SAE The Engineering Society For Advancing Mobility Land Sea Air and Space® INTERNATIONAL

SURFACE VEHICLE RECOMMENDED PRACTICE

SAE J1939-73

REV JUN2001

Issued 1996-02
Revised 2001-06
Superseding J1939-73 FEB96

Submitted for recognition as an American National Standard

Application Layer—Diagnostics

Foreword—The J1939 series of SAE Recommended Practices has been developed by the Truck and Bus Control and Communications Network Subcommittee of the Truck and Bus Electrical and Electronics Committee. The objectives of the subcommittee are to develop information reports, recommended practices, and standards concerned with the requirements, design, and usage of devices which transmit electronic signals and control information among vehicle components.

These SAE Recommended Practices are intended as a guide toward standard practice and are subject to change so as to keep pace with experience and technical advances.

TABLE OF CONTENTS

1. Scope

The SAE J1939 series of recommended practices are intended for light- and heavy-duty vehicle uses on- or off-road as well as appropriate stationary applications which use vehicle-derived components (e.g. generator sets). Vehicles of interest include, but are not limited to: on- and off-highway trucks and their trailers, construction equipment, and agriculture equipment and implements.

The purpose of these documents is to provide an open interconnect system for on-board electronic systems. It is the intention of these documents to allow electronic devices to communicate with each other by providing a standard architecture.

J1939-73 identifies the diagnostic connector to be used for the vehicle service tool interface and defines messages to accomplish diagnostic services. California-regulated OBD II requirements are satisfied with a subset of the specified connector and the defined messages. Diagnostic messages (DMs) provide the utility needed when the vehicle is being repaired. Diagnostic messages are also used during vehicle operation by the networked electronic control modules to allow them to report diagnostic information and self-compensate as appropriate, based on information received. Diagnostic messages include services such as periodically broadcasting active diagnostic trouble codes, identifying operator diagnostic lamp status, reading or clearing diagnostic trouble codes, reading or writing control module memory, providing a security function, stopping/starting message broadcasts, reporting diagnostic readiness, monitoring engine parametric data, etc.

2. References

2.1 Applicable Publications—General information regarding this series of recommended practices is found in SAE J1939. The latest issue of the SAE J1939 publications shall apply.

2.1.1 SAE PUBLICATIONS—Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001. Unless otherwise specified, the latest publication shall apply.

SAE J1587—Joint SAE/TMC Electronic Data Interchange Between Microcomputer Systems in Heavy-Duty Vehicle Applications

SAE J1939—Serial Control and Communications Vehicle Network

SAE J1939-13—Off-Board Diagnostic Connector

SAE J1939-21—Data Link Layer

SAE J1939-71—Vehicle Application Layer

SAE J1979—E/E Diagnostic Test Modes

2.1.2 CALIFORNIA AIR RESOURCES BOARD (CARB) PUBLICATION—Available from Air Resources Board, Haagen-Smit Laboratory, 9528 Telstar Avenue, El Monte, CA 91731-2990. Telephone (818) 575-6800.

Mail Out #95-03, January 19, 1995.

OBD II, California code of regulations, Title 13, 1968.1: Malfunction and Diagnostics Systems Requirements, 1994 and subsequent model year passenger cars, light-duty trucks, and medium-duty vehicles with feedback fuel control systems.

3. Definitions

3.1 Active—The state used to indicate that a fault is currently occurring. Active dominates Previously Active. A fault cannot be both “Active” and “Previously Active” simultaneously.

3.2 Broadcast—Messages that are sent on a periodic basis without having to be solicited. In some cases Broadcasts may be normally off and solicited to come on and then stay on until they are solicited to turn off (see DM13).

3.3 Calibration—The software installed in a control module. This includes executable code and calibration data.

3.4 Continuously Monitored Systems—Continuously Monitored Systems are those which are monitored approximately two times per second. Note that some continuous monitors may require many conditions to be true before monitoring can be performed.

3.5 Diagnostic Trouble Code—A 4-byte value that identifies the kind of trouble, the associated failure mode and its occurrence count.

3.6 Freeze Frame—A sampling of a group of parameters based on the occurrence of a diagnostic trouble code.

3.7 Key—The result of a set of mathematical operations performed upon a Seed to provide a device with a means of authenticating a Tool’s request.

3.8 Malfunction Indicator Lamp—The MIL is used to report trouble codes that are emissions related. Trouble codes that are not emissions related do not illuminate the MIL.

3.9 Memory Access—This defines a set of messages (DM14 through DM18) and outlines the operational procedures for a Tool (or device) wishing to read or write the memory, or storage space, of a device with or without data security.

3.10 Non-continuously Monitored Systems—System monitors that run once a trip. Trip, in this context, is as defined by OBD II. It should be noted that there will be monitors that won’t run every trip, i.e. cold-start aid monitors may only run when the ambient temperature is below 10 °C (50 °F).

3.11 Object—Some entity within a memory and/or a SPACE.

3.12 Password—The number sent when using a simple authentication technique wherein both the device and Tool have a prior knowledge of the specific number and usually use equality as the verification.

3.13 Pointer—A term used to label a device that identifies the memory location which should be read or written. Types of pointers include: direct memory address and directed spatial addressing (Suspect Parameter Number (SPN) Space, OEM Proprietary Space, and Reserved to be assigned).

3.14 Port—Physical connection point(s) from a control module to a specific communications link (see DM13).

3.15 Previously Active—The state used to indicate that a fault has occurred but is not presently occurring. A fault cannot be both “Active” and “Previously Active” simultaneously.

3.16 Readiness Code—This is a code which is set once all of the emissions-related diagnostics have been performed. The readiness code shall be set before the completion of the cold portion of the Federal Test Procedure for Emissions Certification. Service tools can interrogate a controller to determine the state of the readiness code. If it is set, then the controller has had the opportunity to test all emissions-related diagnostics (see DM5).

For example, on engine start up, it is likely that an in-range type of diagnostic on the intake manifold pressure might require operation at speed and torque points beyond the idle condition. Therefore, if the system had its trouble codes erased and then the engine was restarted, the readiness code would not be set until the condition for the in-range intake manifold pressure tests had been executed. It is also expected that other tests, not just the intake manifold pressure, would need to be performed before the readiness code could be set. The readiness code shall go to the not ready condition each time the vehicle is powered down.

3.17 Seed—A number sent by a device to a Tool to obtain authentication of the Tool's right to access the device.

3.18 Space—A memory region containing a collection of OBJECTS.

3.19 User_Level—A number sent by a Tool to a device along with an initial request to inform the device of some specific level of access that the Tool wishes to gain.

4. Abbreviations

CARB	California Air Resources Board
CM	SPN Conversion Method
DM1	Diagnostic Message 1, Active Diagnostic Trouble Codes
DM2	Diagnostic Message 2, Previously Active Diagnostic Trouble Codes
DM3	Diagnostic Message 3, Diagnostic Data Clear/Reset for Previously Active DTCs
DM4	Diagnostic Message 4, Freeze Frame Parameters
DM5	Diagnostic Message 5, Diagnostic Readiness
DM6	Diagnostic Message 6, Continuously Monitored Systems Test Results
DM7	Diagnostic Message 7, Command Non-continuously Monitored Test
DM8	Diagnostic Message 8, Test Results for Non-continuously Monitored Systems
DM9	Diagnostic Message 9, Oxygen Sensor Test Results
DM10	Diagnostic Message 10, Non-continuously Monitored Systems Test Identifiers Support
DM11	Diagnostic Message 11, Diagnostic Data Clear/Reset for Active DTCs
DM12	Diagnostic Message 12, Emissions Related Active DTCs
DM13	Diagnostic Message 13, Stop Start Broadcast
DM14	Diagnostic Message 14, Memory Access Request
DM15	Diagnostic Message 15, Memory Access Response
DM16	Diagnostic Message 16, Binary Data Transfer
DM17	Diagnostic Message 17, Boot Load Data
DM18	Diagnostic Message 18, Data Security
DM19	Diagnostic Message 19, Calibration Information

DTC	Diagnostic Trouble Code
EDC	Error Detection and/or Correction
EDCP	Error Detection and/or Correction Parameter
FMI	Failure Mode Indicator
FTP	Federal Test Procedure
MA	Memory Access
MIL	Malfunction Indicator Lamp
NA	Not applicable
PG	Parameter Group
PID	Parameter Identifier (SAE J1587 or SAE J1979)
OBD II	On-board Diagnostics II
OC	Occurrence Count
SPN	Suspect Parameter Number

See SAE J1939 for any terms and or definitions not found in this document.

5. Technical Requirements

5.1 General— The diagnostic definitions provided herein are intended to satisfy the needs of all potential users of the SAE J1939 network. These definitions are intended to be suitable for applications in any of the industry groups defined within SAE J1939. A broad range of capabilities are provided with provision made for future growth. Additional features, Parameter Groups and Parameter definitions will be defined over time; it is anticipated that this document will continuously evolve as long as the SAE J1939 network is an active Recommended Practice. Such growth will be implemented in such a way as to ensure backward compatibility with earlier versions. At the time of initial publication, many of these growth areas are identified but are yet to be defined. Such identification is provided so that the reader will be aware of those additions that are already planned for the document.

5.2 Overview of Diagnostic Requirements—The diagnostic requirements necessary to provide the type of capability our customers, our industry, and the regulatory bodies are demanding is outlined in 5.2.1. A description of the minimum requirements needed to satisfy regulatory requirements is contained in 5.2.2. A discussion of the general operating conditions for diagnostic procedures is defined in 5.2.3.

5.2.1 DIAGNOSTIC CAPABILITIES ENVISIONED—The following capabilities will be defined in this and future publications of this document:

a. Security—Define a security scheme to be used on the serial data link that allows the industry standard service tools to be able to perform tasks that are necessary during service procedures. This will include accessing diagnostic information, accessing vehicle configuration information and recalibrating control modules.

b. Connectors—Define the connector to be used for connection to the vehicle SAE J1939 network for service tools. The diagnostic connector is defined in SAE J1939-13.

c. Diagnostic Status Message Support—Provide a set of messages that allows the reading of fault information, clearing of fault information, monitoring of vehicle parameters, access to vehicle and component configuration, and other related information.

d. Diagnostic Test Support—Provide a capability that allows the service Tool to put the various controllers into specific test modes in order to determine proper subsystem operation.

5.2.2 SUGGESTED DIAGNOSTIC SUPPORT

5.2.2.1 Emission Related Components—As a minimum capability, all controllers using SAE J1939 that impact emissions and must comply with OBD II or OBD shall support the following functions: read diagnostic trouble codes (see DM1 and DM12); clear diagnostic trouble codes (see DM11 and DM3); read freeze frame data (see DM4); access to real-time information (see 5.5); access to last trip test results (see DM6); and system readiness code access (see DM5); and report Vehicle Identification Number and Calibration Verification Number (see DM19). In addition they shall support J1939-71 PGNs: 65262 (Engine Temperatures: Engine Coolant Temperature); 65265 (Cruise Control/Vehicle Speed: Wheel Based Vehicle Speed); 65270 (Inlet/Exhaust Conditions: Boost Pressure, Intake Manifold Temperature); 61443 (Electronic Engine Controller #2: Accelerator pedal position, Percent Load at current speed); 61444 (Electronic Engine Controller #1: Actual Engine Percent Torque, Engine Speed); 65159 (Ignition Timing; Actual Ignition Timing); 60416 (Transport Protocol-Connection Management); 59392 (Acknowledgement Message); 59904 (Request PGN); and 60160 (Transport Protocol-Data Transfer).

