SURFACE VEHICLE RECOMMENDED PRACTICE	J2602™-1	OCT2021
	Issued 2004-08 Revised 2021-10 Superseding J2602-1 NOV2012	

(R) LIN Network for Vehicle Applications

RATIONALE

The LIN protocol is a low speed, low cost communication protocol that is capable of reducing wire counts to simple devices like switches and sensors. This document defines an implementation of the LIN protocol with the focus on enabling ASIC designs commonly found in switches and sensors. Rationale behind major updates since previous release is listed in Appendix F.

FOREWORD

The objective of this document is to define a level of standardization in the implementation of low speed vehicle serial data network communications using the local interconnect network (LIN) protocol.

The goal of this document is to define a serial data physical layer, data link layer and media design criteria to be installed in various automotive electronic control units (ECUs). This standard will allow ECU and tool manufacturers to satisfy the needs of multiple end users with minimum modifications to the basic design. This standard will benefit vehicle original equipment manufacturers (OEMs) by achieving lower ECU costs due to higher industry volumes of the basic design.

NOTE: Understanding of this document requires a working knowledge of the ISO 17987 specification package.

TABLE OF CONTENTS

1. SCOPE

This document covers the requirements for SAE implementations based on ISO 17987:2016. Requirements stated in this document will provide a minimum standard level of performance to which all compatible ECUs and media shall be designed. This will assure full serial data communication among all connected devices regardless of supplier.

The goal of SAE J2602-1 is to improve the interoperability and interchangeability of LIN devices within a network by adding additional requirements that are not present in ISO 17987:2016 (e.g., fault tolerant operation, network topology, etc.).

The intended audience includes, but is not limited to, ECU suppliers, LIN controller suppliers, LIN transceiver suppliers, component release engineers, and vehicle system engineers.

The term “master” has been replaced by “commander” and term “slave” with “responder” in the following sections.

1.1 Mission/Theme

This serial data link layer network is intended for use in applications where high data rate is not required and a lower data rate can achieve cost reductions in both the physical media components and in the microprocessor and/or dedicated logic devices (ASICs) which use the network.

1.2 Overview

LIN is a single wire, low cost, class A communication protocol. LIN is a commander-responder protocol, and utilizes the basic functionality of most universal asynchronous receiver transmitter (UART) or serial communication interface (SCI) devices as the protocol controllers in both commander and responder-nodes. To meet the target of “lower cost than either an OEM proprietary communications link or CAN link” for low speed data transfer requirements, a single wire transmission media based on the ISO 9141 specification was chosen. The protocol is implemented around a UART/SCI capability set, because the silicon footprint is small (lower cost). Many small microprocessors are equipped with either a UART or SCI interface (lower cost), and the software interface to these devices is relatively simple to implement (lower software cost). Finally, the relatively simplistic nature of the protocol controller (UART/SCI) and the nature of state-based operation, enable the creation of application specific integrated circuits (ASICs) to perform as input sensor gathering and actuator output controlling devices, in the vein of mechatronics.

All message traffic on the bus is initiated by the commander-node. Responder-nodes receive commands and respond to requests from the commander-node. Since the commander-node initiates all bus traffic, it follows that the responder-nodes cannot communicate unless requested by the commander-node. However, responder-nodes can generate a bus wakeup, if their inherent functionality requires this feature.

The LIN Consortium developed the set of LIN specifications. The Consortium was a group of automotive OEMs, semiconductor manufacturers, and communication software and tool developers. The LIN specification set is no longer available through the LIN Consortium. Instead, it has been converted to a series of ISO 17987 specifications.

The LIN specifications contain more than just a definition of the LIN protocol and physical layer. In addition, a work flow process, diagnostics and configuration methods, definition of an application program interface (API), file structures for a node capability file (NCF) and a LIN description file (LDF), and semantics are identified as required (mandatory in all implementations). However, since there is a great deal of flexibility in the protocol and physical layer, applicability of these specifications to SAE J2602-1 networks will be further specified in this document.

1.3 Relationship to the LIN Specifications

As described in the LIN specification package, the ISO 17987:2016 protocol specification suite consists of seven standards:

1.3.1 ISO 17987-1 - Road Vehicles - Local Interconnect Network (LIN) - Part 1 - General Information and Use Case Definition

This document gives an overview of the structure and the partitioning of ISO 17987 (all parts). In addition, it outlines the use case where the ISO 17987 (all parts) will be used. The terminology defined in this document is common for all LIN communication systems and is used throughout ISO 17987 (all parts). This document has been established in order to define the use cases for LIN.

1.3.2 ISO 17987-2 - Road Vehicles - Local Interconnect Network (LIN) - Part 2 - Transport Protocol and Network Layer Services

This document specifies a transport protocol and network layer services tailored to meet the requirements of LIN-based vehicle network systems on local interconnect networks. The protocol specifies an unconfirmed communication.

1.3.3 ISO 17987-3 - Road Vehicles - Local Interconnect Network (LIN) - Part 3 - Protocol Specification

This document specifies the LIN protocol including the signal management, frame transfer, schedule table handling, task behavior and status management and LIN master and slave node. It contains also OSI layer 5 properties according to ISO 14229-7 UDS on LIN-based node configuration and identification services (SID: B016 to B816) belonging to the core protocol specification.

1.3.4 ISO 17987-4 - Road Vehicles - Local Interconnect Network (LIN) - Part 4 - Electrical Physical Layer (EPL) Specification 12 V and 24 V

This document specifies the 12 V and 24 V electrical physical layers (EPL) of the LIN communications system. The electrical physical layer for LIN is designed with bit rates up to 20 kbit/s. The medium that is used is a single wire for each receiver and transmitter with reference to ground. This document includes the definition of electrical characteristics of the transmission itself and also the documentation of basic functionality for bus driver devices.

1.3.5 ISO /TR 17987-5 - Road Vehicles - Local Interconnect Network (LIN) - Part 5 - Application Programmers Interface (API)

This document specifies the LIN application programmers interface (API) and the node configuration and identification services. The node configuration and identification services are specified in the API and define how a slave node is configured and how a slave node uses the identification service.

1.3.6 ISO 17987-6 - Road Vehicles - Local Interconnect Network (LIN) - Part 6 - Protocol Conformance Test Specification

This document specifies the LIN protocol conformance test. This test verifies the conformance of LIN communication controllers with respect to ISO 17987-2 and ISO 17987-3. This document provides all necessary technical information to ensure that test results are identical even on different test systems, provided that the particular test suite and the test system are compliant to the content of this document.

1.3.7 ISO 17987-7 - Road Vehicles - Local Interconnect Network (LIN) - Part 7 - Electrical Physical Layer (EPL) Conformance Test Specification

This document specifies the 12 V and 24 V electrical physical layers (EPL) of the LIN communications system. The electrical physical layer for LIN is designed with bit rates up to 20 kbit/s. The medium that is used is a single wire for each receiver and transmitter with reference to ground. This document includes the definition of electrical characteristics of the transmission itself and also the documentation of basic functionality for bus driver devices.

The remainder of this document (SAE J2602-1) will directly reference these LIN specifications (publication date: 2016).

2. REFERENCES

2.1 Applicable Documents

The following publications form a part of this specification to the extent specified herein. Unless otherwise indicated, the latest issue of SAE publications shall apply.

2.1.1 SAE Publications

Available from SAE International, 400 Commonwealth Drive, Warrendale, PA 15096-0001, Tel: 877-606-7323 (inside USA and Canada) or +1 724-776-4970 (outside USA and Canada), www.sae.org.

SAE J551 (All Parts) Performance Levels and Methods of Measurement of Electromagnetic Compatibility for Vehicles and Devices (30 to 1000 MHz)

SAE J1113 (All Parts) Electromagnetic Sysceptibility Measurement Procedures for Vehicle Components (Except Aircraft)

SAE J1213-1 Glossary of Vehicle Networks for Multiplexing and Data Communications

SAE J1930 Electrical/Electronic Systems Diagnostic Terms, Definitions, Abbreviations, and Acronyms - Equivalent to ISO /TR 15031-2

SAE J2962-1 File Structures for a Node Capability File (NCF) and LIN Description File (LDF)

2.1.2 ISO Publications

Copies of these documents are available online at http://webstore.ansi.org/.

ISO 7498 Data Processing Systems, Open Systems Interconnection Standard Reference Mode

ISO 7637 Road Vehicles - Electrical Interference by Conduction and Coupling - Parts 1 and 2

ISO 9141 Road Vehicles - Diagnostic Systems - Requirements for Interchange of Digital Information

ISO 17987:2016 Road Vehicles - Local Interconnect Network (LIN) Protocol Specification Suite

3. DEFINITIONS

3.1 COMMAND FRAME

An unconditional frame published by the LIN commander-node and subscribed to by one or more responder-nodes.

3.2 DATA LINK LAYER

This provides for the reliable transfer of information across the physical layer. It includes the message structure, framing, and error control.

3.3 DOMINANT SIGNAL

The driven and low voltage state of the LIN bus. If multiple devices access the bus, this state dominates the recessive or non-driven state.

3.4 DORMANT STATE

The state in which the responder-node task state machine is waiting for reception of the break field/sync byte sequence.

3.5 COMMANDER-NODE

Responsible for initiating all message traffic. Refer to ISO 17987-3 for additional information (defined as “master node” in ISO 17987-3).

3.6 MEDIA

The physical entity that conveys the electrical (or equivalent means of communication) signal transmission among ECUs on the network.

3.7 PHYSICAL LAYER

This ISO 7498 subsection consists of the media, mechanical interconnections, and transceivers that provide the interconnection between all ECU nodes.

3.8 PROTOCOL

The formal set of conventions or rules for the exchange of information among the ECUs. This includes the specification of the signal frame administration, frame transfer, and physical layer.

3.9 PUBLISHER

A commander- or responder-node that is the source of the data transmitted onto the bus within a LIN message.

3.10 RADIATED EMISSIONS

The energy that radiates from the LIN physical layer.

3.11 RADIATED IMMUNITY

The level of susceptibility of physical layer components to communication errors in the presence of high energy electromagnetic fields.

3.12 RECESSIVE SIGNAL

The undriven and high voltage state of the LIN bus. If multiple devices access the bus, this state is overridden by the dominant state.

3.13 REQUEST FRAME

A frame with data published by the responder task in one and only one responder-node and only subscribed to by the responder task in the commander-node.

3.14 RESPONDER-NODE

A device that receives messages from the commander-node, or responds to messages initiated by the commander-node. Refer to ISO 17987-3 for additional information (defined as “slave node” in ISO 17987-3).

3.15 SUBSCRIBER

A commander or responder-node that receives the data within a LIN message.

3.16 HEXADECIMAL

Hexadecimal numbers in this document are represented using N_{16} or 0xN or $N format.

4. ACRONYMS, ABBREVIATIONS, AND SYMBOLS

API	Application Program Interface
ASIC	Application Specific Integrated Circuit
CAN	Controller Area Network
CTS	Component Technical Specification
DNN	Device Node Number
ECU	Electronic Control Unit
EMC	Electromagnetic Compatibility
ESD	Electrostatic Discharge
ID	Identifier
ISO	International Organization for Standardization
kbit/s	Thousands of Data Bits per Second
LDF	LIN Description File
LIN	Local Interconnect Network
LSB	Least Significant Byte
lsb	Least Significant Bit
MSB	Most Significant Byte
msb	Most Significant Bit
NAD	Node Address for Diagnostics
NCF	Node Capability File
OEM	Original Equipment Manufacturer
RE	Radiated Emissions
RI	Radiated Immunity
SCI	Serial Communication Interface
UART	Universal Asynchronous Receiver/Transmitter

5. LIN SYSTEM REQUIREMENTS

All ECU LIN interfaces shall conform to the ISO 17987:2016 specification package unless otherwise specified in this specification.

