on the CAN network.

Voltage Levels: The electrical levels representing logical ‘0’ and ‘1’ in CAN signaling.

Wake-up Frame: A frame used to wake up nodes from sleep mode.

WUP (Wake-Up Pattern): Pattern used to signal nodes to exit sleep mode.