5.2.2.2 Non-emission Related Components—To be determined in later revisions of this document.

5.2.3 GENERAL CONDITIONS FOR DIAGNOSTIC PROCEDURES—These guidelines are necessary to ensure proper operation of both the test equipment and the vehicle during diagnostic procedures. Test equipment, when using messages defined in this document, should not affect normal operation of the vehicle except when that is the express purpose of the message.

The off-board test equipment may request data without knowledge of which module on the vehicle will respond. These requests may also be directed to a specific device. The proper method should be used in order to reduce network traffic. In some vehicles, multiple controllers may respond with the information requested. In addition, a single module may send multiple responses to a single request. Any test device requesting information must, therefore, have provisions for receiving multiple responses.

The on-board systems should respond to a request as defined in SAE J1939-21. With multiple responses possible from a single request, this allows as much time as is necessary for all modules to access the data link and transmit their response(s). If there is no response within this time period (i.e. 0.25 seconds), the Tool can either assume no response will be received, or if a response has already been received, that no more responses will be received.

A Tool should always wait for a response from the previous request, or “no response” time-out before sending another request. In no case should a request be sent in less than the times specified in SAE J1939-21 after the previous request.

Destination specific requests require a response. If a request for a parameter group is not supported by the module and a destination specific request was used, a NACK is required (see SAE J1939-21 PGN 59392). If the request for a parameter group was sent to a global destination address and it is not supported by a given device, then that device must not NACK the request.

Unless otherwise specified, parameter values should be formatted in accordance with the parameter ranges as defined by SAE J1939-71 section 5.1.3.

5.3 Security—One of the purposes of this Recommended Practice is to provide a standard protocol (a set of capabilities or diagnostic services) to allow users to access and modify memory areas inside a controller on the network. For these tools to be supported by the manufacturers of the ECUs that will be designed to connect to the J1939 network, sufficient protection against “unauthorized” modifications must be included. The messages described in sections (R) 5.7.14 (DM14) though (R) 5.7.18 (DM18) and their subsections are to be used for this purpose.

This security shall not be used to limit access to the capabilities defined in section 5.7.1 (DM1) through section 5.7.13 (DM13) and section (R) 5.7.19 (DM19), but is intended to allow manufacturers to limit the data that can be accessed by the user. The security systems outlined here represent a recommendation for ECU manufacturers and provide flexibility for them to tailor individual systems to their specific security needs. The vehicle modules addressed are those that are capable of having solid-state memory contents altered by an external command sent through this vehicle communication link. Improper memory content alteration could potentially damage the electronics, reduce the vehicle’s compliance to legislated requirements, or breach the vehicle manufacturer’s security interests.

Proper “Unlocking” of the controller shall be a prerequisite to access certain critical on-board controller functions. Access to the on-board controller while in a “Locked” mode is permitted only as determined by the controller’s manufacturer. This may require that the user obtain specific codes or passwords directly from the manufacturer’s representative and may only be possible when using product-specific software. This permits the controller to protect itself from unauthorized intrusion.

The messages in sections (R) 5.7.14 (DM14) though (R) 5.7.18 (DM18) do not attempt to define capability as a requirement for any controller or to specify what information should be subject to any specific security measures; these decisions are left to the controller manufacturer. Implementation of the security system shall not prevent basic diagnostic communications between an external Tool and the on-board controller.

(R) Appendix C, (R) Appendix D and (R) Appendix E contain additional information that may help implementers understand the intended use of these security processes for gaining access to controller memory and the several different modes available for limiting access areas of that memory.

5.4 Diagnostic Connector—The diagnostic connector is defined in SAE J1939-13.

5.5 Parameter Monitoring Requirements—The parameter definitions shall be those of the referenced SAE J1939 Application Layer document. Any parameter that has been defined in an applications layer document and is included in a Parameter Group (PG) shall be used for diagnostics. Therefore, if a parameter has already been defined, it will not be redefined for diagnostic purposes. In some cases it will be necessary to identify a closely related parameter, such as the value of the accelerator pedal sensor reading when the failure occurred rather than the current reading of the accelerator pedal sensor.

5.6 Diagnostic Trouble Code Definition—A Diagnostic Trouble Code (DTC) is made up of four (4) independent fields, as follows:

a.	Suspect Parameter Number	(SPN)	19	bits
b.	Failure Mode Identifier	(FMI)	5	bits
c.	Occurrence Count	(OC)	7	bits
d.	SPN Conversion Method	(CM)	1	bit

These independent parameters are not used together to form a number. They are merely a set of information that helps in understanding the failure that is being reported.

A diagnostic Tool may also want to use the controller source address and the Name to determine which controller is reporting the diagnostic information. This information is not needed to interpret the SPN but may be beneficial to have during the diagnostic process. Reference SAE J1939 for the Source Address and Name definitions.

Diagnostic trouble codes are transmitted as 4 bytes per trouble code. Those 4 bytes are interpreted as defined in 5.7.1. In an effort to provide continuity between the diagnostics defined in SAE J1587 to that of SAE J1939-73, the fault encoding format remains very similar. When possible SAE J1587 PID numbers have been mapped one for one as SPNs.

Examples of diagnostic trouble codes (see Table 1):

EXAMPLE 1—This is a SAE J1587 parameter.

SPN=91	Suspect parameter is accelerator pedal position
FMI=3	Failure mode is identified as voltage above normal
OC=5	Occurrence count indicates trouble has occurred 5 times
CM= 0_2	(1 bit)

EXAMPLE 2—This is not a parameter communicated as an SAE J1587 PID. Therefore, it is assigned a number above 511.

SPN=656	Suspect parameter is engine injector number 6
FMI=3	Failure mode is identified as voltage above normal
OC=2	Occurrence count indicates trouble has occurred 2 times
CM= 0_2	(1 bit)

EXAMPLE 3—Diagnostic Trouble Code as transmitted in diagnostic messages (e.g. DM1)

Given:

Parameter “Pre-Filter Oil Pressure,” Suspect Parameter Number 1208
Failure Mode Identifier of 3
Occurrence Count of 10
SPN Conversion Method of 0
All fields of DTC sent in Intel Format (least significant byte first)

	Decimal	Hexadecimal	Binary
SPN	1208	= $4B8_{16}$	= 000 00000100 10111000_2 (19 bits)
FMI	3	= 3_{16}	= 00011_2 (5 bits)
OC	10	= A_{16}	= 0001010_2 (7 bits)
CM			= 0_2 (1 bit)

TABLE 1 DTC REPRESENTATION IN CAN DATA FRAME FOR DM1 (BYTE 3 CLOSER TO CAN IDENTIFIER)

<table>
<tr><td></td><td colspan="32">DTC</td></tr>
<tr><td></td><td colspan="8">Byte 3
8 least significant bits of SPN
(bit 8 most significant)</td><td colspan="8">Byte 4
second byte of SPN
(bit 8 most significant)</td><td colspan="8">Byte 5
3 most significant bits of SPN and the FMI
(bit 8 SPN msb and bit 5 FMI msb)</td><td colspan="8">Byte 6</td></tr>
<tr><td></td><td colspan="19">SPN</td><td colspan="5">FMI</td><td>CM</td><td colspan="7">OC</td></tr>
<tr><td>J1939 Frame Format</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr>
<tr><td></td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
</table>

5.7 Diagnostic Parameter Group Definitions—This section contains definitions of those parameter groups that will be used specifically for diagnostics. The format is a little different than the applications SAE J1939-71 layer in that the parameter definitions will follow each parameter group definition as a subsection under that parameter group.

One of the goals of this diagnostic document is to satisfy the OBD II requirements. One of the documents that contains many of the OBD II requirements is SAE J1979. For that reason, Table 2 was created as a way of identifying how SAE J1939 satisfies the SAE J1979 requirements.

A summary listing of all Diagnostic Modes and PIDs from SAE J1979 and their corresponding SAE J1939 PGNs is provided (see Table 2)

TABLE 2 SUMMARY OF DIAGNOSTIC MODE ASSIGNMENTS.

	SAE Motor Vehicle Council SAE J1979 Functions	SAE Motor Vehicle Council SAE J1979 Functions	SAE Motor Vehicle Council SAE J1979 Functions	SAE Truck and Bus Council SAE J1939 Support of Those Functions	SAE Truck and Bus Council SAE J1939 Support of Those Functions
	SAE J1979 Description	**SAE J1979 Mode**	**SAE J1979 PID**	**SAE J1939 DM (PGN)**	**PGN Description**
1	Supported PIDs	01 request 41 response	00	NA	SAE J1939 provides a method for unavailable parameters to be identified
2	Number of DTCs, MIL status and diagnostic monitors supported and their status	01 request 41 response	01	DM5 (65230)	OBD compliance, previously active and active DTC count, monitors supported and their status (diagnostic readiness)
3	Parameters related to the engine operation	01 request 41 response	3 to $1B_{16}$	various PGNs	Normally provided PGs will be used to retrieve these parameters; for example, SAE J1939-71 PGN 61444 contains engine speed
4	Determine OBD type supported (OBD II-ARB, OBD-Federal, OBD and OBD II, OBD 1, other)	01 request 41 response	$1C_{16}$	DM5 (65230)	Tells which OBD support is provided
5	PIDs supported in freeze frame	02 request 42 response	00	DM4 (65229)	Freeze frame definition and support covered in DM4
6	DTC that caused freeze frame	02 request 42 response	02	DM4 (65229)	Freeze frame PG tells what DTC caused it.
7	PID data value in freeze frame record	02 request 42 response	03 to $0D_{16}$	DM4 (65229)	Freeze frame PG contains all parameters (more than one freeze frame can be supported)
8	Emission-related powertrain DTCs	03 request 43 response	NA	DM12 (65236)	Emission-related active DTCs and lamp status information
9				DM1 (65226)	Active DTCs and lamp status information
10				DM2 (65227)	Previously active DTCs and lamp status information

	SAE Motor Vehicle Council SAE J1979 Functions SAE J1979 Description	SAE Motor Vehicle Council SAE J1979 Functions SAE J1979 Mode	SAE Motor Vehicle Council SAE J1979 Functions SAE J1979 PID	SAE Truck and Bus Council SAE J1939 Support of Those Functions SAE J1939 DM (PGN)	SAE Truck and Bus Council SAE J1939 Support of Those Functions PGN Description
11	Clear emission-related diagnostic information	04 request 44 response	NA	DM11 (62235)	Clear diagnostic information for active DTCs
12				DM3 (62228)	Clear diagnostic information for previously active DTCs
13	Oxygen sensor monitoring test results	05 request 45 response	NA	DM9 (65233)	Report oxygen sensor test results
14	On-board monitoring test results for non-continuous monitored systems	06 request 46 response	NA	DM10 (65234)	Test IDs supported
15				DM7 (58112)	Invoke test
16				DM8 (65232)	Test Results
17	On-board monitoring test results for continuously monitored systems	07 request 47 response	NA	DM6 (65231)	Test results for premature DTCs
18	Request control of on-board system, test, or component	08 request	NA	DM7 (58112)	Command on-board system, test, or component
19		48 response		DM8 (65232)	Results commanded system, test, or component
20	Calibration ID	09 request 49 response	NA	DM19 (54016)	Calibration Information (bytes 5-20 are Calibration ID
21	Calibration Verification Number	09 request 49 response	NA	DM19 (54016)	Calibration Information (bytes 1-4 are CVN)

5.7.1 ACTIVE DIAGNOSTIC TROUBLE CODES (DM1)—The information communicated is limited to the currently active diagnostic trouble codes preceded by the diagnostic lamp status. Both are used to notify other components on the network of the diagnostic condition of the transmitting electronic component. The data contains the lamp status and a list of diagnostic codes and occurrence counts for currently active diagnostic trouble codes. This is all DTCs including those that are emissions related.