5.1 LIN Specification Package

The information contained in this section may or may not be representative of SAE J2602-1 based implementations. However, Section 3 of this LIN specification contains definitions and terms needed to comprehend the LIN protocol and SAE J2602-1.

5.2 SAE J2602-1 Serial Data Link Characteristics

- Commander/responder collision avoidance.
- Capable of interoperating with ISO 17987:2016, LIN 2.0, LIN 2.1[1], and LIN 2.2(A)[1] devices, as well as SAE J2602:2012 devices.
- Responder-node to responder-node communication is not supported and is highly discouraged.

[1] Excludes responder-node position detection service, SID5=$B5.

5.3 LIN Product Identification

The following requirements provide further clarification to the intent of the LIN supplier ID, function ID and variant ID assignment conditions.

5.3.1 Supplier ID

The LIN responder-node supplier shall use their own supplier ID that was provided to them.

Supplier IDs are requested using the ISO/CiA validated supplier ID process for use on all ISO and SAE products.

5.3.2 Clarification to Function ID Paragraph

Two products (ECUs) are identical in function and shall be assigned identical function identifiers if they exhibit all of the following characteristics:

- They exhibit identical functional behavior.
- They exhibit identical mandatory node capability file declarations.
- Both LIN communications and application functionality are configured identically by the IC supplier.

5.3.3 Clarification to Variant ID Paragraph

Two products (ECUs) are invariant and shall be assigned identical variant identifiers if they exhibit all of the following characteristics:

- They have identical operating range characteristics (voltage, temperature).
- They are constructed from the identical integrated circuit processes and manufacturing technology.

NOTE: Any change in the binary image loaded in a microprocessor-based responder-node implementation shall constitute a difference in the variant ID.

5.4 Signal Management

5.4.1 Signal Consistency

The requirement in ISO 17987-3 section 5.1.3 indicating that all "scalar signal writing or reading must be atomic" is not externally verifiable in all LIN implementations. Consequently, this requirement is a guideline.

NOTE: Since there is a very close coupling between the signals within the SAE J2602-1 status byte, implementers of SAE J2602-1 nodes are encouraged to make every effort to publish all signals within this byte with consistency, so that any/all error signaling between the responder-node and the commander-node can be interpreted properly.

5.4.2 Signal Repetition

The same signals may belong to multiple frames, e.g., SAE J2602-1 status byte within the same cluster.

5.4.3 Signal Encoding Types

Since only two signal encoding types are defined in the ISO 17987-3 section 5.1.2, and there is an expectation to employ other standard signal encoding types commonly used in other serial data implementations, the following recognized signal encoding types shall be mapped into the two LIN signal encoding types in accordance with the following table. Appendix B contains descriptions and examples of each of the signal encoding types listed.

Table 1 - Common signal encoding type mapping to LIN defined data entities

Signal Encoding Type	Slot Type	Mapped to LIN Data Type	Notes
ASCII	ASC	8-bit Scalar	Most Significant Bit reserved and set to “0”, ASCII codes 0 through 127.
Binary Coded Decimal	BCD	4 bit Scalar per BCD character	Must be on nibble boundaries.
Boolean	BLN	1 bit Scalar	
Enumerated	ENM	N bit Scalar, in increments of 1 bit (1-16 bits)	
Signed Floating Point	SFP	4 byte - Byte Array	As defined in ANSI/IEEE Std 754-1985.
Signed Numeric	SNM	8, 16, or 32 bit Byte Array	Unsigned with a negative offset number.
Unsigned Numeric	UNM	N bit Scalar, increments of 8 bits (1 byte)	Not to exceed 16 bits (2 bytes).

5.5 Detection of Errors by Commander-Node

The commander-node task state machine shall detect errors during the transmission of the break field/sync byte/protected ID sequence. If an error is detected (e.g., data mismatch or data not received), the commander-node shall cease transmission of the frame and shall start transmission of the next frame as dictated by the schedule table.

5.6 Responder-Node Task Error Detection

This section provides clarification of detecting and reporting errors.

The responder task state machine (either in a responder- or the commander-node) shall detect the following errors:

- LIN frame ID parity error: The receiver shall detect an ID parity error if the received ID parity (bits 6 and 7) does not match the ID parity calculated from the equations in ISO 17987-3 section 5.2.2.5.3 based on the received identifier (bits 0-5).

- Byte field framing error (i.e., invalid stop bit): The receiver shall detect a byte field framing error if the ninth bit after a valid start bit is dominant.

- Data error (i.e., data transmitted does not match data read, data transmitted is not received, fixed form data received is incorrect, transmitted stop bit is disturbed): A data error shall be detected when the bit or byte value that is received is different from the bit or byte value that is transmitted. A responder or commander-node that is transmitting a bit on the bus shall also monitor the bus.

- Sync field consistency error: The receiver shall detect a sync field consistency error when the data in the fixed form sync byte is received incorrectly, i.e., is not $55.

- Checksum Error: A checksum error shall be detected if the inverted modulo-256 sum over all received data bytes and the protected identifier (when using enhanced checksum) and the received checksum byte field does not result in $FF. Example of checksum calculation can be found in ISO 17987-3 Annex A.3.

-- Identification of the two errors listed below are optional and NOT required for SAE J2602-1 compliance --

- Header timeout (PID or SYNC byte field is absent or received out of header maximum time): When sync byte field following break field is not received or PID following sync byte field is not received within THEADER_MAX or break field is received before PID.

- Command frame response reception timeout: When the response part of the LIN frame is not received within TRESPONSE_MAX or break field is received before response is completed including checksum.

The detection of errors shall only be performed over frames designed for reception and transmission by the responder-node task. That is to say that the responder-node task shall only monitor the communication errors of the frames with protected identifies that it has been assigned.

Section 5.6.1 defines the method for reporting errors by the LIN responder-node to the LIN commander-node. OEM shall design the LIN responder-node to use this method and assign the signal in the SAE J2602 status byte as specified in 5.8.1.

5.6.1 Communication Error Reporting

Nodes that are compliant with the SAE J2602-1 2021 release of the standard shall implement only 1-bit communication error signal response and shall use V2 (2021 release) version of the status byte as defined in 5.8.1.

5.6.1.1 1-bit Communication Error Signal Response

The communication error signal shall be a 1-bit enumerated value with the following definition ($0 = No Comm Error, $1 = Comm Error Detected). This is equivalent to ERR2 used in the V1 (legacy) version of the status byte.

The LIN responder-node shall have the capability to detect errors listed in 5.6. If one or multiple of these errors are detected, the “communications error” bit shall be set to $1 (Comm Error Detected). This bit shall be considered a “sticky” bit, which remains set until it is successfully reported in the communication error signal to the commander-node. After successful transmission, all pending communication errors are cleared.

Since the responder-node will be physically unable to report the error in the frame that the error was detected, the responder-node shall transmit the error in the succeeding response frame. The communication error bit also serves the purpose of the 1-bit response error in ISO 17987-3 section 5.5.4.

5.6.1.2 3-bit Error Signal Response

Details of this reporting method can be found in the SAE J2602-1 NOV2012 release. This method of error reporting is used in networks that are compliant with the SAE J2602-1 NOV2012 release of the standard and utilizes V1 (legacy) version of the status byte.

5.7 Device Communication Error Detection and Report Management

In addition to the requirements listed in ISO 17987-3 section 5.5, the following error behavior shall be followed.

5.7.1 Responder-Node Behavior in the Presence of Errors When Transmitting

When a responder-node task state machine detects a byte field framing error, data error, or checksum error, the responder-node task state machine shall cease transmission prior to transmission of the next byte field, unless the error occurs during the transmission of the checksum byte, and return to the "dormant" state. The responder-node task shall also set the communication error signal as defined in 5.6.

5.7.2 Responder-Node Behavior in the Presence of Errors When Receiving

When a responder-node task state machine detects a sync field consistency, ID parity error, byte field framing error, data error, checksum error, header timeout, or response reception timeout the responder-node task state machine shall discard any data buffered from the current frame and return to the "dormant" state. The responder-node task shall also set the communication error signal as defined in 5.6.

5.8 Message Management

5.8.1 SAE J2602-1 Status Byte

This section defines SAE J2602 status byte, which encompasses both LIN protocol response error reporting and application specific information. The status byte shall be included in each response message published by the LIN responder-node. Status byte shall be transmitted as the first byte of every responder-node transmission, where the identifier is in the range of $00 through $3B (unconditional frame).

SAE J2602-1 Status Byte

Verson	Bit 7 (msb)	6	5	4	3	2	1	Bit 0 (lsb)
V1 (Legacy)	ERR2	ERR1	ERR0	APINFO4	APINFO3 to APIFNO0			
V2 (2021 release)	Comm Error	Reset	APINFO5 to APINFO0					

5.8.1.1 V1 - Legacy from November 2012 Release

Details for this verson of the status byte can be found in J2602-1 NOV 2012 release and is refered to in this document as V1 or legacy version of the status byte. It is used in networks that are compliant with the J2602-1 NOV 2012 release of the standard.
Nodes that are compliant with J2602-1 2021 release of the standard shall implement only V2 version of the status byte.

5.8.1.2 V2 - New for 2021 Release

Defines one bit for communication error, one bit for reset and remaining APINFO 0:5 are undefined.

Comm error shall be used to report errors specified in 5.6 and shall be implemented using details specified in 5.6.1.

Reset shall be used to indicate any reset including targetted reset. A responder-node shall set this state upon interruption and resumption of power, after a watchdog timeout, for targetted rest application as defined in section 5.9.6 or after receipt of a reset command.

Note that for those devices that require configuration and store the configuration information in volatile memory, this state indicates that the device is currently unconfigured and requires configuration. For those devices that use non-volatile memory for configuration information storage, then the state indicates configuration is required the first time the part is powered on and only indicates a reset from then on.

The application information field (APINFOx) is defined and implemented on a case-by-case basis by the OEM, dependent on the requirements of each responder-node application. These signals may be implemented as any combination of discrete or enumerated status bits or unsigned numerical values. The clearing mechanism used will also be defined by the OEM. The resulting application information field implementation shall be documented in the NCF for each responder-node.

Examples of signals that could be included in the status byte are as follows:

- Alive rolling counter.
- Cyclic redundancy check.
- Internal operation fault (APINFO4).
- Application resets.
- Device configuration required (APINFO3).

5.8.2 Unused Bits in the Data Field

The length of each frame in bytes, used for LIN normal communications (not diagnostics or configuration), shall be defined for each LIN network, on a frame-by-frame basis. As per ISO 17987-3 section 6.3.4.6, all bits not used or not defined shall be transmitted as recessive (logical 1) symbol. Padding of unused data bytes within LIN normal communications messages is not required. That is, normal communications messages may vary in length between 1 byte and 8 bytes.

5.8.3 Checksum

The enhanced checksum method shall be used for all protected identifiers except for those in the range of $3C to $3F.

Enhanced checksum is defined in ISO 17987-3 section 5.2.2.7.

5.8.4 Message Types

5.8.4.1 Availability of Unconditional Frames

The requirement that subscribers of an unconditional frame shall make it available to the application is not externally verifiable in all LIN implementations. Consequently, this requirement is a guideline.

5.8.4.2 Event Triggered Frames

Event Triggered frames shall not be utilized in SAE J2602-1 compliant LIN networks.

