



SURFACE VEHICLE RECOMMENDED PRACTICE

J2534™-2/11_0500

JAN2022

Issued

2022-01

Pass-Thru Extended Features - CAN with Flexible Data Rate

RATIONALE

Each of the features contained in the original SAE J2534-2 document (API version 04.04) is being published as a separate document for ease of maintenance and extensibility. At the same time, these features are being updated to be compatible with SAE J2534-1_0500 (API version 05.00) and to address feedback from the field.

TABLE OF CONTENTS

1.	SCOPE.....	3
2.	REFERENCES.....	3
2.1	Applicable Documents	3
2.1.1	SAE Publications.....	3
2.1.2	ISO Documents.....	3
3.	DEFINITIONS	3
4.	ACRONYMS	4
5.	DOCUMENTATIONS CONVENTIONS	4
6.	PASSTHRU INTERFACE REQUIREMENTS	4
6.1	Specific Protocol Requirements	5
6.2	Pin Usage.....	5
6.3	Data Buffering	6
6.3.1	Message Reception	6
6.3.2	Message Transmission	6
6.4	Error Handling	6
6.4.1	Device Not Connected	6
6.4.2	Receive Buffer Overflow	6
6.4.3	Message Termination.....	6
6.4.4	Network Errors	6
7.	WIN32 APPLICATION PROGRAMMING INTERFACE.....	7
7.1	API Changes	7
7.2	Message Structure	8
7.2.1	C/C++ Definition	8
7.2.2	Description of Structure Elements	8
7.2.3	Protocol Specific Details	10
7.2.4	Format Checks for Messages Passed to the API	11
7.2.5	Conventions for Returning Messages from the API.....	11

SAE Executive Standards Committee Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be revised, reaffirmed, stabilized, or cancelled. SAE invites your written comments and suggestions.

Copyright © 2022 SAE International

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of SAE.

TO PLACE A DOCUMENT ORDER: Tel: 877-606-7323 (inside USA and Canada)
Tel: +1 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
http://www.sae.org

SAE WEB ADDRESS:

Full document details: SAE J2534-2_11_5_00

For more information on this standard, visit
https://www.sae.org/standards/content/J2534-2/11_5_00

Documents

Document Sharing Hub

7.2.6	Indications	12
7.3	API Functions - Detailed Information	12
7.3.1	PassThruConnect	12
7.3.2	PassThruIoctl	15
7.4	IOCTL Section.....	15
7.4.1	GET_CONFIG/SET_CONFIG.....	16
7.4.2	CLEAR_TX_QUEUE.....	16
7.4.3	CLEAR_RX_QUEUE	16
7.4.4	CLEAR_PERIODIC_MSGS	16
7.4.5	CLEAR_MSG_FILTERS	17
7.4.6	BUS_ON	17
8.	DISCOVERY MECHANISM.....	17
8.1	<ioctlID> - GET_DEVICE_INFO	17
8.2	<ioctlID> - GET_PROTOCOL_INFO	17
8.3	<ioctlID> - GET_RESOURCE_INFO	20
9.	NOTES	21
9.1	Revision Indicator.....	21
Table 1	Required connector/pin combinations.....	5
Table 2	Minimum receive buffer size	6
Table 3	Minimum transmit buffer size	6
Table 4	Network errors.....	7
Table 5	SAE J2534 API functions	7
Table 6	RxStatus bit definitions.....	9
Table 7	TxFlags bit definitions	10
Table 8	List of indications.....	12
Table 9	List of description codes for the ErrInd indication	12
Table 10	Connect flag values.....	13
Table 11	Baud rate values	13
Table 12	Data rate values for the data phase portion of the CAN FD frame.....	14
Table 13	Resource structure example	14
Table 14	Return values for PassThruConnect.....	14
Table 15	Return values for PassThruIoctl.....	15
Table 16	IOCTL.....	15
Table 17	GET_CONFIG/SET_CONFIG.....	16
Table 18	GET_CONFIG/SET_CONFIG details	16
Table 19	GET_DEVICE_INFO.....	17
Table 20	GET_DEVICE_INFO details	17
Table 21	GET_PROTOCOL_INFO	18
Table 22	GET_PROTOCOL_INFO details	18
Table 23	GET_RESOURCE_INFO.....	20
Table 24	GET_RESOURCE_INFO details	20