The currently defined lamps (Malfunction Indicator Lamp, Red Stop Lamp, Amber Warning Lamp, and Protect Lamp) are associated with DTCs. If the transmitting electronic component does not have active DTCs, then the lamp status from that component will indicate that the lamps should be off. However, the component controlling the actual lamp illumination must consider the status from all components that use these lamps before changing the state of the lamps.

There may be applications that require additional lamp definitions to accomplish their function (e.g. a lamp that indicates when cruise control is actively controlling would require a separate lamp in another PG).

Transmission Rate:		A DM1 message is transmitted whenever a DTC becomes an active fault and at a normal update rate of only once per second thereafter. If a fault has been active for 1 second or longer, and then becomes inactive, a DM1 message shall be transmitted to reflect this state change. If a different DTC changes state within the 1 second update period, a new DM1 message is transmitted to reflect this new DTC. To prevent a high message rate due to intermittent faults that have a very high frequency, it is recommended that no more than one state change per DTC per second be transmitted. Thus a DTC that becomes active/inactive twice within a 1 second interval, such as shown in Example Case 1, would have one message identifying the DTC becoming active, and one at the next periodic transmission identifying it being inactive. This message is sent only when there is an active DTC existing or in response to a request. Note that this Parameter Group will require using the “multipacket Transport” Parameter Group (reference SAE J1939-21) when more than one active DTC exists.	
Data Length:		Variable	
Data page:		0	
PDU Format:		254	
PDU Specific:		202	
Default Priority:		6	
Parameter Group Number:		65226 ($00FECA_{16}$)	
Byte: 1	bits 8-7	Malfunction Indicator Lamp Status	see 5.7.1.1
	bits 6-5	Red Stop Lamp Status	see 5.7.1.2
	bits 4-3	Amber Warning Lamp Status	see 5.7.1.3
	bits 2-1	Protect Lamp Status	see 5.7.1.4
Byte: 2	bits 8-7	Reserved for SAE assignment Lamp Status	
	bits 6-5	Reserved for SAE assignment Lamp Status	
	bits 4-3	Reserved for SAE assignment Lamp Status	
	bits 2-1	Reserved for SAE assignment Lamp Status	
Byte: 3	bits 8-1	SPN, 8 least significant bits of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 4	bits 8-1	SPN, second byte of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 5	bits 8-6	SPN, 3 most significant bits (most significant at bit 8)	see 5.7.1.5
	bits 5-1	FMI (most significant at bit 5)	see 5.7.1.6
Byte: 6	bit 8	SPN Conversion Method	see 5.7.1.7
	bits 7-1	Occurrence Count	see 5.7.1.8

NOTE—When the occurrence count is not available it should be set to all ones.

EXAMPLE 1—The following illustrates the message format for when there are more than one diagnostic trouble code.

Given:

a=lamp status
b=SPN
c=FMI
d=CM and OC

Message form will be as follows: a,b,c,d,b,c,d,b,c,d,b,c,d....etc. In this example, the transport protocol of SAE J1939-21 will have to be used to send the information because it requires more than 8 data bytes. Actually any time there is more than one fault the services of the transport protocol will have to be used.

EXAMPLE 2—The following illustrates the message format for when a request of the DM1 is made and there are zero active faults. In order for one of the currently defined lamps (Malfunction Indicator Lamp, Red Stop Lamp, Amber Warning Lamp, and Protect Lamp) to be on, an active DTC must be in existence.

The original publication of this recommended practice defined that bytes 6 through 3 should be set to all ones when there are zero faults. This particular implementation is permitted but not preferred. Therefore, this is the Grandfathered Setting. Implementations are preferred to set bytes 6 through 3 to all zeros. This is the Recommended Setting.

Given:

Byte 1	bits 8-7	= 00
	bits 6-5	= 00
	bits 4-3	= 00
	bits 2-1	= 00
Byte 2	bits 8-7	= 11
	bits 6-5	= 11
	bits 4-3	= 11
	bits 2-1	= 11

		Grandfathered Setting		**Recommended Setting**
Byte 6-3	SPN	= 524,287	-Indicates not available	= 0
	FMI	= 31	-Indicates not available	= 0
	OC	= 127	-Indicates not available	= 0
	CM	= 1	-Indicates not available	= 0
Byte 7		= 255		= 255
Byte 8		= 255		= 255

EXAMPLE 3—Three cases are enumerated as follows to define the transmission rate requirements (see Figure 1.)

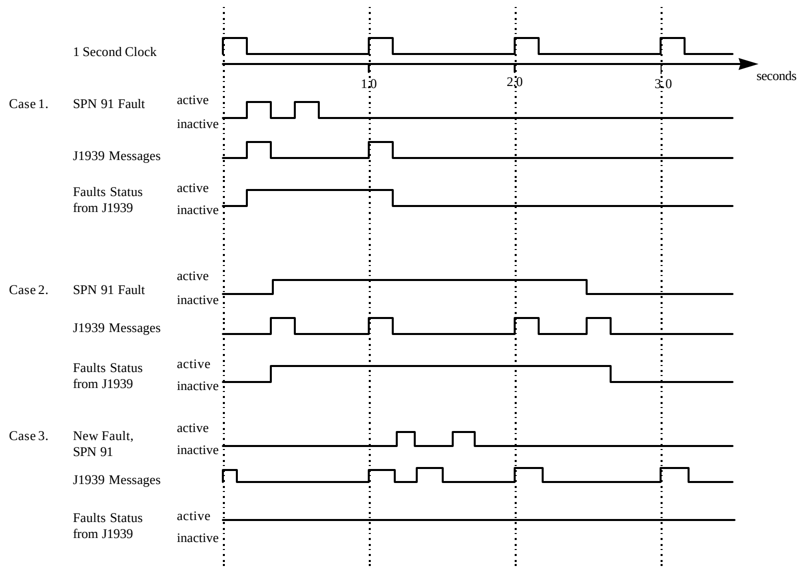


FIGURE 1 DEFINING THE TRANSMISSION RATE REQUIREMENTS

Case 1 illustrates that not every transition of a fault (active to inactive or inactive to active) results in a SAE J1939 message being sent. In this case, there are no other faults active when the example SPN 91 fault occurs. The SPN 91 fault is the Accelerator Pedal Position parameter which has an update faster than once a second. Therefore, the “SAE J1939 Message” (DM1 message) will be sent every 1 second while this fault is active. Three observations should be made. First, note that the first SAE J1939 message is sent when the “SPN 91 fault” becomes active on the first occurrence and not when it goes inactive for the first occurrence or active/inactive for the second occurrence. The inactive state is sent once at the next normal 1-second update (T=1 second). The second observation is that the “SAE J1939 Message” (DM1) is required to be sent at the 1 second interval even though the fault is no longer active and the actual DM1 message will contain no active faults. This is done as the action to show the fault went away. The way this is done for this specific case (where there are no longer any active faults) is as shown in the preceding Example 2. If there were other active faults they would have been sent in this message. The third observation is that if the second SPN 91 would have been a different SPN it would have been sent prior to the 1 second in a DM1 sent in between normal 1 second updates. The 1 second interval message would not contain this new SPN or SPN 91 assuming they both transitioned on and off before the 1 second message. Therefore, the 1 second DM1 message would still contain no faults.

Case 2 illustrates that the transition states can occur between the normal 1 second intervals. Therefore, a “SAE J1939 Message” is sent in between time equals 0 and time equals 1 to indicate that the SPN 91 fault has gone active. It is sent per the normal 1 second update at the 1 and 2 second points. It is sent at the time between 2 and 3 second to convey the transition to the inactive state. To do this the “J1939 Message” (DM1) is sent as shown in the preceding Example 2.

Case 3 shows the situation where there are already active faults in existence when SPN 91 becomes active. Note that the transition of SPN 91 to active state is sent between the 1 and 2 second points. The message contains all active faults, not just the new one. The transition to the inactive state is sent during the normal 2 second update. This message would contain all active faults and since SPN 91 went inactive it would not be in this message.

5.7.1.1 *Malfunction Indicator Lamp*—A lamp used to relay only emissions-related trouble code information. This lamp is only illuminated when there is an emission-related trouble code active.

00	Lamp Off
01	Lamp On
Type:	Status
Suspect Parameter Number:	1213
Reference:	5.7.1, 5.7.2, 5.7.6 and 5.7.12

5.7.1.2 *Red Stop Lamp*—This lamp is used to relay trouble code information that is of a severe enough condition that it warrants stopping the vehicle.

00	Lamp Off
01	Lamp On
Type:	Status
Suspect Parameter Number:	623
Reference:	5.7.1, 5.7.2, 5.7.6 and 5.7.12

5.7.1.3 *Amber Warning Lamp*—This lamp is used to relay trouble code information that is reporting a problem with the vehicle system but the vehicle need not be immediately stopped.

00	Lamp Off
01	Lamp On
Type:	Status
Suspect Parameter Number:	624
Reference:	5.7.1, 5.7.2, 5.7.6 and 5.7.12

5.7.1.4 *Protect Lamp*—This lamp is used to relay trouble code information that is reporting a problem with a vehicle system that is most probably not electronic subsystem related. For instance, engine coolant temperature is exceeding its prescribed temperature range.

00	Lamp Off
01	Lamp On
Type:	Status
Suspect Parameter Number:	987
Reference:	5.7.1, 5.7.2, 5.7.6 and 5.7.12

5.7.1.5 *Suspect Parameter Number*—This 19-bit number is used to identify the item for which diagnostics are being reported. The SPN is used for multiple purposes, some of those that are specific to diagnostics are: 1. to identify a least repairable subsystem that has failed; 2. to identify subsystems and or assemblies that may not have hard failures but may be exhibiting abnormal operating performance; 3. identifying a particular event or condition that will be reported; and 4. to report a component and non-standard failure mode. SPNs are assigned to each individual parameter in a Parameter Group and to items that are relevant to diagnostics but are not a parameter in a Parameter Group. SPNs are independent of the source address for the message. However, the source address may be necessary to determine which controller on the network performed the diagnosis.

The first 511 SPNs are reserved and will be assigned the exact same number as the Parameter Identifier (PID) used in SAE J1587. That is, the SPN for an accelerator problem will be reported as SPN 91 which is SAE J1587 PID 91. All other SPNs will be numbered sequentially starting at 512 and incrementing by one for each new assignment. Refer to SAE J1939 Appendix C.

Data Length:	19 bits
Resolution:	1 SPN/bit

Data Range:	0 to 524,287
Type:	Status
Suspect Parameter Number:	1214
Reference:	5.7.1, 5.7.2, 5.7.4, 5.7.6, and 5.7.12

5.7.1.6 *Failure Mode Identifier*—The FMI defines the type of failure detected in the subsystem identified by an SPN. Note that the failure may not be an electrical failure but may instead be a subsystem failure or condition needing to be reported to the service technician and maybe also to the operator. Conditions can include system events or status that need to be reported. The FMI, SPN, SPN Conversion Method and Occurrence Count fields combine to form a given diagnostic trouble code. The "Reserved for SAE Assignment" FMIs will be assigned by the SAE J1939 Control and Communications Subcommittee if additional failure modes become necessary. The currently defined FMIs are listed in Appendix A.