5.8.4.3 Identifier Assignment (SAE J2602-1 Requirement Resulting from Event Triggered Frame Anomaly)

One and only one responder-node shall be defined as the publisher to a single frame identifier.

5.8.4.4 Sporadic Frame

Messages of the sporadic frame type may be utilized within the SAE J2602-1 environment, if appropriate. The commander-node shall be the only publisher of sporadic frames.

5.9 Network Management

5.9.1 Message Transmission Time Tolerance

Each maximum message transmission time may be specified to be within the range of $T_{Frame\_Minimum}$ and $T_{Frame\_Maximum}$, providing both publisher and subscriber of the frame support a maximum message transmission time of less than $T_{Frame\_Maximum}$.

The minimum $T_{Frame\_Maximum}$ value to which a responder-node can respond shall be identified in its node capability file when the SAE J2602 language is used. Otherwise, in the event a value is not provided in the node capability file (either when an SAE J2602 language or an ISO 17987 language is used), the commander-node shall presume a value of $T_{Frame\_Maximum}$ of no more than 40%.

5.9.2 Device Configuration Using LIN Diagnostic Services

The following LIN diagnostic services are to be used for SAE J2602 LIN responder-node configuration only; they are not to be used for LIN responder-node diagnostics reporting since all relevant diagnostic information is included in the SAE J2602-1 status byte (if available). Furthermore, these configuration messages are only initiated by the LIN commander-node (ECU or test tool for bench testing).

No direct access to any LIN cluster shall be provided through the SAE J1962 connector.

Support of all remaining LIN diagnostics services and configuration as defined in ISO 17987-3 section 6 node configuration and identification is optional in SAE J2602-1 nodes. In other words, support of the services defined as mandatory in ISO 17987-3 section 6 node configuration and identification is optional and is not required for SAE J2602-1 compliance.

The following sections define the required SAE J2602-1 node configuration command/response messages.

5.9.3 General Configuration Requirements

5.9.3.1 Responder-Node Execution of Configuration

Within the limit of its capabilities, a responder-node shall immediately execute any configuration command upon receipt of the $3C message; it shall not wait for a $3D message.

5.9.3.2 Responder-Node Configuration Capabilities

A NCF file shall be included with an SAE J2602-1 device.

5.9.3.3 Commander-Node Configuration Message Pairing

A commander-node shall always transmit $3C/$3D coupled pairs for all capabilities described within this section. The commander-node shall never send multiple successive $3C messages without corresponding (interleaved) $3D messages. There shall be exactly one $3D response for each $3C command. The only exception to this is in the case of a broadcast $3C message, which shall have no corresponding $3D message.

5.9.4 NAD and Message ID Assignment

5.9.4.1 NAD Assignment

The NAD for an SAE J2602-1 device shall be in the range $60 to $6D, where the lower nibble of the SAE J2602-1 NAD contains a four bit device node number (DNN). An uninitialized node shall have a NAD of $6F. A NAD of $6E may be used; however, its message IDs must be assigned via $3C or $3E messages.

IC manufacturers may provide the ability to select the desired device node number, and hence the NAD, based on up to four memory bits, four external pins, or other methods developed by silicon suppliers. These bits or pins shall only impact the value of the lower nibble of the NAD. If the device has a selectable NAD, it shall default to $6F prior to initialization.

SAE J2602-1 NAD

Bit 7 (msb)	6	5	4	3	2	1	Bit 0 (lsb)
0	1	1	0	DNN3	DNN2	DNN1	DNN0

5.9.4.2 Message ID Assignment

Each device shall be assigned four message IDs based on the DNN, and, therefore, the NAD. If a device does not need that many messages, it shall use the messages with the lowest IDs. If a device requires more than four message IDs, it shall be assigned messages in powers of two; i.e., 4, 8, 16, 32. The system designer must ensure that multiple devices do not use the same message IDs.

As a consequence of having the message IDs associated with the DNN, after a power on reset, or reset command the protected identifiers are still marked as valid for devices with a DNN in the range $0-$D. Devices with a DNN of $E or $F will have the protected identifiers marked as invalid. The following table shows the boundaries for 4, 8, or 16 messages per node. If 32 message IDs are required, NAD $60 must be used for this node. SAE J2602-1 networks can combine nodes that use various numbers of message IDs.

Table 2 - NAD to message ID mapping

NAD	Message ID		NAD	Message ID		NAD	Message ID
$60	$00 $01 $02 $03		$60	$00 $01 $02 $03		$60	$00 $01 $02 $03
$61	$04 $05 $06 $07			$04 $05 $06 $07			$04 $05 $06 $07
$62	$08 $09 $0A $0B		$62	$08 $09 $0A $0B			$08 $09 $0A $0B
$63	$0C $0D $0E $0F			$0C $0D $0E $0F			$0C $0D $0E $0F
$64	$10 $11 $12 $13		$64	$10 $11 $12 $13		$64	$10 $11 $12 $13
$65	$14 $15 $16 $17			$14 $15 $16 $17			$14 $15 $16 $17
$66	$18 $19 $1A $1B		$66	$18 $19 $1A $1B			$18 $19 $1A $1B
$67	$1C $1D $1E $1F			$1C $1D $1E $1F			$1C $1D $1E $1F
$68	$20 $21 $22 $23		$68	$20 $21 $22 $23		$68	$20 $21 $22 $23
$69	$24 $25 $26 $27			$24 $25 $26 $27			$24 $25 $26 $27
$6A	$28 $29 $2A $2B		$6A	$28 $29 $2A $2B			$28 $29 $2A $2B
$6B	$2C $2D $2E $2F			$2C $2D $2E $2F			$2C $2D $2E $2F
$6C	$30 $31 $32 $33		$6C	$30 $31 $32 $33		$6C	$30 $31 $32 $33
$6D	$34 $35 $36 $37			$34 $35 $36 $37		$6D	$34 $35 $36 $37
$6E	No Message IDs defined		$6E	No Message IDs defined		$6E	No Message IDs defined
$6F	No Message IDs defined		$6F	No Message IDs defined		$6F	No Message IDs defined

NOTE: The message IDs listed in this table are the unprotected identifiers.

5.9.5 Configuration Messages

LIN protocol level and application level configuration of each responder-node may be accomplished using any combination of the following:

1. Using optional method defined in ISO 17987-3 section 6 node configuration and identification.

2. Using $3C messages with NADs in the user reserved range of $80-$FF.

3. Using $3E messages with any NAD.

When using $3C or $3E messages to configure the responder-nodes, the lower nibble of the NAD shall be the DNN to prevent conflicts between responder-nodes .

$3C NAD

Bit 7 (msb)	6	5	4	3	2	1	Bit 0 (lsb)
1	X	X	X	DNN3	DNN2	DNN1	DNN0

$3E NAD

Bit 7 (msb)	6	5	4	3	2	1	Bit 0 (lsb)
X	X	X	X	DNN3	DNN2	DNN1	DNN0

5.9.5.1 Response Message to Options 2 and 3 in 5.9.5

The $3D response message that follows each $3C and $3E command message shall return the SAE J2602-1 status byte (defined in section 5.8.1) in data byte 0 if available. The value of the other seven data bytes is implementation dependent and beyond the scope of this specification.

Tx by commander	*Tx by responder*	*Tx by responder*	*Tx by responder*	*Tx by responder*	*Tx by responder*	*Tx by responder*	*Tx by responder*	*Tx by responder*
LIN ID	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
$3D	J2602-1 Status Byte	XX	XX	XX	XX	XX	XX	XX

5.9.6 Targeted Reset

The targeted reset command (5.9.6.1) provides a mechanism for the commander-node to cause a re-initialization of a specific responder-node on the network, designated by the NAD in the command. Upon receipt of the targeted reset command, the responder-node shall cause an internal reset of operational variables to occur. Examples of operational variables include, but are not limited to, program counters, mode control variables, communications error counters, input source re-initializations, and output device re-initializations, but shall not alter any previously configured application level configuration information stored in non-volatile memory. This reset operation shall not cause the responder-node to destructively alter any LIN configuration data or addresses stored in non-volatile memory. Also, the reset operation shall not cause any configuration parameters in the LIN data link device (UART) to be altered. Upon conclusion of the reset operation, the responder-node shall remain configured, and shall assume a state consistent with a power-on initialization, with the exception of the LIN configuration information. The responder-node shall also retain knowledge that it has undergone a reset operation, such a reset shall be indicated in the SAE J2602-1 status byte.

The LIN responder-node shall be able to respond to a $3D request frame which starts 20 ms after the start of the break symbol of the $3C targeted reset command frame. A device may elect not to perform the requested reset command. The device shall respond to $3D request frame with the appropriate reset status set in the SAE J2602-1 status byte (5.8.1).

5.9.6.1 Targeted Reset Command

Tx by commander	Tx by commander	Tx by commander	Tx by commander	Tx by commander	Tx by commander	Tx by commander	Tx by commander	Tx by commander
LIN ID	Data0 NAD	Data1 PCI	Data2 SID	Data3	Data4	Data5	Data6	Data7
$3C	NAD	$01	$B5	$FF	$FF	$FF	$FF	$FF

5.9.6.2 Targeted Reset Response

Tx by commander	Tx by responder	Tx by responder	Tx by responder	Tx by responder	Tx by responder	Tx by responder	Tx by responder	Tx by responder
LIN ID	Data0 NAD	Data1 PCI	Data2 RSID	Data3	Data4	Data5	Data6	Data7
$3D	NAD	$06	$F5	Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	Variant ID

5.9.7 Application Strategies

The following is a library of application functions (strategies) that may be implemented by the OEM. Each function is optional and shall be tested as part of the LIN conformance test if it is implemented by the OEM.

The following sections define a list of optional application strategies and provides series of use cases demonstrating how these optional strategies can be implemented.

5.9.7.1 DNN Based Broadcast Messages

The following capability can be used by an OEM if all nodes participating on the network adhere to the following requirements. Otherwise, the associated message IDs shall only be used using traditional broadcast messaging.

There are four message IDs reserved for broadcast messages. The NAD that will use a specific broadcast message, and the data byte within the message that it will use is based on its DNN. The broadcast message IDs for a specific NAD are b11 100x and b11 101x, where x = DNN3. The relevant data byte number is equivalent to the three lsbs of the DNN. In the case where a node has more than four message IDs, i.e., 8 or 16, it shall also be assigned a proportionate number of broadcast message bytes.