1. SCOPE

This SAE Recommended Practice is part of the SAE J2534-2/X_0500 set of documents that extends the SAE J2534-1_0500 API (version 05.00) specification, and defines how to implement CAN with flexible data rate (CAN FD) within the SAE J2534 API framework. This document details only the changes from SAE J2534-1_0500 and items not specifically detailed in this document are assumed to have not changed.

An SAE J2534-2/11_0500 interface shall be compliant to the CAN with flexible data rate (CAN FD) feature only when all the required functionality in this SAE Recommended Practice is implemented. Any functionality not required for compliance will be specifically marked as “optional” in this document.

This document must be used in conjunction with the SAE J2534-2/BA_0500 and SAE J2534-2/RE_0500 documents.

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), www.sae.org.

SAE J1962	Diagnostic Connector
SAE J2284-4	High-Speed CAN (HSC) for Vehicle Applications at 500 Kbps with CAN FD Data at 2 Mbps
SAE J2284-5	High-Speed CAN (HSC) for Vehicle Applications at 500 Kbps with CAN FD Data at 5 Mbps
SAE J2534-1_0500	Recommended Practice for Pass-Thru Vehicle Programming (API version 05.00)
SAE J2534-2/BA_0500	Pass-Thru Extended Features - Base Document (API version 05.00)
SAE J2534-2/RE_0500	Pass-Thru Extended Features - Resource Document (API version 05.00)

2.1.2 ISO Documents

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

ISO 11898-1:2015	Road Vehicles - Controller Area Network (CAN)
ISO 11898-3	Road Vehicles - Controller Area Network (Flexible Data Rate CAN)

3. DEFINITIONS

ACTIVE MESSAGE: Refers to a message that has started the transmission process on the physical vehicle bus but has not completed. (An example of an active message is: a 512 byte J1939 BAM Message where the first 27 bytes have been transmitted but the remainder has not.)

CAN MESSAGE FORMAT: Refers to the message format of the CAN frame, CAN2.0 or CAN FD.

LOGICAL COMMUNICATION CHANNEL: A specific path through the Pass-Thru Interface to the physical vehicle bus that uses an existing Physical Communication Channel but adds the Network and Transport layers (as defined by OSI).

PASS-THRU DEVICE: Refers only to the physical hardware portion of the SAE J2534 Interface, where the circuitry resides.

PASS-THRU INTERFACE: Refers to all components of a SAE J2534 Interface, including both hardware and software.

PHYSICAL COMMUNICATION CHANNEL: A specific path through the Pass-Thru Interface to the physical vehicle bus that encompasses the physical layer and data link layer (as defined by OSI) and includes all the associated physical resources (such as, the data link controller, connector, pins, etc.) in the Pass-Thru Device.

QUEUED MESSAGE: Refers to a full message from the application that has been accepted by the Pass-Thru Interface (via the API) for transmission, which is intended to become an Active Message at some point in the future.

TERMINATE A MESSAGE: Refers to the process of stopping reception or transmission of a message by the Pass-Thru Interface.

TRANSMITTED MESSAGE: Refers to a previously Active Message that has been successfully transmitted on the physical vehicle bus.