Data Length:	5 bits
Resolution:	1 FMI/bit
Data Range:	0 to 31
Type:	Status
Suspect Parameter Number:	1215
Reference:	5.7.1, 5.7.2, 5.7.4, 5.7.6, and 5.7.12

5.7.1.7 *SPN Conversion Method*—When this 1-bit field is equal to a zero, the SPN should be converted as it is defined in this document (see definition below for Version 4). The February 1996 version of J1939-73 contained inadequate definitions to assure consistent implementations. Products implementing to February 1996 version of the document will always have this bit set to a one. When this is the case, the SPN is in either Version 1, 2 or 3 format.

To clarify the ordering of bits and bytes within the SPN parameter (which is 19 bits long) and to keep that ordering consistent with other parameters in J1939-71 and J1939-73, the bit order has been respecified. See Version 4 below for the recommended formatting.

To reduce problems in interpretation of the SPNs the bit between the FMI field and the Occurrence Count field, previously reserved, will be cleared to zero to identify use of the currently specified SPN bit pattern. This bit now comprises an SPN Conversion Method for the purpose of maintaining usability of those implementations that are already in use.

Data Length:	1 bit
Resolution:	Not Applicable
Data Range:	0 means convert SPNs per the Version 4 definition below 1 means convert SPNs per Version 1, 2 or 3 specified below. The four versions of interpretation are: 1. SPN assumed to be sent most significant bit first 2. SPN represented as Intel format for most significant 16 bits with 3 least significant bits of 19 bits in with FMI value. 3. SPN represented as Intel format for all 19 bits (least significant sent first) 4. SPN represented as Intel format for all 19 bits with the SPN Conversion Method set to 0.
Type:	Status
Suspect Parameter Number:	1706
Reference:	5.7.1, 5.7.2, 5.7.4, 5.7.6 and 5.7.12

Given:

SPN 1208	$= 4B8_{16}$	$= 000\ 00000100\ 10111000_2$ (19 bits)
FMI 3	$= 3_{16}$	$= 00011_2$ (5 bits)
OC 10	$= A_{16}$	$= 0001010_2$ (7 bits)
CM		$= 0_2$ (1 bit)

Version 1.

	DTC																															
	Byte 3 8 most significant bits of 16 most significant bits of SPN (bit 8 most significant)								Byte 4 8 least significant bits of 16 most significant bits of SPN (bit 8 most significant)								Byte 5 3 least significant bits of SPN and the FMI (bit 8 SPN msb and bit 5 FMI msb)								Byte 6							
	SPN																			FMI					CM	OC						
J1939 Frame Format	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	0

Version 2.

	DTC																															
	Byte 3 8 least significant bits of 16 most significant bits of SPN (bit 8 most significant)								Byte 4 8 most significant bits of 16 most significant bits of SPN (bit 8 most significant)								Byte 5 3 least significant bits of SPN and the FMI (bit 8 SPN msb and bit 5 FMI msb)								Byte 6							
	SPN																			FMI					CM	OC						
J1939 Frame Format	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
	1	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	0

Version 3.

	DTC																															
	Byte 3 8 least significant bits of SPN (bit 8 most significant)								Byte 4 second byte of SPN (bit 8 most significant)								Byte 5 3 most significant bits of SPN and the FMI (bit 8 SPN msb and bit 5 FMI msb)								Byte 6							
	SPN																			FMI					CM	OC						
J1939 Frame Format	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
	1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	0

Version 4.
Recommended Version

	DTC																															
	Byte 3 8 least significant bits of SPN (bit 8 most significant)								Byte 4 second byte of SPN (bit 8 most significant)								Byte 5 3 most significant bits of SPN and the FMI (bit 8 SPN msb and bit 5 FMI msb)								Byte 6							
	SPN																			FMI					CM	OC						
J1939 Frame Format	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
	1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	1	0

5.7.1.8 *Occurrence Count*—The 7-bit occurrence count field contains the number of times a fault has gone from active to previously active. If an occurrence count is not available, then this field should be set to all ones.

Data Length:	7 bits
Resolution:	1 occurrence count/bit
Data Range:	0 to 126 (The value 127 is reserved for indicating *not available*)
Type:	Status
Suspect Parameter Number:	1216
Reference:	5.7.1, 5.7.2, 5.7.4, 5.7.6 and 5.7.12

5.7.2 PREVIOUSLY ACTIVE DIAGNOSTIC TROUBLE CODES (DM2)—The information communicated is limited to the previously active trouble codes. It is used to notify other components on the network of the diagnostic condition of the transmitting electronic component. The data contains a list of diagnostic codes and occurrence counts for previously active trouble codes. Whenever this message is sent, it should contain all previously active trouble codes with an occurrence count not equal to zero. Note that this Parameter Group will be sent using the “Multipacket Transport” Parameter Group as specified in SAE J1939-21 when applicable.

Transmission Rate:		On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:		Variable	
Data page:		0	
PDU Format:		254	
PDU Specific:		203	
Default Priority:		6	
Parameter Group Number:		65227 ($00FECB_{16}$)	
Byte: 1	bits 8-7	Malfunction Indicator Lamp Status	see 5.7.1.1
	bits 6-5	Red Stop Lamp Status	see 5.7.1.2
	bits 4-3	Amber Warning Lamp Status	see 5.7.1.3
	bits 2-1	Protect Lamp Status	see 5.7.1.4
Byte: 2	bits 8-7	Reserved for SAE assignment Lamp Status	
	bits 6-5	Reserved for SAE assignment Lamp Status	
	bits 4-3	Reserved for SAE assignment Lamp Status	
	bits 2-1	Reserved for SAE assignment Lamp Status	
Byte: 3	bits 8-1	SPN, 8 least significant bits of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 4	bits 8-1	SPN, second byte of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 5	bits 8-6	SPN, 3 most significant bits (most significant at bit 8)	see 5.7.1.5
	bits 5-1	FMI (most significant at bit 5)	see 5.7.1.6
Byte: 6	bit 8	SPN Conversion Method	see 5.7.1.7
	bits 7-1	Occurrence Count	see 5.7.1.8

NOTE—When the occurrence count is not available it should be set to all ones.

EXAMPLE 1—The following illustrates the message format for when there are more than one diagnostic trouble code.

Given:

a=lamp status (LS)
b=SPN
c=FMI
d=CM and OC

Message form will be as follows: a,b,c,d,b,c,d,b,c,d,b,c,d....etc. In this example, the transport protocol of SAE J1939-21 will have to be used to send the information because it requires more than 8 data bytes. Actually any time there is more than one fault the services of the transport protocol will have to be used.

EXAMPLE 2—The following illustrates the message format for when a request of the DM2 is made and there are zero previously active faults. The currently defined lamps (Malfunction Indicator Lamp, Red Stop Lamp, Amber Warning Lamp, and Protect Lamp) should reflect the present state of the transmitting electronic component. In this example, the amber lamp is identified as being on.

The original publication of this recommended practice defined that bytes 6 through 3 should be set to all ones when there are zero faults. This particular implementation is permitted but not preferred. Therefore, this is the Grandfathered Setting. Implementations are preferred to set bytes 6 through 3 to all zeros. This is the Recommended Setting.

Given:

Byte 1	bits 8-7	= 00
	bits 6-5	= 00
	bits 4-3	= 01
	bits 2-1	= 00
Byte 2	bits 8-7	= 11
	bits 6-5	= 11
	bits 4-3	= 11
	bits 2-1	= 11

		Grandfathered Setting		**Recommended Setting**
Byte 6-3	SPN	= 524,287	-Indicates not available	= 0
	FMI	= 31	-Indicates not available	= 0
	OC	= 127	-Indicates not available	= 0
	CM	= 1	-Indicates not available	= 0
Byte 7		= 255		= 255
Byte 8		= 255		= 255

5.7.3 DIAGNOSTIC DATA CLEAR/RESET OF PREVIOUSLY ACTIVE DTCS (DM3)—All of the diagnostic information pertaining to the previously active trouble codes should be erased when this PG is requested. The diagnostic data associated with active trouble codes will not be affected. Upon the completion of this operation, a Positive Acknowledgment is required (see SAE J1939-21 PGN 59392). If for some reason a device can not perform the requested action, then it is required to send a Negative Acknowledgement (see SAE J1939-21 PGN 59392).

Transmission Rate:	On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:	0	
Data page:	0	
PDU Format:	254	
PDU Specific:	204	
Default Priority:	6	
Parameter Group Number:	65228 ($00FECC_{16}$)	

5.7.4 FREEZE FRAME PARAMETERS (DM4)—A freeze frame is defined as the list of recorded parameters at the time a diagnostic trouble code was captured. The freeze frame recorded for each diagnostic trouble code will contain the required parameters first and then any manufacturer specific information. It is possible that controllers will have more than one freeze frame available and each may have some manufacturer specific information. A freeze frame is specific to one diagnostic trouble code and one diagnostic trouble code only has one freeze frame. This then limits the amount of freeze frame data per fault and for all faults that are included in this message to 1785 bytes (see SAE J1939-21 transport protocol).

This diagnostic message is best suited for systems which may impact emissions and or be powertrain related. However, the use of this message is not limited to just emission-related failures or just powertrain devices. It can be used to report non-emission related or non-powertrain related failures.

Transmission Rate:		On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:		Variable	
Data page:		0	
PDU Format:		254	
PDU Specific:		205	
Default Priority:		6	
Parameter Group Number:		65229 (00FECD$_{16}$)	
Byte: 1		Freeze Frame Length	see 5.7.4.1
Byte: 2	bits 8-1	SPN, 8 least significant bits of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 3	bits 8-1	SPN, second byte of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 4	bits 8-6	SPN, 3 most significant bits (most significant at bit 8)	see 5.7.1.5
	bits 5-1	FMI (most significant at bit 5)	see 5.7.1.6
Byte: 5	bit 8	SPN Conversion Method	see 5.7.1.7
	bits 7-1	Occurrence Count	see 5.7.1.8
Byte: 6		Engine Torque Mode	see SAE J1939-71
Byte: 7		Boost	see SAE J1939-71
Byte: 9-8		Engine Speed (MSB, 31.5 rpm/bit)	see SAE J1939-71
Byte: 10		Engine % Load	see SAE J1939-71
Byte: 11		Engine coolant temperature	see SAE J1939-71
Byte: 13-12		Vehicle Speed (MSB, 0.62 MPH/bit)	see SAE J1939-71
Byte: n-14		Manufacturer Specific information	

NOTE—When the occurrence count is not available it should be set to all ones.

NOTE—If no DTCs (active or previously active) have been accumulated, then the response will be:

PGN		= 65229
Byte:	1	= 0
	5-2	= 0
	6	= 255
	7	= 255
	8	= 255

When byte 1 is equal to zero identifies to the receiver that the other parameters in the message should not be interpreted. Also notice that the values of the information put in bytes 1 through 5 are zero even though some of the parameters may have normally been set to all ones (binary) to indicate not available.

EXAMPLE—The following illustrates the message format for when there are more than one freeze frame.

Given:

a=freeze frame length
b=required parameters (bytes 2 through 13 for the first DTC and the corresponding bytes for each of the remaining DTCs)
c=manufacturer specific freeze frame information

Message form will be as follows: a,b,c,a,b,c,a,b,c,a,b,c,a,b,c....etc. The transport protocol of SAE J1939-21 will have to be used to send freeze frames because they are more than 8 data bytes.