5.9.7.1.1 DNN Based Broadcast Message Assignment for Four Messages per Node

LIN ID	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
$38	DNN = $0	DNN = $1	DNN = $2	DNN = $3	DNN = $4	DNN = $5	DNN = $6	DNN = $7
$39	DNN = $8	DNN = $9	DNN = $A	DNN = $B	DNN = $C	DNN = $D	XX	XX
$3A	DNN = $0	DNN = $1	DNN = $2	DNN = $3	DNN = $4	DNN = $5	DNN = $6	DNN = $7
$3B	DNN = $8	DNN = $9	DNN = $A	DNN = $B	DNN = $C	DNN = $D	XX	XX

5.9.7.1.2 DNN Based Broadcast Message Assignment for Eight Messages per Node

LIN ID	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
$38	DNN = $0	DNN = $0	DNN = $2	DNN = $2	DNN = $4	DNN = $4	DNN = $6	DNN = $6
$39	DNN = $8	DNN = $8	DNN = $A	DNN = $A	DNN = $C	DNN = $C	XX	XX
$3A	DNN = $0	DNN = $0	DNN = $2	DNN = $2	DNN = $4	DNN = $4	DNN = $6	DNN = $6
$3B	DNN = $8	DNN = $8	DNN = $A	DNN = $A	DNN = $C	DNN = $C	XX	XX

5.9.7.1.3 DNN Based Broadcast Message Assignment for 16 Messages per Node

LIN ID	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
$38	DNN = $0	DNN = $0	DNN = $0	DNN = $0	DNN = $4	DNN = $4	DNN = $4	DNN = $4
$39	DNN = $8	DNN = $8	DNN = $8	DNN = $8	DNN = $C	DNN = $D	XX	XX
$3A	DNN = $0	DNN = $0	DNN = $0	DNN = $0	DNN = $4	DNN = $4	DNN = $4	DNN = $4
$3B	DNN = $8	DNN = $8	DNN = $8	DNN = $8	DNN = $C	DNN = $D	XX	XX

5.9.7.2 Selectable NAD via PIN

All “device node number” DNN0, DNN1, DNN2, and DNN3 that are connected to a physical ECU HW IO PIN shall have a circuit connection between the HW IO PIN and a digital IO on the micro. The value of the DNN shall follow the state of the digital IO. By default, the digital IO shall be connected to an internal power source through a resistor. The logical value of the digital IO (or DNN) shall be set to 1. When the physical pin (or digital IO) is pulled to ground, the logical value of the digital IO (or DNN) shall switch from 1 to 0 for the duration of time when the pin is connected to ground.

NOTE: The use of the grounding a DNN allows an OEM to allocate multiple LIN responder-nodes (performing the same functions) to the same LIN Network. This technique allows an OEM to reduce part numbers, maintain a robust validatable interface and minimized efforts for repairs in service.

5.9.7.3 Calibration/Configuration Methods

This section defines three methods for calibrating/configuring responder-nodes in an SAE J2602-1 network using normal messaging. This requires one message ID for every 8 bytes of data. Additional details can be found in Appendix D.

- Method 1 - Large calibration/configuration method: This large calibration/configuration method can be used for any size calibration/configurations and always requires two message IDs. The large calibration/configuration requires additional complexity in the drivers.
- Method 2 - Large calibration/configuration method: This large calibration/configuration method can be used for calibration/configurations of up to 400 bytes using an interpretation of the LIN TP methodology defined in ISO 17987.
- Method 3 - Limited calibration/configuration method: This small calibration/configuration should only be used for calibrations/configurations of 160 bytes or less.

6. SAE J2602-1 API REQUIREMENTS

6.1 Commander-Node Configuration API

For commander-node implementations, the node configuration API (ISO /TR 17987-5 section 4.4) is mandatory if any node configuration commands will be utilized on the network. If node configuration is not used, these API calls are optional.

The node configuration API only applies to commander-node implementations.

6.2 Diagnostic Transport Layer API

For commander or responder-node implementations, the diagnostic transport layer API (ISO /TR 17987-5 section 4.5) is optional.

Diagnostic information is supplied via the SAE J2602-1 status byte.

6.3 Additional API Requirements

Please see SAE J2602-3 for additional details on commander and responder-node API requirements.

7. SAE J2602-1 BUS OPERATION

The physical layer is responsible for providing a method of transferring digital data symbols (1s and 0s) to the communication medium. The physical layer interface is a single wire, vehicle battery referenced bus, with low side voltage drive.

7.1 Communication Mode and Transmission Rate

7.1.1 Communication Mode

Slow communication mode transmission bit rate is 10.417 kbit/s, with a nominal bit time of 96 µs.

Fast communication mode transmission bit rate is 19.231 kbit/s, with a nominal bit time of 52 µs.

In both transmission modes, transmitters with controlled waveform fall and undershoot times should be used. Waveform rising edge control is recommended to assure that high frequency components are minimized at the beginning of the upward voltage slope. The remaining rise time occurs after the bus is inactive with drivers off and is determined by the RC time constant of the total bus load.

7.1.2 Auto Baud Rate Detect

This is an optional responder-node function and not required for SAE J2602-1 compliance.

Responder-nodes that support the auto baud rate detect function shall be capable of detecting the baud rate of the commander-node, 10.417 kbits/s or 19.231 kbits/s, when power is applied.

The responder-node shall detect slow or fast communication mode within the first two LIN headers transmitted by the commander-node after the responder-node initialization time, as defined by ISO 17987. Response handling for the first two headers is optional. Starting with the third LIN header, the responder-node shall be able to communicate at the detected baud rate as long as power is continuously connected.

7.2 Sleep/Wake Mode

7.2.1 Wake-up

The responder-node may continue to issue wake-up requests until the commander-node responds by transmitting a frame identifier. Following a minimum delay of 1.5 seconds and a maximum delay of 3.0 seconds (after the third wake-up attempt), the responder-node may initiate one additional sequence of three wake-up requests separated by a minimum of 150 ms and a maximum of 250 ms. The cycle is concluded following the sixth wake-up transmission. The minimum elapsed time for one complete wake-up cycle, as described, is 2.1 seconds.

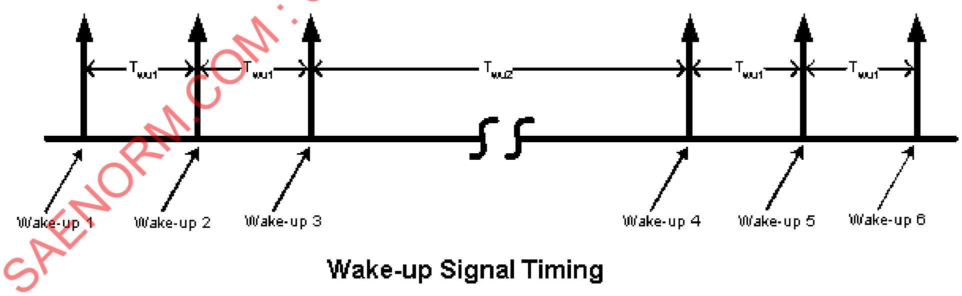

Figure 1 - Wake-up signal timing

Table 3 - Wake-up signal timing

Parameter	Minimum	Maximum
Wake-up Pulse	250 µs	5 ms
T_{wu1}	150 ms	250 ms
T_{wu2}	1.5 seconds	3 seconds

If no response (valid identifier transmission) is produced by the commander-node in response to the wake-up requests, the responder-node shall default to a network sleep state. In the event the responder-node detects another local wake-up input after the sleep default, it may again attempt to wake the commander-node through a new wake-up signal cycle. There is no limit to the number of wake-up cycles that may be attempted.

7.2.2 Go to Sleep

The commander-node must transmit an explicit “go to sleep” command to the network prior to ceasing to transmit.

7.2.3 Responder-Node Sleep

All responder-node(s) shall interpret a cessation of all message traffic for at least 4 seconds and no more than 10 seconds on a given network, without receiving an explicit “go to sleep command” as a failure condition of the commander-node or the physical layer. When this condition occurs, the responder-node(s) shall assume a default state that may include sleep and/or low power consumption state or application defined functionality.

7.3 LIN Controller Clock Tolerance

7.3.1 Commander-Node to Responder-Node Communication

Table 4 - Commander-responder-node communication clock tolerance

Device Type	Clock Tolerance	Notes
Commander	±0.5%	Initial Tolerance + Divide Error
Responder	±1.5%	From the nominal bit time with a fixed clock → Initial Tolerance + Divide Error when synchronized
Responder (autobauding)	±2.0%	From the Commander bit time → Initial Tolerance + Divide Error when synchronized

7.3.2 Responder-Node to Responder-Node Communication

This mode is currently not supported and is not recommended.

7.4 Bus Electrical Parameters

This section describes the bus electrical voltage level parameters required by devices that drive and receive signals on the LIN bus.

7.4.1 LIN Bus Signals and Loading Requirements

Table 5 - LIN bus signals and loading requirements

Parameter	Symbol	Min	Typ	Max	Units
Maximum ECU Battery Voltage Input for No Damage [5]	$V_{batt\ max\ ECU}$	-13		40	V
Maximum IC Battery Voltage Input for No Damage	$V_{batt\ max\ IC}$	0		34	V
High Voltage (Recessive) Input Threshold	V_{ih}	0.47 $V_{batt\ IC}$		0.6 $V_{batt\ IC}$	V
Output Low Voltage	V_{ol}	0.0		0.2 $V_{batt\ IC}$	V
Low Voltage (Dominant) Input Threshold	V_{il}	0.4 $V_{batt\ IC}$		0.53 $V_{batt\ IC}$	V
Input Threshold Hysteresis (V_{ih} - V_{il}) [4]	V_{HYS}	0.07 $V_{batt\ IC}$		0.175 $V_{batt\ IC}$	V
LIN bus to Ground ISO Lation Resistance	$V_{L\text{-}G\ ISO}$	500 K			Ω
Network Total Resistance	R_{tl}	537		1081	Ω
Device Bus Leakage Current V_{batt} Disconnected	$I_{leak\ batt}$	-30		30	⌠A
Device Bus Leakage Current Ground Disconnected, Bus Recessive	$I_{leak\ gnd}$	-100		100	µA
Device Bus Leakage Current Ground Disconnected, Bus Dominant	$I_{leak\ gnd}$	-1000		1000	µA
Responder-Node Capacitance [2]	$C_{responder}$	90	220	272	pF
Commander-Node Capacitance [2]	$C_{commander}$	90	680	2450	pF
Network Total Capacitance [3]	C_{tl}	926		9310	pF
Bus Wiring Capacitance	C_w			100	pF/m
Network Time Constant [1]	$\tau_{network}$	1.0		5.0	µs
Commander-Node Termination Resistance	R_M	900	1000	1100	Ω
Responder-Node Termination Resistance	R_S	20000	30000	60000	Ω
Total Network Length Connecting All ECU Nodes	Bus length	—		40 (see Table 8 for application limits)	meters
Number of System Nodes		2		16	

1. The normal mode network time constant ($\tau_{network}$) is the product of R_{tl} and C_{ltl}. The network time constant incorporates the bus wiring capacitance. The minimum value is selected to limit radiated emissions. The maximum value is selected to ensure proper communication under all communication modes and is the absolute maximum allowed under normal operating conditions. The system should be designed to have a time constant no larger than 5.3 µs under an error condition at a responder-node. This should be considered when determining the fusing for the vehicle. Not all combinations of R and C are possible. Only those combinations of R and C, and bus length, and PCB trace capacitance, etc., are possible that meet the specified network time constant.
2. The ECU capacitance includes the actual load capacitor as well as the PCB trace capacitance, connector capacitance, etc. The ECU capacitance is the room temperature capacitance and does not include temperature effects. The capacitance shall not change by more than 10% over the entire operating temperature range.
3. The network total capacitance includes the capacitors placed on the ECUs as well as the capacitance of the bus wires.
4. Input threshold hysteresis (V_{ih} - V_{il}) cannot be less than 0.0.
5. See 7.12 for details.