4. ACRONYMS

API	Application Programming Interface
BRS	Bit Rate Switch
CAN	Controller Area Network (2.0b)
CAN FD	CAN with Flexible Data rate Controller Area Network
CAN ID	Controller Area Network Message Identifier (11 or 29 bit)
CRC	Cyclic Redundancy Check
DLL	Dynamic Link Library
EDL	Extended Data Length
FDF	Flexible Data rate Format Indicator (also known as EDL)
IOCTL	Input/Output Control
IOCTLID	Identifier assigned to a specific ioctl

5. DOCUMENTATIONS CONVENTIONS

In the document body and in tables:

- All SAE J2534 API functions shall be in bold print (for example, **PassThruOpen**).
- All SAE J2534 API variables shall be enclosed in "<" and ">" (for example, <DeviceID>).
- All SAE J2534 API DEFINES shall be in upper-case print (for example, STATUS_NOERROR).
- All numbers shall be in decimal unless otherwise indicated. (Hexadecimal values have a leading "0x"; for example, 0x7AF or 0x800 are hexadecimal values.)

6. PASSTHRU INTERFACE REQUIREMENTS

The CAN FD feature shall be implemented as a physical communication channel and cannot be implemented on a logical communication channel.

This feature shall comply with the CAN FD protocol as specified in the SAE J2284-4 and SAE J2284-5 specifications.

6.1 Specific Protocol Requirements

A pass-thru interface that supports the CAN FD feature shall:

- Support 125k, 250k, 500k, and 1000k bps arbitration phase bit rates.
- Support data phase bit rates of 125k, 250k, 500k, 1000k, 2000k, 4000k, and 5000k bps.
- Support 11 and 29 bit CAN identifiers for both CAN 2.0 and CAN FD messages.
- Be capable of transmitting both CAN 2.0 and CAN FD messages.
- Be capable of receiving both CAN 2.0 and CAN FD messages.
- Support frame sizes as defined by CAN FD protocol specification.
- Support a switchable terminating resistance between CAN high and CAN low. Refer to SAE J2284-3, SAE J2284-4, and SAE J2284-5 for resistor specifications.
- Support filters that ignore the CAN message format; the message will be filtered on the address and data regardless of whether the message was received as a CAN 2.0 or CAN FD message.
- Support a minimum of 64 logical communication channels per physical communication channels.
- Support periodic messages that are limited to a maximum <DataLength> of 12 bytes, which includes the 4-byte CAN ID.

6.2 Pin Usage

When the pass-thru interface is initially powered or when the CAN FD feature is no longer in use, the pins shall default to high impedance. High impedance is defined as greater than 500 kΩ impedance relative to both signal ground and chassis ground.

[Table 1](#) defines the required connector/pin usage for the CAN FD feature as they appear in the structure RESOURCE_STRUCT, where the order of elements in the array ResourceListPtr defines its purpose. Other connector/pin combinations may optionally be supported by the pass-thru interface and must be queried the using the discovery mechanism.

Table 1 - Required connector/pin combinations

RESOURCE_STRUCT Contents
<ResourceList.Connector> = J1962_CONNECTOR; <ResourceList.NumOfResources> = 2; <ResourceList.ResourceListPtr[0]> = 6; // CAN_H <ResourceList.ResourceListPtr[1]> = 14; // CAN_L
<ResourceList.Connector> = J1962_CONNECTOR; <ResourceList.NumOfResources> = 2; <ResourceList.ResourceListPtr[0]> = 3; // CAN_H <ResourceList.ResourceListPtr[1]> = 11; // CAN_L
<ResourceList.Connector> = J1962_CONNECTOR; <ResourceList.NumOfResources> = 2; <ResourceList.ResourceListPtr[0]> = 12; // CAN_H <ResourceList.ResourceListPtr[1]> = 13; // CAN_L
<ResourceList.Connector> = J1962_CONNECTOR; <ResourceList.NumOfResources> = 2; <ResourceList.ResourceListPtr[0]> = 1 ; // CAN_H <ResourceList.ResourceListPtr[1]> = 9; // CAN_L



6.3 Data Buffering

6.3.1 Message Reception

The CAN FD feature shall provide a receive buffer as specified in [Table 2](#) and shall be capable of receiving and buffering messages while the receive buffer is not full. The number of messages buffered will depend on the size of the messages received. This capability shall be available on each channel at all supported baud rates.