5.7.4.1 *Freeze Frame Length*—The Freeze Frame Length shall be equal to the number of bytes in the required parameters (that is bytes 2 through 13) plus the number of bytes in the manufacturer specific parameters. That is: a = b + c

Data Length:	8 bits
Resolution:	1 byte/bit
Data Range:	0 to 255
Type:	Status
Suspect Parameter Number:	1217
Reference:	5.7.4

EXAMPLE

b = 12
c = 2..........oil pressure, intake manifold temperature
a = b + c
a = 12 + 2 = 14

5.7.4.2 *Freeze Frame Parameters*—The parameters collected in the Freeze Frame shall use the same scaling as is defined in the SAE J1939-71 document.

5.7.5 DIAGNOSTIC READINESS (DM5)—Reports the diagnostics information that relates to diagnostic readiness.

Transmission Rate:	On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:	Variable (presently 8 bytes)	
Data page:	0	
PDU Format:	254	
PDU Specific:	206	
Default Priority:	6	
Parameter Group Number:	65230 ($00FECE_{16}$)	
Byte: 1	Active Trouble Codes	see 5.7.5.1
2	Previously Active Diagnostic Trouble Codes	see 5.7.5.2
3	OBD Compliance	see 5.7.5.3
4	Continuously Monitored Systems Support/Status	see 5.7.5.4
6-5	Non-continuously Monitored Systems Support	see 5.7.5.5
8-7	Non-continuously Monitored Systems Status	see 5.7.5.6

NOTE—The transport protocol of SAE J1939-21 may in the future have to be used to send this PG because it may then require more than 8 data bytes.

5.7.5.1 *Active Trouble Codes*—Identifies the number of active trouble codes that are present in a specific controller. If no DTCs are active, this field should be set to zero.

Data Length:	1 byte
Resolution:	1 trouble code/bit
Data Range:	0 to 240
Type:	Measured
Suspect Parameter Number:	1218
Reference:	5.7.5

5.7.5.2 *Previously Active Diagnostic Trouble Codes*—Identifies the number of previously active trouble codes that are present in a specific controller. If no DTCs have been previously active, this field should be set to zero.

Data Length:	1 byte
Resolution:	1 trouble code/bit
Data Range:	0 to 240
Type:	Measured
Suspect Parameter Number:	1219
Reference:	5.7.5

5.7.5.3 *OBD Compliance*—Identifies the OBD compliance capability of the responding controller. Identifies the requirements level to which the controller was built.

Data Length:	1 byte
Resolution:	See below
Data Range:	0 to 240
Type:	Measured
Suspect Parameter Number:	1220
Reference:	5.7.5

Value	**Description**
00	Reserved for SAE Assignment
01	OBD II (California, ARB)
02	OBD (Federal, EPA)
03	OBD and OBD II
04	OBD I
05	Not intended to meet OBD II requirements
06-240	Reserved for SAE Assignment

5.7.5.4 *Continuously Monitored Systems Support/status*—Identifies the continuously monitored system support and status.

Data Length:	1 byte
Resolution:	See below
Data Range:	Bit mapped, see below
Type:	Measured
Suspect Parameter Number:	1221
Reference:	5.7.5

Byte	**Bit**	**Description**
4	8	Reserved for SAE Assignment
	7	Comprehensive component monitoring status
	6	Fuel System monitoring status
	5	Misfire monitoring status
	Where each status bit (bits 7, 6, 5) is interpreted:	
		0 = test complete, not supported
		1 = test not complete
	4	Reserved for SAE Assignment
	3	Comprehensive component monitoring support
	2	Fuel system monitoring support
	1	Misfire monitoring support
	Where each supported bit (bits 3, 2, 1) is interpreted:	
		0 = test not supported by this controller
		1 = test supported by this controller

NOTE—Notice that a bit set to zero can mean test not supported. This is different than the typical J1939 use of the value 1 to indicate not available.

5.7.5.5 *Non-continuously Monitored Systems Support*—Identifies the non-continuously monitored systems support.

Data Length:	2 bytes (sent as a magnitude; therefore it is byte swapped)
Resolution:	See below
Data Range:	Bit mapped, see below
Type:	Measured
Suspect Parameter Number:	1222
Reference:	5.7.5

Byte	**Bit**	**Description**
5	8	EGR system monitoring Support
	7	Oxygen sensor heater monitoring Support
	6	Oxygen sensor monitoring Support
	5	A/C system refrigerant monitoring Support
	4	Secondary air system monitoring Support
	3	Evaporative system monitoring Support
	2	Heated catalyst monitoring Support
	1	Catalyst monitoring Support
6	8-2	Reserved for SAE Assignment
	1	Cold start aid system monitoring Support
	Where each bit is interpreted:	
		0 = test not supported by this controller
		1 = test supported by this controller

NOTE—The "Non-continuously Monitored Systems Support" parameter is in the Intel Format (byte-swapped format). Also notice that a bit set to zero means test not supported. This is different than the typical J1939 use of the value 1 to indicate not available.

5.7.5.6 *Non-continuously Monitored Systems Status*—Identifies the non-continuously monitored systems status. Each bit identifies whether a particular test is complete for a given controller.

Data Length:	2 bytes (sent as a magnitude; therefore it is byte swapped)
Resolution:	See below
Data Range:	Bit mapped, see below
Type:	Measured
Suspect Parameter Number:	1223
Reference:	5.7.5

Byte	**Bit**	**Description**
5	8	EGR system monitoring Status
	7	Oxygen sensor heater monitoring Status
	6	Oxygen sensor monitoring Status
	5	A/C system refrigerant monitoring Status
	4	Secondary air system monitoring Status
	3	Evaporative system monitoring Status
	2	Heated catalyst monitoring Status
	1	Catalyst monitoring Status
6	8-2	Reserved for SAE Assignment
	1	Cold start aid system monitoring Status

Where each bit is interpreted:

0 = test complete, or not supported
1 = test not complete

NOTE—The "Non-continuously Monitored Systems Status" parameter is in the Intel Format (byte-swapped format). Also notice that a bit set to zero can mean test not supported. This is different than the typical J1939 use of the value 1 to indicate not available.

5.7.6 CONTINUOUSLY MONITORED SYSTEMS TEST RESULTS (DM6)—The purpose of this PG is to enable the off-board test device to obtain test results for emission-related powertrain components/systems that are continuously monitored during normal driving conditions. The intended use of this data is to assist the service technician after a vehicle repair, and after clearing diagnostic information, by reporting test results after a single driving cycle. If the test fails during the driving cycle, the DTC associated with that test is reported. Test results reported by this mode do not necessarily indicate a faulty component/system. If test results indicate a failure after additional driving, then the MIL is illuminated and a DTC is set and reported with PGN 65226.

Reporting the Continuously Monitored Systems Test Results is done using the same format as is used to report active DTCs.

Transmission Rate:	On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:	Variable	
Data page:	0	
PDU Format:	254	
PDU Specific:	207	
Default Priority:	6	

Parameter Group Number:		65231 ($00FECF_{16}$)	
Byte: 1	bits 8-7	Malfunction Indicator Lamp Status	see 5.7.1.1
	bits 6-5	Red Stop Lamp Status	see 5.7.1.2
	bits 4-3	Amber Warning Lamp Status	see 5.7.1.3
	bits 2-1	Protect Lamp Status	see 5.7.1.4
Byte: 2	bits 8-7	Reserved for SAE assignment Lamp Status	
	bits 6-5	Reserved for SAE assignment Lamp Status	
	bits 4-3	Reserved for SAE assignment Lamp Status	
	bits 2-1	Reserved for SAE assignment Lamp Status	
Byte: 3	bits 8-1	SPN, 8 least significant bits of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 4	bits 8-1	SPN, second byte of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 5	bits 8-6	SPN, 3 most significant bits (most significant at bit 8)	see 5.7.1.5
	bits 5-1	FMI (most significant at bit 5)	see 5.7.1.6
Byte: 6	bit 8	SPN Conversion Method	see 5.7.1.7
	bits 7-1	Occurrence Count	see 5.7.1.8

NOTE—When the occurrence count is not available it should be set to all ones.

EXAMPLE 1—The following illustrates the message format for when there are more than one diagnostic trouble code.

Given:

a=lamp status (LS)
b=SPN
c=FMI
d=CM and OC

Message form is as follows: a,b,c,d,b,c,d,b,c,d,b,c,d....etc. In this example, the transport protocol of SAE J1939-21 has to be used to send the information because it requires more than 8 data bytes. Actually any time there is more than one fault, the services of the transport protocol have to be used.

EXAMPLE 2—The following illustrates the message format for when a request of the DM6 is made and all test results indicate no trouble information. The currently defined lamps (Malfunction Indicator Lamp, Red Stop Lamp, Amber Warning Lamp, and Protect Lamp) should reflect the present state of the transmitting electronic component. In this example, the amber lamp is identified as being on.

The original publication of this recommended practice defined that bytes 6 through 3 should be set to all ones when there are zero faults. This particular implementation is permitted but not preferred. Therefore, this is the Grandfathered Setting. Implementations are preferred to set bytes 6 through 3 to all zeros. This is the Recommended Setting.

Given:

Byte 1	bits 8-7 =	00
	bits 6-5 =	00
	bits 4-3 =	01
	bits 2-1 =	00
Byte 2	bits 8-7 =	11
	bits 6-5 =	11
	bits 4-3 =	11

bits 2-1 = 11

		Grandfathered Setting		**Recommended Setting**
Byte 6-3	SPN	= 524,287	-Indicates not available	= 0
	FMI	= 31	-Indicates not available	= 0
	OC	= 127	-Indicates not available	= 0
	CM	= 1	-Indicates not available	= 0
Byte 7		= 255		= 255
Byte 8		= 255		= 255

5.7.7 COMMAND NON-CONTINUOUSLY MONITORED TEST (DM7)—The purpose of this command in the diagnostic process is to provide the ability to command on-board diagnostic monitoring tests of specific components/ systems that are not continuously monitored.

The component manufacturer is responsible to assign test identifiers and component identifiers for tests of different systems and components. PGN 58112 (DM7) is used to invoke one of the manufacturer defined test identifiers. Test results are reported by test identifier using PGN 65232 (DM8). If DM7 or the specific test identifier is not supported, then a NACK is required—SAE J1939-21 PGN 59392) to be returned.

Transmission Rate:	Sent whenever a test is desired	
Data Length:	8	
Data page:	0	
PDU Format:	227	
PDU Specific:	Destination Address	
Default Priority:	6	
Parameter Group Number:	58112 ($00E300_{16}$)	
Byte: 1	Test Identifier	see 5.7.7.1
8-2	Reserved for SAE Assignment	

5.7.7.1 *Test Identifier*—Designates the test to be run. These identifiers are manufacturer-defined test identifiers. There are 64 valid test identifiers, 1 to 64.

Data Length:	1 byte
Resolution:	See 5.7.8
Data Range:	1 to 64 (Note: 0 and 65 to 250 are reserved.)
Type:	Status
Suspect Parameter Number:	1224
Reference:	5.7.7 and 5.7.8

5.7.8 TEST RESULTS FOR NON-CONTINUOUSLY MONITORED SYSTEMS (DM8)—The purpose of this response PG is to report the test results for one of the non-continuously monitored tests invoked using DM7. The component manufacturer is responsible to assign test identifiers and component identifiers for tests of different systems and components. PGN 58112 (DM7) is used to invoke one of the manufacturer defined test identifiers. Test results are reported by test identifier using PGN 65232 (DM8).