Table 6 - LIN bus loading requirements for normal battery supply operating range

Parameter	Symbol	Min	Max	Unit	Conditions
ECU Battery Voltage Input [1]	$V_{batt\ ECU}$	8	18	V	
IC Battery Voltage Input	$V_{batt\ IC}$	7	18	V	
Output High Voltage	V_{oh}	0.8 $V_{batt\ IC}$	$V_{batt\ IC}$	V	
Output Low Voltage	V_{ol}	0.0	0.2 $V_{batt\ IC}$	V	
Ground Offset Voltage [3]	$V_{g\ off}$	—	0.115 $V_{batt\ ECU}$	V	
Battery ECU Offset Voltage [3]	$V_{b\ off}$	—	0.115 $V_{batt\ ECU}$	V	
Supply Shift Difference	ΔV_{Shift}	- 0.07 $V_{batt\ IC}$	0.07 $V_{batt\ IC}$	V	
Duty Cycle 1 [2]	D_1	0.396	—		$V_{TH\_Rec(max)}$ = 0.744 x $V_{batt\ IC}$ $V_{TH\_Dom(max)}$ = 0.581 x $V_{batt\ IC}$ $V_{batt\ IC}$ = 7.0 V to 18 V; t_{BIT} = 52 µs
Duty Cycle 2 [2]	D_2	—	0.581		$V_{TH\_Rec(min)}$ = 0.422 x $V_{batt\ IC}$ $V_{TH\_Dom(min)}$ = 0.284 x $V_{batt\ IC}$ $V_{batt\ IC}$ = 7.6 V to 18 V; t_{BIT} = 52 µs
Duty Cycle 3 [2]	D_3	0.417	—		$V_{TH\_Rec(max)}$ = 0.778 x $V_{batt\ IC}$ $V_{TH\_Dom(max)}$ = 0.616 x $V_{batt\ IC}$ $V_{batt\ IC}$ = 7.0 V to 18 V; t_{BIT} = 96 µs
Duty Cycle 4 [2]	D_4	—	0.590		$V_{TH\_Rec(min)}$ = 0.389 x $V_{batt\ IC}$ $V_{TH\_Dom(min)}$ = 0.251 x $V_{batt\ IC}$ $V_{batt\ IC}$ = 7.6 V to 18 V; t_{BIT} = 96 µs

1. $V_{batt\ ECU}$ is measured at the ECU input power pins. All voltages are referenced to the local ECU ground.
2. Equations for converting from Duty Cycle parameters into timing parameters can be found in Appendix C.
3. Battery and ground offset voltages are referenced from the ECU supply voltage to the battery, not from the battery to the ECU.

Table 7 - LIN bus loading requirements for low battery supply operating range

Parameter	Symbol	Min	Max	Unit	Conditions
Low ECU Battery Voltage Input [1]	$V_{batt\ ECU\ low}$	6.5	8	V	
Low IC Battery Voltage Input	$V_{batt\ IC\ low}$	5.5	7	V	
Duty Cycle 1 at Low Battery Supply [2]	D_{1_low}	0.396	—		$V_{TH\_Rec(max)}$ = 0.665 x $V_{batt\ IC\_low}$ $V_{TH\_Dom(max)}$ = 0.499 x $V_{batt\ IC\_low}$ $V_{batt\ IC\ low}$ = 5.5 V to 7 V; t_{BIT} = 52µs
Duty Cycle 2 at Low Battery Supply [2]	D_{2_low}	—	0.581		$V_{TH\_Rec(min)}$ = 0.496 x $V_{batt\ IC\_low}$ $V_{TH\_Dom(min)}$ = 0.361 x $V_{batt\ IC\_low}$ $V_{batt\ IC\ low}$ = 6.1 V to 7.6 V; t_{BIT} = 52µs
Duty Cycle 3 at Low Battery Supply [2]	D_{3_low}	0.417	—		$V_{TH\_Rec(max)}$ = 0.665 x $V_{batt\ IC\_low}$ $V_{TH\_Dom(max)}$ = 0.499 x $V_{batt\ IC\_low}$ $V_{batt\ IC\ low}$ = 5.5 V to 7 V; t_{BIT} = 96µs
Duty Cycle 4 at Low Battery Supply [2]	D_{4_low}	—	0.590		$V_{TH\_Rec(min)}$ = 0.496 x $V_{batt\ IC\_low}$ $V_{TH\_Dom(min)}$ = 0.361 x $V_{batt\ IC\_low}$ $V_{batt\ IC\ low}$ = 6.1 V to 7.6 V; t_{BIT} = 96µs

1. $V_{batt\ ECU}$ is measured at the ECU input power pins. All voltages are referenced to the local ECU ground.
2. Equations for converting from duty cycle parameters into timing parameters can be found in Appendix C.

7.5 LIN Data Link (UART) Requirements

Any device (e.g., UART, SCI, software, etc.) chosen to implement an SAE J2602-1 LIN data link interface shall meet all requirements in this section.

7.5.1 Sample Point

The device shall meet the Sample Point requirements as defined by ISO 17987-4 section 5.2.3.

7.5.2 Synchronization

- The device shall be able to synchronize within 1/16 of a bit time.
- The device shall only synchronize on recessive to dominant edges.
- The device shall always synchronize on the recessive to dominant edge of the start bit.

7.5.3 Transmit Message Buffering

Double buffering shall not be used for transmit messages to ensure that the requirement for error detection in 5.6 is not violated.

7.6 LIN ECU Requirements

7.6.1 ECU Circuit Requirements

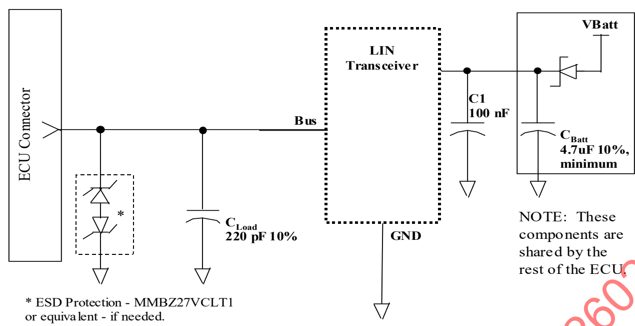

Figure 2 - Typical LIN responder-node bus interface

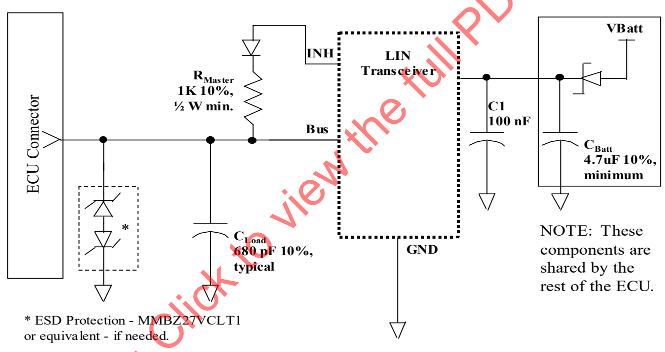

Figure 3 - Typical LIN commander-node bus interface

7.6.1.1 Commander-Node Resistor

The commander-node resistor shall be as specified in Table 5 and Figure 3. It shall have a minimum power rating of 0.5 W, or 0.36 W at the maximum specified operating temperature.

This assumes that the transceiver limits the current to the load pin to a maximum of 20 mA. In the event the transceiver will provide more current to the resistor, the maximum power dissipation shall be the maximum of 0.78 W and that calculated by the following equation: P (watts) = I^2 * R, where I is the maximum current in amps and R is 900 Ω.

7.6.1.2 Commander-Node Pull-up Reverse Blocking Diode

The maximum voltage drop in the commander-node pull-up reverse blocking diode is 1.0 V at the maximum current; the current limited by the transceiver or 30 mA, whichever is smaller.

The minimum power dissipation of the diode is determined by the current through the diode. It shall be at least 20 mW if the transceiver limits the current to the load pin to a maximum of 20 mA; otherwise, it shall be 30 mW.

7.6.1.3 Commander-Node Capacitance

The commander-node load capacitor shall be as specified in Table 5 and Figure 3 with voltage rating appropriate to a maximum loaded network under worst case environmental and electrical conditions.

7.6.1.4 Slave Responder-Node Capacitance

The responder-node load capacitor shall be as specified in Table 5 with voltage rating appropriate to a maximum loaded network under worst-case environmental and electrical conditions.

7.6.1.5 ESD Transient Suppressor

If necessary, a circuit element such as a transorb (back-to-back zener) or a varistor device may be added to the network in one or more places to provide ESD protection. However, when these devices are used they may add capacitance or introduce voltage and/or temperature variability to the network time constant. When such devices are used, the device load capacitor shall be reduced by an amount equivalent to the capacitance of the ESD transient suppressor. See Figures 2 and 3.

7.6.1.6 LIN Bus Protection Device

If any series protection device is included on the LIN bus, e.g., inductor, ferrite bead, resistor, etc., the DC series resistance of this device shall be less than 3.0 Ω.

7.6.2 Board Layout Requirements

- All grounding of the LIN transceiver and the filter capacitors shall be made to ECU signal ground.
- C1 and C_{LOAD} shall be monolithic ceramic chip capacitors. (Ceramic chip capacitors have low ESR and high self-resonant frequencies.)
- A ground plane is required under the transceiver chip on the same side of the board as the component.
- Transceiver shall be located as close to edge connector as possible. Other ICs are not permitted between edge connector and the transceiver.
- The LIN bus circuit between the edge connector and transceiver shall be as short as possible. Guard tracks are required for all LIN bus and Tx and Rx circuits.
- All guard tracks shall be at least 0.5 mm wide and grounded at least every 10 mm. No signals shall be routed between the guard track and the LIN bus trace.

7.7 Network Topology

7.7.1 Loss of ECU Ground

The loss of ground by any single responder-node, with or without an accompanying loss of V_{batt}, shall not cause any bus voltage offset that will disable normal communications (see $I_{leak\ gnd}$ in Table 5).

7.7.2 Loss of ECU Battery

The loss of battery by any single responder-node, with or without an accompanying loss of ground, shall not cause any bus voltage offset that will disable normal communications (see $I_{leak\ batt}$ in Table 5).

7.7.3 Bus Electrical Load Distribution

Each responder-node shall contain a responder-node capacitance load and a responder-node termination resistive load.

The total network equivalent minimum resistance (R_{tl}) and maximum capacitance (C_{tl}) shall comply with the totals specified in Table 5.

7.7.4 Bus Wiring Topology Configurations

The data link physical medium wiring mechanization can be implemented in any of the following ways:

- Vehicles may be wired in a ring, a star, or a combination of both. Note that ECUs that are intended for use across multiple platforms may have two connector pins as shown below to allow a ring connection.
- The pins of the ECU when applied in a ring, shall be adjacent and in the same connector, shorted together as close to the connector as possible, and share EMC and/or loading components.
- If in a star configuration, the ECU requires only one pin for LIN.

The topology of the LIN bus shall be determined for each vehicle platform based on the vehicle's fault tolerance, serviceability, and bus length requirements. A second bus wire connector terminal for the LIN circuit at each ECU allows for implementation of a ring configuration, although it is not required that both terminals be used. A ring, star, or combination of ring and star configuration is acceptable as long as all other LIN wiring requirements are met. Illustrations of the LIN topologies are shown in Figures 4, 5, 6, and 7.