Table 2 - Minimum receive buffer size

Maximum Size of EACH Message	Minimum Receive Buffer Size
68 Bytes	4128 Bytes

6.3.2 Message Transmission

The CAN FD feature shall provide a transmit buffer as specified in [Table 3](#) for individual messages (using **PassThruQueueMsgs**) and shall be capable of buffering messages while the transmit buffer is not full. The number of messages buffered will depend on the size of the messages to be transmitted. This capability shall be available on each channel at all specified baud rates.

Table 3 - Minimum transmit buffer size

Maximum Size of Each Message	Minimum Transmit Buffer Size
68	4128 bytes

NOTE: This buffer is only for messages filled using **PassThruQueueMsgs**.

6.4 Error Handling

6.4.1 Device Not Connected

As defined in SAE J2534-1_0500.

6.4.2 Receive Buffer Overflow

As defined by CAN in SAE J2534-1_0500.

6.4.3 Message Termination

As defined for CAN in SAE J2534-1_0500. The message termination boundary is at the CAN FD frame.

6.4.4 Network Errors

The pass-thru interface shall be capable of detecting transmit and receive errors. To deal with these errors, generic recovery strategies have been defined SAE J2534-1_0500. [Table 4](#) details the types of errors to be detected and the associated recovery strategy, as well as the required user notification that shall occur at the end of the designated recovery strategy.

Table 4 - Network errors

Error	Application Notification Mechanism	Recovery Strategy
Acknowledgement error or loss of arbitration	NONE or TxFailed indication (if a TxDone was requested)	Controller retry strategy as defined in SAE J2534-1_0500
Bus warning	NONE	Controller retry strategy as defined in SAE J2534-1_0500
Overrun error	Set BUFFER_OVERFLOW in <RxStatus>	Drop strategy as defined in SAE J2534-1_0500
During message reception: form error, stuff error, CRC error, bit 0 error, bit 1 error	NONE	NONE
During message transmission: form error, stuff error, CRC error, bit 0 error, bit 1 error	NONE or TxFailed indication (if a TxDone was requested)	Controller retry strategy as defined in SAE J2534-1_0500
BUS OFF	Generate LINK_DOWN ErrInd indication	Passive operation strategy as defined in SAE J2534-1_0500
During message reception: a network error for which the message was still correctly received	Set LINK_FAULT in <RxStatus>	NONE

7. WIN32 APPLICATION PROGRAMMING INTERFACE

7.1 API Changes

[Table 5](#) summarizes the changes to the SAE J2534-1_0500 API functions for this feature. The pass-thru API DLL must include entry points for all functions, regardless of whether or not the API function call is used for this feature.

Table 5 - SAE J2534 API functions

Function	Description
PassThruScanForDevices	Function implementation independent of this feature
PassThruGetNextDevice	Function implementation independent of this feature
PassThruOpen	As defined in SAE J2534-1_0500
PassThruClose	As defined in SAE J2534-1_0500
PassThruConnect	Added new <ProtocolID> values; added additional element to resource list to support data phase bit rate
PassThruDisconnect	As defined in SAE J2534-1_0500
PassThruLogicalConnect	This function shall always return an error (applicable error values defined in SAE J2534-1_0500)
PassThruLogicalDisconnect	As defined in SAE J2534-1_0500
PassThruSelect	As defined in SAE J2534-1_0500
PassThruReadMsgs	Added new RxStatus bit to the PASSTHRU_MSG structure
PassThruQueueMsgs	Added new TxFlags bit to the PASSTHRU_MSG structure
PassThruStartPeriodicMsg	Added new TxFlags bit to the PASSTHRU_MSG structure
PassThruStopPeriodicMsg	As defined in SAE J2534-1_0500
PassThruStartMsgFilter	<TxFlag> CAN FD format bits are ignored by filters
PassThruStopMsgFilter	As defined in SAE J2534-1_0500
PassThruSetProgrammingVoltage	Function implementation independent of this feature
PassThruReadVersion	Function implementation independent of this feature
PassThruGetLastError	Function implementation independent of this feature
PassThruIoctl	Added discovery mechanism, added new IOCTLs to support CAN FD capability