Transmission Rate:	Sent in response to PGN 58112 when the results are available A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	
Data Length:	Variable but sent in 8 byte sets (see example)	
Data page:	0	
PDU Format:	254	
PDU Specific:	208	
Default Priority:	6	
Parameter Group Number:	65232 ($00FED0_{16}$)	
Byte: 1	Test Identifier	see 5.7.7.1

2	Test Type/Component Identifier	see 5.7.8.1
4-3	Test Value	see 5.7.8.2
6-5	Test Limit Maximum	see 5.7.8.3
8-7	Test Limit Minimum	see 5.7.8.4

Further guidelines for the use of the Test Value, Test Limit Maximum and the Test Limit Minimum to convey results for tests are enumerated in the Table 3 below. For example with a test where there is not a test limit maximum or test limit minimum the results are determined from the test value alone (cases 1 to 4).

TABLE 3

Case #	Test Value	Test Maximum	Test Minimum	Interpretation
1.	0 to 64255	$FFFF_{16}$	$FFFF_{16}$	Test Pass
2.	$FE00_{16}$ (Error)	$FFFF_{16}$	$FFFF_{16}$	Test Fail
3.	$FB00_{16}$	$FFFF_{16}$	$FFFF_{16}$	Test Not Complete
4.	$FB01_{16}$	$FFFF_{16}$	$FFFF_{16}$	Test Can Not Be Performed
5.	0003	0004	0001	Test Pass
6.	0000	0004	0001	Test Fail
7.	0005	0004	0001	Test Fail
8.	0000	$FFFF_{16}$	0001	Test Fail
9.	0002	$FFFF_{16}$	0001	Test Pass
10.	$FAFF_{16}$	$FAFE_{16}$	$FFFF_{16}$	Test Fail
11.	$AF57_{16}$	$AF59_{16}$	$FFFF_{16}$	Test Pass

Another observation is that if there are more than one test value to report on a given test then the results will be sent using the transport protocol defined in J1939-21. See the following example. If multiple test results are reported, then the component identifier parameter is used to distinguish the different result values. All test identifier values must be the same when multiple test results are reported in one DM8 response.

EXAMPLE:

Given: Assume three separate Test Values are desired to be communicated.

Where: a = Test Identifier, b = Test Type/Component Identifier, c = Test Value,
d = Test Limit Maximum, e = Test Limit Minimum

Message form will be as follows: a,b,c,d,e,a,b,c,d,e,a,b,c,d,e

The transport protocol of SAE J1939-21 will have to be used when there are more than one Test Value to send because 16 or more data bytes would be required. In this example 24 bytes of data would need sent for three Test Values.

5.7.8.1 Test Type/Component Identifier—Identifies the non-continuously monitored component identifier that was tested. These component identifiers are defined by the manufacturer. They are necessary when multiple components or systems are present on the vehicle and have the same definition of test identifier.

Data Length:	1 byte
Resolution:	See below
Data Range:	1 to 64 (Note: 0 and 65 to 250 are reserved.)
Type:	Measured
Suspect Parameter Number:	1225
Reference:	5.7.8

5.7.8.2 Test Value—The test value collected during the test. If the test performed does not have both a test limit minimum and maximum, then the appropriate limit value (Maximum or Minimum) should be set to all ones.

SAE J1939-71 defines this to mean not available.

Data Length:	2 bytes
Resolution:	Not defined
Data Range:	0 to 64255
Type:	Measured
Suspect Parameter Number:	1226
Reference:	5.7.8

5.7.8.3 Test Limit Maximum—The threshold which the test value must be below to pass the test.

Data Length:	2 bytes
Resolution:	Not defined
Data Range:	0 to 64255
Type:	Measured
Suspect Parameter Number:	1227
Reference:	5.7.8

5.7.8.4 Test Limit Minimum—The threshold which the test value must be above to pass the test.

Data Length:	2 bytes
Resolution:	Not defined
Data Range:	0 to 64255
Type:	Measured
Suspect Parameter Number:	1228
Reference:	5.7.8

5.7.9 OXYGEN SENSOR TEST RESULTS (DM9)—To be determined in later revisions of this document.

5.7.10 NON-CONTINUOUSLY MONITORED SYSTEMS TEST IDENTIFIERS SUPPORT (DM10)—The purpose of this PG is to report the list of non-continuously monitored systems tests supported by the controller. The component manufacturer is responsible to assign test identifiers and component identifiers for tests of different systems and components. PGN 58112 (DM7) is used to invoke one of the manufacturer-defined test identifiers. Test results are reported by test identifier using PGN 65232 (DM8). Service tools can determine the supported tests by requesting PGN 65234 (DM10).

Transmission Rate:	On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:	8 (this is not byte swapped)	see 5.7.10.1
Data page:	0	
PDU Format:	254	
PDU Specific:	210	
Default Priority:	6	
Parameter Group Number:	65234 ($00FED2_{16}$)	
Byte: 8-1	Test Identifiers Supported	see 5.7.10.1

5.7.10.1 Test Identifiers Supported—Tests Identifiers Supported—Indicates the test identifiers that the controller supports. Each bit is assigned to one test. Therefore, we can have up to 64 tests without having to use the transport protocol of SAE J1939-21. The assignment of a given test identifier to a given bit is manufacturer specific.

Data Length:	8 bytes
Resolution:	See below

Data Range: 64 bits
Note: Bit mapped, each bit indicates an individual test identifier
Type: Measured
Suspect Parameter Number: 1229
Reference: 5.7.10

Byte	Bit	Description
1	8	Test one
	7	Test two
	6	Test three
	5	Test four
	4	Test five
	3	Test six
	2	Test seven
	1	Test eight
2	8	Test nine
2-8	64-10	Manufacturer assigned test 10 through 64

Where each bit is interpreted:
0 = test not supported
1 = test supported

See Table 4 for an example:

TABLE 4 EXAMPLE — USE OF TEST IDENTIFIERS SUPPORTED

Test Identifier Representations	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1. Binary	00000100	00000001	00000000	00000101	10100000	00000000	00000000	00000001
2. Hex	04	01	00	05	A0	00	00	01
3. Test Identifiers	6	16		30, 32	33, 35			64

5.7.11 DIAGNOSTIC DATA CLEAR/RESET FOR ACTIVE DTCS (DM11)—All of the diagnostic information pertaining to the active diagnostic trouble codes should be erased. Sent as a request whenever the service Tool wishes to clear/reset diagnostic data for active DTCs. This is expected to occur once the problem has been corrected. Upon the completion of this operation, a Positive Acknowledgment is required (see SAE J1939-21 PGN 59392). If for some reason a device can not perform the requested action, then it is required to send a Negative Acknowledgement (see SAE J1939-21 PGN 59392).

Transmission Rate: On request using PGN 59904 see SAE J1939-21
A NACK is required if PG is not supported
(see SAE J1939-21 PGN 59392)
Data Length: 0
Data page: 0
PDU Format: 254
PDU Specific: 211
Default Priority: 6
Parameter Group Number: 65235 ($00FED3_{16}$)

5.7.12 EMISSIONS-RELATED ACTIVE DIAGNOSTIC TROUBLE CODES (DM12)—The information communicated is limited to the currently active emission-related diagnostic trouble codes preceded by the diagnostic lamp status. Both are used to notify other components on the network of the diagnostic condition of the transmitting electronic component. The data contains the lamp status and a list of diagnostic codes and occurrence counts for currently active emission-related diagnostic trouble codes.

Transmission Rate:		On request using PGN 59904 A NACK is required if PG is not supported (see SAE J1939-21 PGN 59392)	see SAE J1939-21
Data Length:		Variable	
Data page:		0	
PDU Format:		254	
PDU Specific:		212	
Default Priority:		6	
Parameter Group Number:		65236 ($00FED4_{16}$)	
Byte: 1	bits 8-7	Malfunction Indicator Lamp Status	see 5.7.1.1
	bits 6-5	Red Stop Lamp Status	see 5.7.1.2
	bits 4-3	Amber Warning Lamp Status	see 5.7.1.3
	bits 2-1	Protect Lamp Status	see 5.7.1.4
Byte: 2	bits 8-7	Reserved for SAE assignment Lamp Status	
	bits 6-5	Reserved for SAE assignment Lamp Status	
	bits 4-3	Reserved for SAE assignment Lamp Status	
	bits 2-1	Reserved for SAE assignment Lamp Status	
Byte: 3	bits 8-1	SPN, 8 least significant bits of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 4	bits 8-1	SPN, second byte of SPN (most significant at bit 8)	see 5.7.1.5
Byte: 5	bits 8-6	SPN, 3 most significant bits (most significant at bit 8)	see 5.7.1.5
	bits 5-1	FMI (most significant at bit 5)	see 5.7.1.6
Byte: 6	bit 8	SPN Conversion Method	see 5.7.1.7
	bits 7-1	Occurrence Count	see 5.7.1.8

NOTE—When the occurrence count is not available it should be set to all ones.

EXAMPLE 1—The following illustrates the message format for when there are more than one diagnostic trouble code.

Given:

a=lamp status
b=SPN
c=FMI
d=CM and OC

Message form is as follows: a,b,c,d,b,c,d,b,c,d,b,c,d....etc. In this example, the transport protocol of SAE J1939-21 has to be used to send the information because it requires more than 8 data bytes. Actually any time there is more than one fault the services of the transport protocol have to be used.

EXAMPLE 2—The following illustrates the message format for when a request of the DM12 is made and there are zero active emissions faults. Note that the Malfunction Indicator Lamp is off while any of the other three —Red Stop Lamp, Amber Warning Lamp, and Protect Lamp) could be on. In this example, all three are on.

The original publication of this recommended practice defined that bytes 6 through 3 should be set to all ones when there are zero faults. This particular implementation is permitted but not preferred. Therefore, this *is the Grandfathered Setting. Implementations are preferred to set bytes 6 through 3 to all zeros. This is the Recommended Setting.*

Given:

Byte 1

bits 8-7	= 00
bits 6-5	= 01
bits 4-3	= 01
bits 2-1	= 01

Byte 2

bits 8-7	= 11
bits 6-5	= 11
bits 4-3	= 11
bits 2-1	= 11

				Grandfathered Setting	Recommended Setting
Bytes 6-3					
	SPN	= 524,287	-Indicates not available		= 0
	FMI	= 31	-Indicates not available		= 0
	OC	= 127	-Indicates not available		= 0
	CM	= 1	-Indicates not available		= 0
Byte 7	=	255			= 255
Byte 8	=	255			= 255

5.7.13 STOP START BROADCAST (DM13)—This message is used to stop or start broadcast messages. These broadcast messages may be on networks other than SAE J1939.

The following notes help to clarify to use of this command PGN.

1. This command shall only be initiated when the vehicle is at zero kilometers/hour and at zero engine rpm.

2. All nodes shall “power-up” in their normal broadcasting mode. Therefore, if any node was “power-down”, while in a “Stop Broadcast” condition, it would revert to its normal operation on power-up.

3. This is not a message to ignore all communications. It is a message to minimize network traffic. It is recognized that some network messages may be required to continue even during the “Stop Broadcast” condition. If an unsafe or undesirable vehicle operating condition would result from the lack of normal messages then this mode would cause all nonessential messages to be inhibited.

4. Requests that are generated during the “Stop Broadcast” state should be responded to. However, devices that may be programmed to periodically issue requests should postpone these requests until the “Stop Broadcast” state is exited.

5. All devices that have been told to change state, plus those nodes that may be affected by the absence of broadcast messages could look for the “Hold Signal” as a plausible explanation for why the information is missing. In addition all devices that have been told to change state shall monitor the “Hold Signal”. If the “Hold Signal” disappears for 6 seconds then all applicable nodes shall revert back to the normal state.

6. Diagnostic Trouble Codes should not be recorded for failed communications due to broadcast PGNs missing during the modified Broadcast state. Network devices should look for the Hold signal to be absent for more than 6 seconds before recording any applicable Diagnostic Trouble Code.