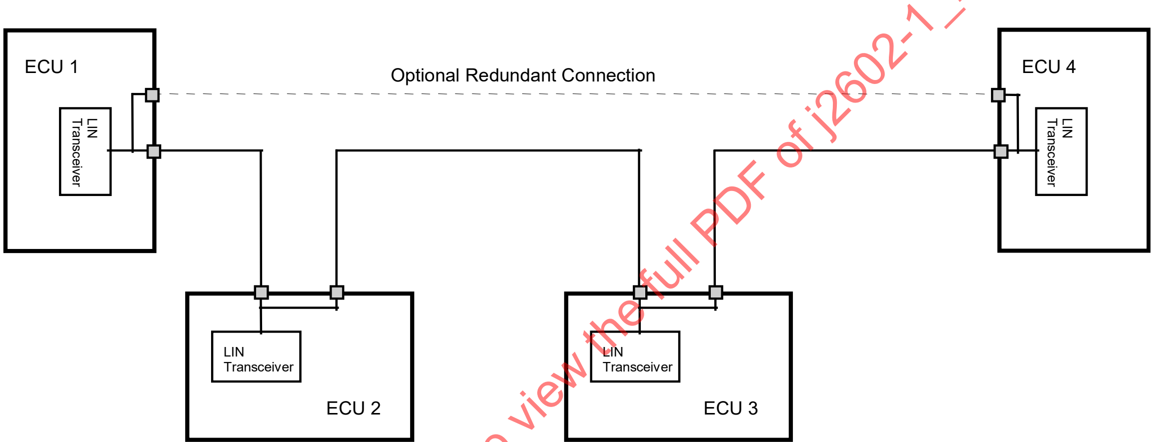

Figure 4 - LIN ring topology

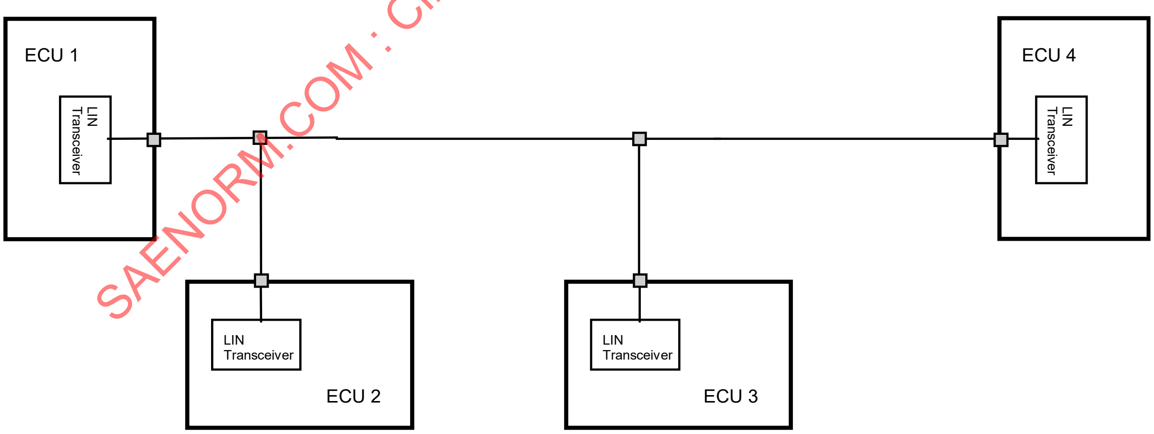

Figure 5 - LIN linear topology

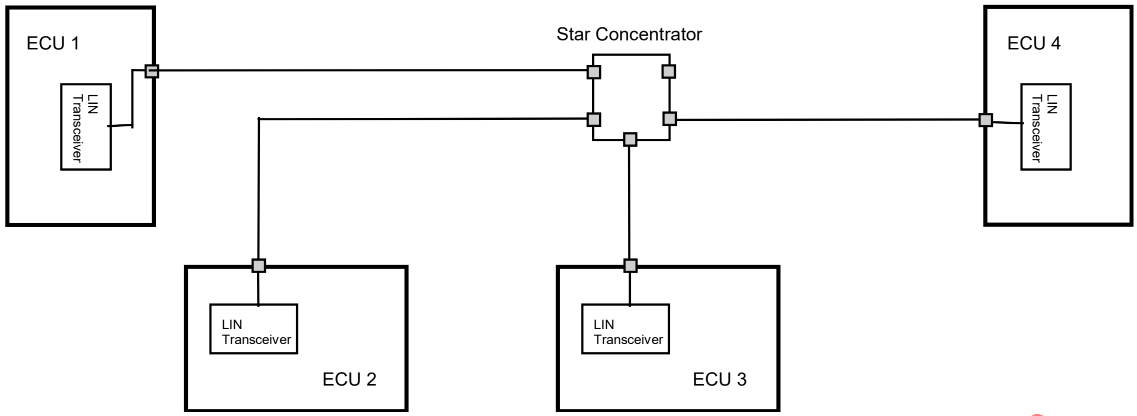

Figure 6 - LIN star topology

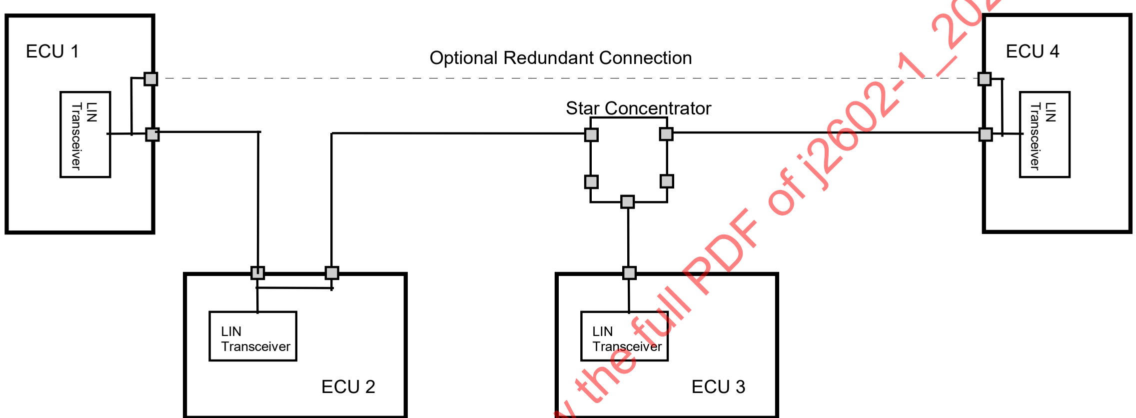

Figure 7 - LIN combination ring and star topology

7.7.5 Bus Wiring Constraints

The vehicle network wiring and ECU system shall meet the following constraints:

- The total bus wiring capacitance shall not cause the network time constant to be exceeded (see Table 5). The maximum bus length allowed is determined by the number of nodes in the network system and their R-C characteristics.

- There shall be no more than 40 m between any two network system ECU nodes.

The following table uses the maximum network time constant to determine the maximum wire length that can be used to connect a given number of ECUs. This maximum wire length depends on the commander-node capacitance and the number of responder-nodes. In calculating these numbers it was assumed that all resistors are at the maximum allowed values and capacitors (10%) are at their maximum tolerance, that the capacitance of the wire was 100 pF/m, that the responder-nodes all have nominal capacitances of 220 pF and that an additional 30 pF of capacitance are added due to PCB traces and connectors.

Table 8 shows the maximum wire length allowed on the vehicle based on the number of nodes on the vehicle and their characteristics. The legend in Table 8 gives the capacitance in the commander-node.

The maximum vehicle wire length is calculated using Equation 1:

$$\text{Wire} = [\tau_{Network} / (R_s / \text{\# responder-nodes} \parallel R_m) - C_m - C_s * \text{\# responder-nodes}] / \text{wire}(C/m) \quad \text{(Eq. 1)}$$

Table 8 - #Nodes/network resistance/commander-node capacitance versus max wire length

	$C_{commander\ pF\ (max)}$		
#responder nodes	272	778	2450
1	40.85 m	35.79 m	19.07 m
2	38.96 m	33.90 m	17.18 m
3	37.07 m	32.01 m	15.29 m
4	35.19 m	30.13 m	13.41 m
5	33.30 m	28.24 m	11.52 m
6	31.41 m	26.35 m	9.63 m
7	29.53 m	24.47 m	7.75 m
8	27.64 m	22.58 m	5.86 m
9	25.75 m	20.69 m	3.97 m
10	23.87 m	18.81 m	2.09 m
11	21.98 m	16.92 m	0.20 m
12	20.09 m	15.03 m	0 m
13	18.21 m	13.15 m	0 m
14	16.32 m	11.26 m	0 m
15	14.43 m	9.37 m	0 m

7.7.6 Bus Wiring Practices to Improve EMC Performance

- Avoid routing bus wire signals with noisy (e.g., injector drivers) and sensitive (e.g., low signal level sensors, antenna feeds) circuits.
- Precaution shall be taken when routing signals near antennae or antenna amplifiers to prevent inducing noise into these circuits. A shielded wire may be needed near an active antenna.
- Avoid wire loops by locating bus wires close to the vehicle's metal ground plane or route a ground wire with the bus wire.

7.7.7 Bus Wiring Harness and ECU Connectors

Connectors shall have less than 50 mΩ resistance over total vehicle life.

7.8 ESD Immunity

The ECU LIN Bus I/O pin shall withstand the following electrostatic discharges without any damage to the ECU when subjected to the SAE J2962-1:2019 Section 6. The particular vehicle manufacturer's ECU component technical specification shall state the criticality level of the device. If the component technical specification does not specify the ESD level, use the requirements in Table 9.

Table 9 - ESD immunity requirements

ECU Condition	Contact	Air (Non-Contact)
Unpowered	±6 KV	±8 KV

7.9 EMC Testing Requirements

The LIN physical layer, when incorporated into an ECU design, shall function as specified in the ECUs intended electromagnetic environment. Additionally, the electromagnetic emissions produced during LIN related operations shall not interfere with the normal operation of other ECUs or subsystems.

The transceiver selected shall be tested as specified in SAE J2962-1.

7.10 Fault Tolerant Modes

The Network shall meet the requirements as defined per the following failure modes:

- ECU power loss: ECUs shall not interfere with normal communication among the remaining bus ECUs during a loss of power (or low voltage) condition. Upon return of power, normal operation shall resume without any operator intervention within a time determined by the vehicle manufacturer.

- Bus wiring short to ground: Network data communications may be interrupted but there shall be no damage to any ECU when the bus is shorted to ground. A network impedance of less than 50 Ω between the bus and ground shall be considered a short to ground and continued communications are not guaranteed or required. Upon removal of the fault, normal operation shall resume without any operator intervention within a time determined by the vehicle manufacturer.

- Bus wiring short to battery: Network data communications may be interrupted but there shall be no damage to any device when the bus is shorted to positive battery less than 26.5 V (V_{batt} < 26.5 V). A network impedance of less than 50 Ω between the bus and battery shall be considered a short and continued communications are not guaranteed or required. Upon removal of the fault, normal operation shall resume without any operator intervention within a time determined by the vehicle manufacturer.

- A short or open in any single wiring circuit of an ECU, except for power, ground, or serial data, shall not preclude the ability to communicate with that ECU for diagnostic purposes.

7.11 Ground Offset Voltage

Ground offset voltage limits at the ECU as specified in Table 5 must be maintained over the entire supply voltage range.

7.12 Operating Battery Power Voltage Range

7.12.1 Normal Battery Voltage Power Operation

Unless otherwise specified by the Component Technical Specification, ECUs shall be capable of meeting all requirements specified in this document when the $V_{batt\ ECU}$ voltage as measured at the ECU power input pin is within the range of 8 to 18 VDC. The ECU shall provide $V_{batt\ IC}$ to the bus transceiver within the range of 7 to 18 V when $V_{batt\ ECU}$ is in the range of 8 to 18 V.

7.12.2 Battery Power Over-Voltage Operation

Communication in the 18 < $V_{batt\ ECU}$ < 26.5 V range is not guaranteed. If communication is required in this range, the ECU must limit the voltage to the transceiver to 18 V. If the voltage to the transceiver is not limited to 18 V, communication is not guaranteed with $V_{batt\ ECU}$ above 18 V. If deterministic behavior of the network is desired in this range, the commander-node shall determine the state of the bus, i.e., by forcing the bus to go to sleep, communicating with specific devices, etc.