7. When this command is used to disable broadcasts of information on other networks it could result in

diagnostic trouble codes being reported about this situation. Therefore, it is recommended that the use of this Stop/Start broadcast command be used with caution.

One of the uses for the “Stop Start Broadcast PG” is to reduce network traffic during certain diagnostic procedures. As an example while calibrating a control module, the diagnostic Tool will likely want to stop the normal broadcasts of all network devices keeping in mind the comments made in the notes section above. Another use is that it allows the diagnostic Tool to potentially emulate a remote device during a diagnostic procedure. In this case the diagnostic Tool could generate the messages that the remote device would normally generate.

Transmission Rate:	Sent whenever a Stop or Start broadcast event is necessary. To maintain the modified state of the vehicle network(s) the commanding device must send the Hold Signal once every 5 seconds. A NACK is required if PG is not supported (see J1939-21 PGN 59392). Note that the NACK is only provided if PGN 57088 is directed to a specific destination address.
Data Length:	8
Data Page:	0
PDU Format:	223
PDU Specific:	DA
Default Priority:	6
Parameter Group Number:	57088 ($00DF00_{16}$)

Stop Start Broadcast[1]

Byte: 1	bits 8-7	Current Data Link	see 5.7.13.1
	bits 6-5	J1587	see 5.7.13.2
	bits 4-3	J1922	see 5.7.13.3
	bits 2-1	J1939 Network #1, Primary vehicle network	see 5.7.13.4
Byte: 2	bits 8-7	J1939 Network #2	see 5.7.13.5
	bits 6-5	ISO 9141	see 5.7.13.6
	bits 4-3	J1850	see 5.7.13.7
	bits 2-1	Other, Manufacture Specified Port	see 5.7.13.8
Byte: 3	bits 8-7	SAE J1939 Network #3	see 5.7.13.9
	bits 6-5	SAE Reserved	
	bits 4-3	SAE Reserved	
	bits 2-1	SAE Reserved	
Byte: 4	bits 8-5	Hold Signal	see 5.7.13.10
	bits 4-1	SAE Reserved	
Byte: 5-8		SAE Reserved	

[1] For each of the 2-bit fields in the Stop Start Broadcast command, they are interpreted as follows:

Bits	Information
00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)

The sequence of operation is to first direct DM13 to each (or all) device(s) for which the broadcast state is desired to be modified. The second step is to send DM13 to the global destination address with the appropriate bits set to indicate the “Hold Signal” is being communicated. See Example 1 in Figure 2 and Example 2 in Figure 3. The Hold Signal allows the issuer of the DM13 message to not have to send DM13 to specific addresses but rather to the group of controllers that were modified or all devices. This reduces the number of messages that are required to keep the modified broadcast state of each individual controller active. This has benefit when the individual devices are commanded to turn off different communication ports.

TABLE 5 DM13 USAGE REQUIREMENTS

Purpose	Destination Address	Communication Ports	Hold Signal	Receiving Device Required Action
1. Setup broadcasts to be modified	Specific or Global	Set the action for each communications port to: stop, start, or leave as is	Not Available	Modify Broadcast State
2. Hold modified broadcast state	Global	Set action for each communications port to leave as is	All Devices or Devices with Broadcast State changed	Maintain Modified Broadcast State

EXAMPLE 1—The following illustrates the sequence of messages for a command to stop broadcast to 2 specific nodes to turn off all ports.

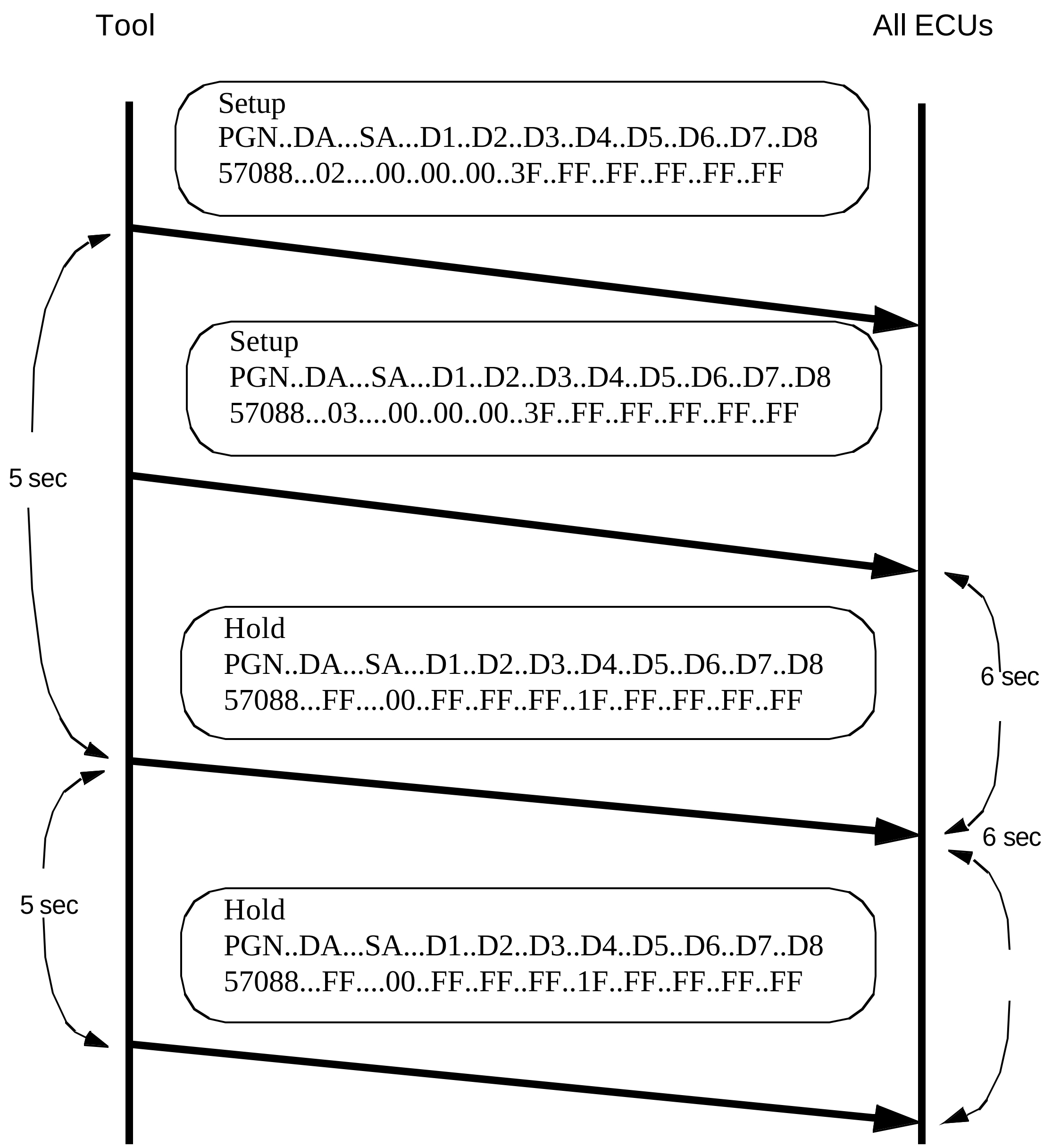


FIGURE 2 STOP START BROADCAST TO 2 SPECIFIC NODES TURNING OFF ALL PORTS

EXAMPLE 2—The following illustrates the sequence of messages for a command to stop broadcast on all nodes and all ports.

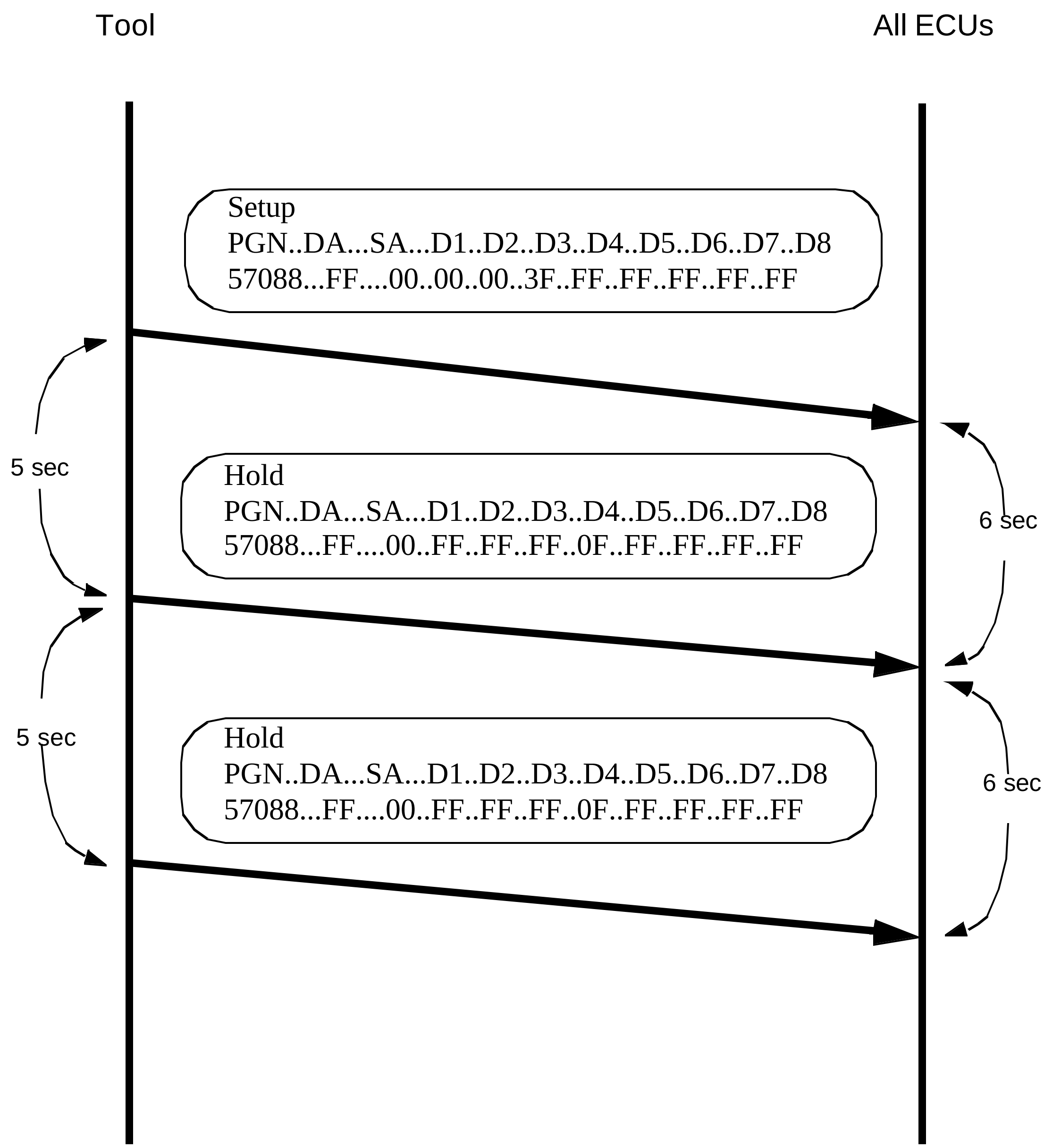


FIGURE 3 STOP START BROADCAST TO ALL NODES TURNING OFF ALL PORTS

5.7.13.1 Current Data Link

Identifies the action to be performed on the communications port that this parameter was received on.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number:	1230
Reference:	5.7.13

5.7.13.2 J1587

Identifies the action to be performed on the J1587 communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number:	608
Reference:	5.7.13

5.7.13.3 J1922

Identifies the action to be performed on the J1922 communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number:	622
Reference:	5.7.13

5.7.13.4 J1939 Network #1, Primary Vehicle Network

Identifies the action to be performed on the J1939 Network #1, Primary Vehicle Network” communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number:	639
Reference:	5.7.13

5.7.13.5 J1939 Network #2

Identifies the action to be performed on the J1939 Network #2 communications port.