- For 18 < $V_{batt\ ECU}$ < 26.5 V, the bus may operate in either the normal or the passive mode. Nodes shall only enter the passive mode in this voltage range in order to protect themselves, they shall not shut down due to voltage alone. In the passive mode, the bus shall be recessive (not be pulled or driven to ground) and RxD shall be in the high state.

- Recessive state transceiver leakage current limits shall be maintained over this range.

- ECUs shall not sustain permanent damage when subjected to $V_{batt\ ECU}$ up to 26.5 V.

- Vbatt ECU transients of greater than 40 V and/or duration greater than 10 ms must be clamped to avoid damage to the transceiver.

- The transceiver shall not transmit a dominant state on the bus when the TxD is in a recessive state.

7.12.3 Low Battery Voltage Operation

Unless otherwise specified by the component technical specification, ECUs shall be capable of meeting all low battery operation requirements specified in this document when the $V_{batt\ ECU}$ voltage as measured at the ECU power input pin is within the range of 6.5 to 8 VDC. The ECU shall provide $V_{batt\ IC}$ to the bus transceiver within the range of 5.5 to 8 V when $V_{batt\ ECU}$ is in the range of 6.5 to 8 V.

7.12.4 AutoStart Cranking Voltage Operation

The LIN node shall continuously communicate when tested with the stop/start crank waveforms as provided by the OEM.

7.12.5 Initial Cranking Voltage Operation

The LIN node shall not reset when tested with the crank pulse waveforms as provided by the OEM.

7.12.6 Very Low Battery Voltage Operation

For $0 < V_{batt\ ECU} < 6.5$ V, the bus may operate in either the normal or the passive mode. In the passive mode, the bus shall be recessive (not be pulled or driven to ground) and RxD shall be in the high state.

7.12.7 Battery Offset Voltage

The battery offset voltage limits, between the battery input pins of any ECU, specified in Tables 6 and 7 must be maintained over the entire range of $6.5 < V_{batt\ ECU} < 26.5$ V.

7.12.8 Reverse Battery Blocking Diode

The reverse battery blocking diode voltage drop between the ECUs $V_{batt\ ECU}$ input pin and the transceiver's $V_{batt\ IC}$ input pin shall be $V_{diode} \leq 1.0$ V.

7.12.9 LIN Transceiver Supply Voltage

When the ECU is connected to the vehicle battery voltage directly, the voltage supplied to the LIN transceiver shall be $\geq(V_{batt\ ECU}$ - 1.0V). This shall include all sources of voltage drop between the controller connector pins and the transceiver supply input pin.

When the ECU is connected to a switched voltage source, the voltage supplied to the LIN transceiver shall be $\geq(V_{batt\ ECU}$ - (1.0 V - V_{drop_switch})). V_{drop_switch} is the voltage drop in another ECU that provides power to this ECU (diode) or is the voltage drop through the device that switches the power to this ECU (FET, relay, etc.).

7.13 Environmental Requirements

ECU environmental requirements shall be specified in the individual ECU component technical specifications that call out this specification. In general, communications devices, which are installed in these ECUs, shall operate in the -40 to 125 °C temperature range.

7.13.1 Transmit Operating Conditions

7.13.1.1 Commander-Node

Commander-node shall be able to transmit continuously when 90% of the bits transmitted are dominant to accommodate transmitting $00 data consecutively.

7.13.1.2 Responder-Node

A responder-node shall be able to transmit continuously when it is transmitting a dominant for 70% of the message time.

7.13.1.2.1 Stand-Alone Transceivers

Stand-alone transceivers shall also be able to drive the bus dominant for 5 ms, the maximum wake-up pulse time.

7.13.1.2.2 Integrated Transceivers

Integrated transceivers shall be able to drive the bus dominant for the time requested by the application, no more than 5 ms.

8. VALIDATION

ECUs shall be required to pass the network functional performance validation tests as specified by the vehicle manufacturer. Environmental and other requirements shall be specified by the vehicle manufacturer's component technical specification that references this document.

ECU suppliers shall validate that after the vehicle manufacturer's salt fog and biased humidity accelerated life tests, the resistance between LIN pin and V_{batt} pin shall be greater than 20 KΩ for responder-nodes and 900 Ω for commander-nodes.

ECU suppliers shall validate that after the vehicle manufacturer's salt fog and biased humidity accelerated life tests, the resistance between LIN pin and ground pin shall be greater than 500 KΩ.

9. NOTES

9.1 Revision Indicator

A change bar (l) located in the left margin is for the convenience of the user in locating areas where technical revisions, not editorial changes, have been made to the previous issue of this document. An (R) symbol to the left of the document title indicates a complete revision of the document, including technical revisions. Change bars and (R) are not used in original publications, nor in documents that contain editorial changes only.

PREPARED BY THE VEHICLE ARCHITECTURE FOR DATA COMMUNICATIONS STANDARDS COMMITTEE

APPENDIX A - LIN DEVICE SUPPLIERS

There are many vendors providing LIN devices. They are not listed here as they are constantly changing.

APPENDIX B - SIGNAL ENCODING TYPES

B.1 ASCII (ASC)

ASCII data uses a one byte code to represent a text character. ASCII data is most often used where the consumer of the data is a display device which recognizes ASCII characters and can therefore display the data without further conversion. The least significant 7 bits represent the standard ASCII codes from 0 to 127. The most significant bit is reserved at this time but may be assigned a special function in the future. All ASCII signals shall have a length in bits which is a multiple of 8.

Example: The ASCII code for “A” is $41.

B.2 BOOLEAN (BLN)

Boolean signals are used to encode data that contains binary parameters, such as status bits or flags. A Boolean signal is always a single bit and is always encoded with the values True (=1) and False (=0). Boolean signals are named such that the name implies the information being transferred when the value of the variable is True.

Examples: For a signal named Windshield Wiper Switch Active: $0 = False, $1 = True.

B.3 ENUMERATED (ENM)

Enumerated signals are used for data that can take one of several states such as “day of week” or “wiper mode.” Enumerated definitions contain a field for describing states within the signal. There are up to 2^n possible states, where n is the number of bits reserved for the signal. The state definitions should be created such that all of the states are mutually exclusive. Note that all states do not need to be defined.

Example: A signal representing day of the week may be defined as a three bit scalar with the following decoding

b000 = Invalid Day of the Week
b001 = Sunday
b010 = Monday
b011 = Tuesday
b100 = Wednesday
b101 = Thursday
b110 = Friday
b111 = Saturday

B.4 NUMERIC SIGNALS

The units preferred for numeric data type signals are SI units (meters, km/h, °C, etc.). However, use of dimensionless quantities (% open, etc.) is also allowed.

A numeric signal must encode values which are continuous throughout the defined range of the signal. Use of specific numeric values to communicate state information is not allowed.

B.4.1 Binary Coded Decimal (BCD)

Binary coded decimal (BCD) encoding is used when it is desirable to report decimal data in a nibble, and is often used where the data consumer is a display device. A BCD encoded signal shall be 4 bits long. Valid BCD data are the hex characters 0-9 with the following encoding:

$0 = 0 decimal
$1 = 1 decimal
$2 = 2 decimal
$3 = 3 decimal
$4 = 4 decimal
$5 = 5 decimal
$6 = 6 decimal
$7 = 7 decimal
$8 = 8 decimal
$9 = 9 decimal
$A - $F = invalid

For example, the hex byte $25 would be interpreted as 37 decimal. As two BCD characters, the value is interpreted as 25 decimal.

B.4.2 Signed Numeric (SNM)

Signed numeric are represented as unsigned numeric with a negative offset; see Equation B1.

$$E=N^{*}R + (\text{negative})\text{Offset} \quad \text{(Eq. B1)}$$

B.4.3 Unsigned Numeric (UNM)

Unsigned numeric (UNM) encoding is used for signals which are continuous in range, such as temperature, speed, or percent. The signal may or may not have an offset. Unsigned numeric signals can also be used for sequential data such as month (1-12) or day of month (1-31). Each signal definition contains a field for resolution per bit (R), data length in bits (L), offset, minimum and maximum value, and units. Non-symmetrical numbers can be represented by a negative offset. From these characteristics, the transfer function between computer units (N) in decimal, and engineering units (E) of the data may be derived as follows:

$$E=N^{*}R + \text{Offset} \quad \text{(Eq. B2)}$$

The maximum value of the signal is:

$$\text{Max} = R^{*}(2^{L} -1)+ \text{Offset} \quad \text{(Eq. B3)}$$

The minimum value of the signal is:

$$\text{Min} = \text{Offset} \quad \text{(Eq. B4)}$$

B.4.4 Signed Floating Point [Scientific Notation] (SFP)

Signed floating point (SFP) is used to encode signals requiring representation in floating point arithmetic, and always includes a leading sign character. The format exactly follows the ANSI/IEEE Standard (Std 754-1985) single format. Please note that this signal type consumes 4 data bytes, where the data byte boundaries of the transmitted frame do not align with the boundaries of this format. The floating point signal is sent as a 32 bit (4 byte) variable. The bit order is shown below:

	1 bit	8 bits	23 bits	
. . .	Sign Bit	Exponent	Fractional Part	. . .
	MSB		LSB	

APPENDIX C - BIT TIMING CALCULATIONS

C.1 RELATION BETWEEN PROPAGATION DELAY AND DUTY CYCLE

Figure C1 shows the relation between the transmitter propagation delay and the duty cycle. The duty cycles in 7.4 can be calculated as follows:

$$D_{1(low)/3} = \frac{t_{BIT} - t_{REC(MAX)} + t_{DOM(MIN)}}{2 \cdot t_{BIT}} \quad \text{(Eq. C1)}$$

$$D_{2(low)/4} = \frac{t_{BIT} - t_{REC(MIN)} + t_{DOM(MAX)}}{2 \cdot t_{BIT}} \quad \text{(Eq. C2)}$$

Equation C1 refers to D_1, D_{1_low}, and D_3; Equation C2 refers to D_2, D_{2_low}, and D_4.

It is important to notice that the timings of t_{REC} and t_{DOM} are a combination of both transmitter propagation delay and slope time/slew rate.

The requirements for transmitter propagation delay can be derived out of the duty cycle Equations C1 and C2:

$$t_{REC(max)} - t_{DOM(min)} \leq t_{BIT} - 2 \times t_{BIT} \times D_{1(low)/3} \ @F_{TOL} = 2\% \quad \text{(Eq. C3)}$$

$$t_{DOM(max)} - t_{REC(min)} \leq 2 \times t_{BIT} \times D_{2(low)/4} - t_{BIT} \ @F_{TOL} = 2\% \quad \text{(Eq. C4)}$$

It can be seen that the absolute transmitter propagation time is irrelevant. Only the difference between t_{REC} and t_{DOM} is relevant. As long as both Equations C3 and C4 are fulfilled, the LIN transmission is reliable.

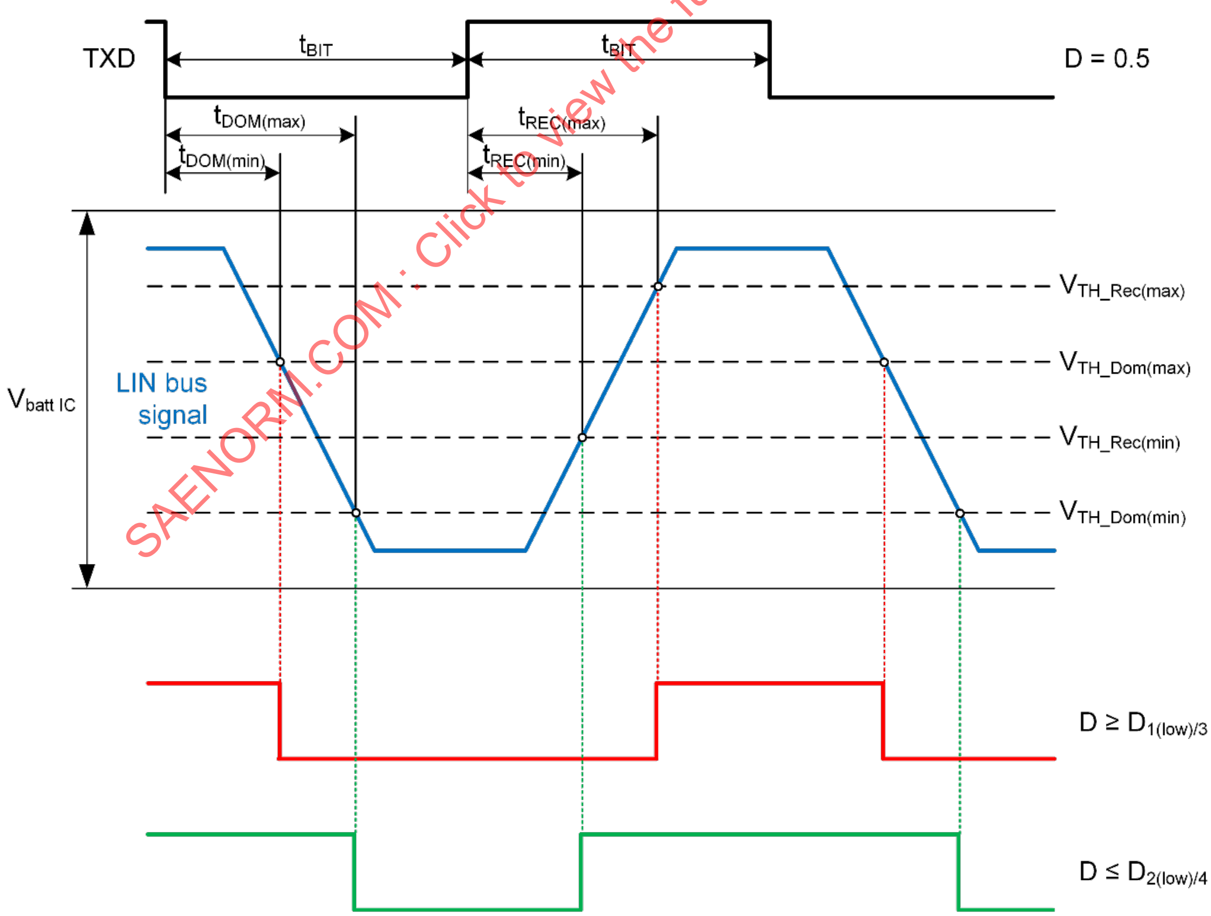

Figure C1 - Relation between propagation delay and duty cycle

NOTE: The threshold values shown are for the receiving node relative to the transmitting node's $V_{batt\ IC}$.

The transmitter propagation delay timings for the duty cycle parameter are listed in Table C1.

Table C1 - Transmitter propagation delay timings

Parameter	Symbol	Max	Unit	Reference
$t_{REC(MAX)\_D1}$ - $t_{DOM(MIN)\_D1}$	$T_{r\text{-}d\ max\_D1}$	10.8	µs	Duty Cycle 1
$t_{DOM(MAX)\_D2}$ - $t_{REC(MIN)\_D2}$	$T_{d\text{-}r\ max\_D2}$	8.4	µs	Duty Cycle 2
$t_{REC(MAX)\_D3}$ - $t_{DOM(MIN)\_D3}$	$T_{r\text{-}d\ max\_D3}$	15.9	µs	Duty Cycle 3
$t_{DOM(MAX)\_D4}$ - $t_{REC(MIN)\_D4}$	$T_{d\text{-}r\ max\_D4}$	17.28	µs	Duty Cycle 4
$t_{REC(MAX)\_low}$ - $t_{DOM(MIN)\_low}$	$T_{r\text{-}d\ max\_low}$	10.8	µs	Duty Cycle 1 at Low Battery Supply
$t_{DOM(MAX)\_low}$ - $t_{REC(MIN)\_low}$	$T_{d\text{-}r\ max\_low}$	8.4	µs	Duty Cycle 2 at Low Battery Supply

APPENDIX D - CALIBRATION/CONFIGURATION

D.1 LARGE CALIBRATION/CONFIGURATION METHOD

The mechanics of calibrating/configuring a LIN responder-node:

Calibrating/configuring a LIN responder-node is not a download. It uses a functional message to transfer data one time to the LIN responder-node. The LIN responder-node must retain this configuration information in EEPROM (or equivalent). In order to support a relatively large amount of configuration info (inclination/intrusion sensor), a multiplex message is used.

D.1.1 Calibrating/Configuration Data

Calibrating/configuration information is limited to data that is downloaded when the LIN responder-node is first installed and never changes. If the information can change, it is deemed to be personalization information and must be transmitted to the responder-node using one or more normal (non-configuration) LINxx schedule tables.

In order to avoid limiting the amount of calibrating/configuration information to what can fit into one LIN frame, we have chosen to use a multiplex type transmit style. The LIN commander-node will transmit an index byte (Calibrate/ConfigIndex) plus seven data bytes (Calibrate/ConfigData). The commander-node will always start with Calibrate/ConfigIndex x = 0 and then increment the Calibrate/Config Index every time it transmits the next set of Calibrate/ConfigData. When all of the data has been transmitted, the LIN commander-node will then go back and start transmitting from zero again. This is a simple algorithm in the LIN commander-node and allows different responder-nodes to have differing amounts of Calibrate/ConfigData.

A LIN responder-node will "know" that it has completely received all of the Calibrate/ConfigData when it sees Calibrate/ConfigIndex x = 0 a second time. At this point, the LIN responder-node will need to write out the calibrate/config data to EEPROM. Once the write is completed the LIN responder-node will lower the "I'm not configured" error flag (APINFO4 = 1).

D.1.2 Calibrate/Configuration Data Mapping

Table D1 explicitly lists how the multiplexed Calibrate/Config information signals are uniquely mapped in the LIN responder-node.

Table D1 - Multiplexed calibrate/configindex/calibrate/configdata mapping to unique internal LIN responder-node calibrate/config data

Calibrate/ConfigIndex Value	Calibrate/ConfigData Byte	LIN responder-node Calibrate/Config Data
0	Data0	LINCalibrate/Config[0]
0	Data1	LINCalibrate/Config[1]
0	Data2	LINCalibrate/Config[2]
0	Data3	LINCalibrate/Config[3]
0	Data4	LINCalibrate/Config[4]
0	Data5	LINCalibrate/Config[5]
0	Data6	LINCalibrate/Config[6]
1	Data0	LINCalibrate/Config[7]
1	Data1	LINCalibrate/Config[8]
1	Data2	LINCalibrate/Config[9]
1	Data3	LINCalibrate/Config[10]
1	Data4	LINCalibrate/Config[11]
1	Data5	LINCalibrate/Config[12]
1	Data6	LINCalibrate/Config[13]
2	Data0	LINCalibrate/Config[14]
2	Data1	LINCalibrate/Config[15]
2	Data2	LINCalibrate/Config[16]
2	Data3	LINCalibrate/Config[17]
2	Data4	LINCalibrate/Config[18]
2	Data5	LINCalibrate/Config[19]
2	Data6	LINCalibrate/Config[20]
3	Data0	LINCalibrate/Config[21]
3	Data1	LINCalibrate/Config[22]
3	Data2	LINCalibrate/Config[23]
3	Data3	LINCalibrate/Config[24]
3	Data4	LINCalibrate/Config[25]
3	Data5	LINCalibrate/Config[26]
3	Data6	LINCalibrate/Config[27]
•	•	•
•	•	•
•	•	•

Table D2 - Calibrate/configuring LIN responder-nodes - responder-node requirements

Rule
All LIN Responder-nodes shall store any configuration data in EEPROM/NVM. Configuration data will not be continuously transmitted during normal operation.
A responder-node shall infer that *configuration Mode* has started when it first receives any of these signals: *Calibration/ConfigIndex* signal *Calibration/ConfigData* signal *PartNumIndex* signal *PartNumData* signal **Note:** Calibration/configuration information is repeatedly transmitted until all responder-nodes on a subnet report "Calibrated/Configured" or 4 seconds elapse (whichever is sooner).
Once a responder-node detects that the subnet as switched to *Calibration/Configuration Mode* then it shall immediately start reporting "Not Calibrated/Configured" until the next requirement is met. The LIN responder-node must transmit the "Not Calibrated/Configured" status at least one time (even if the responder-node doesn't support configuration information).
The responder-node shall only indicate that it is configured after: All calibration/config information has been received All calibration/config information is saved in NVM/EEPROM
Note 1: The responder-node can detect that it has received all *Calibration/ConfigData* when the *Calibration/ConfigIndex* is zero the **second** time. **Note 2:** Even after a responder-node has been completely calibrated/configured, the responder-node may continue to receive the calibration/configuration information repeatedly. This allows other LIN responder-nodes with more calibration/configuration information to receive their calibration/config data. It also allows the LIN commander-node to collect all LIN responder-node part numbers (see next section).

D.1.3 Responder-Node Part Number Reporting

LIN responder-nodes are given the opportunity to report their Part Number at the same time as the LIN responder-node is placed into Calibrate/ConfigurationMode. See above for a description of when calibration/configuration is started.

The Mechanics of LIN Responder-Node Part Number Reporting:

Since we are limited to only 7 bytes of data (besides the LIN status byte), transmitting a 23 character part number will take several packets. So, like the configuration data, we use 1 byte to indicate an index and the rest of the bytes (six others) are the data.

D.1.4 Part Number Sequencing

In order to allow the LIN responder-node manufacturer to reduce cycle time on their production line, the LIN responder-node is allowed to start with any PartNumIndex. They then cycle through the remaining indices. Once they repeat the very first PartNumIndex value, the LIN commander-node will consider the Part Number complete. All unused bytes must be set to 0 (NULL).

Notice that PartNumIndex is multiplied by 6 when placing the partial part number data in the correct location. See D.1.6 for more information.

D.1.5 Commander-Node Part Number Retention

The LIN commander-node will erase these part numbers when the ignition is cycled in order to avoid misreporting LIN responder-node part numbers due to replacing one in service.

D.1.6 Responder-Node Part Number Mapping

Table D3 - Multiplexed partnumindex/parnumdata mapping to LIN commander-node partnumber data

PartNumIndex Value	PartNumData Byte	LIN Commander-Node PartNumber Data
0	Data0	PartNum[0]
0	Data1	PartNum[1]
0	Data2	PartNum[2]
0	Data3	PartNum[3]
0	Data4	PartNum[4]
0	Data5	PartNum[5]
1	Data0	PartNum[6]
1	Data1	PartNum[7]
1	Data2	PartNum[8]
1	Data3	PartNum[9]
1	Data4	PartNum[10]
1	Data5	PartNum[11]
2	Data0	PartNum[12]
2	Data1	PartNum[13]
2	Data2	PartNum[14]
2	Data3	PartNum[15]
2	Data4	PartNum[16]
2	Data5	PartNum[17]
3	Data0	PartNum[18]
3	Data1	PartNum[19]
3	Data2	PartNum[20]
3	Data3	PartNum[21]
3	Data4	PartNum[22]
3	Data5	PartNum[23]