00	Stop Broadcast

01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number	1231
Reference:	5.7.13

5.7.13.6 ISO 9141

Identifies the action to be performed on the ISO 9141 communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number	1232
Reference:	5.7.13

5.7.13.7 J1850

Identifies the action to be performed on the J1850 communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number	1233
Reference:	5.7.13

5.7.13.8 Other, Manufacture Specified Port

Identifies the action to be performed on the “Other, Manufacture Specified Port” communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number	1234
Reference:	5.7.13

5.7.13.9 J1939 Network #3

Identifies the action to be performed on the J1939 Network #3 communications port.

00	Stop Broadcast
01	Start Broadcast
10	Reserved
11	Don’t Care/take no action (leave as is)
Type:	Status
Suspect Parameter Number	1235
Reference:	5.7.13

5.7.13.10 Hold Signal

Indicator to all nodes that the communication ports that have been acted upon by the Stop Start Broadcast PGN are remaining in the modified state. Therefore all nodes should act accordingly. The Hold signal is required to be broadcast every 5 seconds plus or minus one second. A device requesting stop broadcast must send the hold signal every 5 seconds and if the message is not received for 6 seconds all applicable nodes revert back to their normal state.

HOLD SIGNAL States

Bit States for bits 8-5	**Devices to take action**
0000	All Devices
0001	Devices whose broadcast state has been modified
0010 to 1110	Reserved
1111	Not Available
Type:	Status
Suspect Parameter Number	1236
Reference:	5.7.13

(R) 5.7.14 MEMORY ACCESS REQUEST (DM14)—The main use for the Memory_Access_Request message is by a Tool wishing to alter the memory of a device. The Tool uses this message to convey its request as well as any security information that must be passed to the device to prove the Tool has authority to request said operation. The message may also be used to obtain the current status of a device in terms of the availability of said device's memory access. The capabilities of the functions are outlined in (R) Appendix B. The procedures for data interchange are outlined in (R) Appendix C. A Memory Access State Transition Diagram (DM14 through DM18) is in (R) Appendix C, Figure C.1.

Transmission Rate:	As needed
Data Length:	8
Data Page:	0
PDU Format:	217
PDU Specific:	DA
Default Priority:	6
Parameter Group Number:	55552 (00D900$_{16}$)

MEMORY_ACCESS_REQUEST

Byte: 1	bits	8-1	Length/Number Requested (Least significant 8 bits) (Bit 1 is least significant bit)	see 5.7.14.5
Byte: 2	bits	8-6	Length/Number Requested (Most significant 3 bits) (Bit 8 is most significant bit)	see 5.7.14.5
	bit	5	Pointer Type	see 5.7.14.1
	bits	4-2	Command	see 5.7.14.6
	bit	1	SAE Reserved (sent as a 1)	
Byte: 3-5			Pointer (Byte 3 is least significant byte) (Bit 1 is least significant bit)	see 5.7.14.3
Byte: 6			Pointer Extension (Most significant byte of Pointer/Pointer Extension) (Bit 8 is most significant bit)	see 5.7.14.2
Byte: 7-8			Key/User_Level	see 5.7.14.4

5.7.14.1 Pointer Type—Indicates whether the Pointer and Pointer Extension are direct memory addresses (Pointer Type identifier = 0) or if the Pointer Extension is identifying a particular SPACE with the Pointer referencing a specific OBJECT within that particular SPACE (Pointer Type identifier = 1).

Data Length:	1 bit
Resolution:	Bitmap
Data Range:	0 or 1
0	Direct Memory Addressing
1	Directed Spatial Addressing
Type:	Status
Suspect Parameter Number:	1641
Reference:	5.7.14

5.7.14.1.1 DIRECT MEMORY ADDRESSING—A Pointer Type value of 0 implies a memory access with a direct memory address (in bytes) with the Pointer Extension (8 bits) simply concatenated as the higher order bits and with the Pointer (24 bits) as the lower order bits to form a single 32-bit address. (Parsing for device memory widths other than 1 byte is explained within "Memory Parsing" on page41.)

5.7.14.1.2 DIRECTED SPATIAL ADDRESSING—A Pointer Type value of 1 implies a memory structure where the Pointer Extension provides identification of a particular SPACE within memory and the Pointer provides identification of a specific OBJECT within that particular SPACE. This provides a form of directed spatial (object) addressing where the user can control the meaning of the pointer used to interrogate a device. For this directed spatial addressing, half of the available SPACES (128 of the 256 formed by the 8-bit Pointer Extension) will be reserved for assignment by the committee. The other half will be labeled proprietary and not constrained by the standard, allowing manufacturer proprietary assignment. The first committee-assigned SPACE is 0 (i.e. Pointer Type = 1 and Pointer Extension = 0) and it is assigned to be the SPACE containing those parameters which can be identified by a particular SPN; this shall be referred to as the SPN SPACE. Objects contained within the Direct Spacial Address Space may be variable length and they may have different lengths.

5.7.14.2 Pointer Extension—This 8-bit parameter is either the high order 8 bits of a complete direct memory address, 5.7.14.1.1 (Pointer Type = 0) or the identifier of a particular SPACE, 5.7.14.1.2 (Pointer Type = 1) (see Table 6).

Data Length:	8 bits
Resolution:	Per definitions in this section
Data Range:	0 to 255
Type:	Status
Suspect Parameter Number:	1643
Reference:	5.7.14

TABLE 6 POINTER EXTENSION STATES
(IF Pointer Type is '1')

Bit States	Pointer Extension States
00000000_{02}	SPN SPACE
00000001_{02} - 01111111_{02}	Presently reserved
10000000_{02} - 11111111_{02}	OEM proprietary definition

5.7.14.2.1 *SPN SPACE*—A Pointer Extension value of '0' in combination with a Pointer Type of '1' implies that the Pointer is to the parameter identified by a specific SPN. Hence the data will be the value of the parameter known by that SPN. (Note: Since some SPNs have no parameter associated with them and in some cases the Device may not know the particular SPN's data, the Memory Access Response may be 'Busy' with or without an Error Indicator.) The length of the data associated with each SPN is a function of that SPN and the overall length of the response message data will be the sum of the byte lengths of the data for each of the SPNs, with the actual number of SPNs being determined from the value of Length/Number Requested parameter. The 5 leading bits should be '000002' to fill the 24-bit Pointer when using a 19-bit SPN. Later at the committee's discretion, functions may be assign to different values of these 5 bits. Therefore, they should be included both when interpreting a value and when sending a value.

5.7.14.2.2 *PRESENTLY RESERVED*—Implies these values are not yet defined by the committee and are therefore not available for use. Future versions of this document will assign specific meanings.

5.7.14.2.3 *OEM PROPRIETARY DEFINITION*—Implies these values are available to OEMs for proprietary definition and use. (One such example use might be for memory block access.)

5.7.14.3 *Pointer*—If Pointer Type 0 is used, this 24-bit parameter, which has a value of 0 to 16,777,215 (0 to $FFFFFF_{16}$) with no reserved ranges, is concatenated with the 8-bit Pointer Extension to form a direct memory address. The address thus formed represents the first address to be accessed within the memory in units of bytes. If Pointer Type 1 is used, the Pointer is to provide the identification of the specific OBJECT within whatever particular SPACE is being identified by the Pointer Extension. The direct memory address should be parsed as outlined below, 5.7.14.3.1, if the device memory width is other than 1 byte.

Data Length:	24 bits
Resolution:	1 byte/bit
Data Range:	0 to 16,777,215 (0 to $FFFFFF_{16}$)
Type:	Status
Suspect Parameter Number:	1644
Reference:	5.7.14

5.7.14.3.1 *MEMORY PARSING*—For all memory widths the starting address is simply the Pointer Extension concatenated with the Pointer (the pointer being the lower 24 bits and the extension the upper 8). For memory widths of one byte there is a one-to-one mapping between data and the memory. Hence the first data byte goes into the memory at the starting address, while the second data byte corresponds to the memory at the starting address plus 1. For widths other than 1 byte, the data cannot map directly to the memory, but must be used to assemble the necessary width. Hence it will take as many data bytes per address as seven plus the memory width in bits all divided by 8 ((memory width + 7)/8). To maintain consistency with the rest of this standard the first data byte should be used for the byte containing bits 1 to 8 at the starting address. The second data byte should be used for bits 9 to 16. This should continue for the number of bytes required; then the address should be incremented and those bytes filled. When the memory width is less than 1 byte (as might happen when addressing a 2-bit parameter through the SPN Space), a whole byte is used to contain each object's data (in other words no packing is to occur). The two bits will be placed in the least significant bits of the byte. For systems where the memory width is not an integer number of bytes, some bits in the highest byte are unused, reducing transfer efficiency, but enabling all memory widths to be handled. Examples of address calculation and byte association (see also 5.7.16.2):

> 8-bit-wide memory, Pointer Extension = 10_{16}, Pointer = 367800_{16} then the starting memory address is 10367800_{16} and the first byte of *Raw Binary Data* would map directly into the memory at 10367800_{16}, the second byte of *Raw Binary Data* would then map into memory 10367801_{16}, and so on until completed.

16-bit-wide memory, Pointer Extension = 10_{16}, Pointer = 367800_{16} then the starting memory address is 10367800_{16} and the first byte of *Raw Binary Data* would map into bits 1 to 8 of the memory at 10367800_{16}, while the second byte of *Raw Binary Data* would map into bits 9 to 16 of the same memory. The third byte of *Raw Binary Data* would then map into bits 1 to 8 of the memory at 10367801_{16}, while the fourth byte of *Raw Binary Data* would map into bits 9 to 16 of the memory at 10367801_{16}.

32-bit-wide memory, Pointer Extension = 10_{16}, Pointer = 367800_{16} then the starting memory address is 10367800_{16} and the first byte of *Raw Binary Data* would map into bits 1 to 8 of the memory at 10367800_{16}, while the second byte of *Raw Binary Data* would map into bits 9 to 16 of the memory at 10367800_{16}, the third byte of *Raw Binary Data* would then map into bits 17 to 24 of the same memory and the fourth byte of *Raw Binary Data* would map bits 25 to 32. The fifth byte of *Raw Binary Data* would then map bits 1 to 8 of 10367801_{16}, while the sixth byte of *Raw Binary Data* would map into bits 9 to 16 of 10367801_{16}, the seventh byte of *Raw Binary Data* then mapping bits 17 to 24 and the eight byte of *Raw Binary Data* mapping to bits 25 to 32 of the memory.

12-bit-wide memory, Pointer Extension = 10_{16}, Pointer = 367800_{16} then the starting memory address is 10367800_{16} and the first byte of *Raw Binary Data* would map into bits 1 to 8 there, while bits 9 to 12 of the second byte of *Raw Binary Data* would map into bits 9 to 12 of 10367800_{16} (bits 13 to 16 are simply unused). The third byte of *Raw Binary Data* would then map into bits 1 to 8 of 10367801_{16}, while bits 9 to 12 of the fourth byte of *Raw Binary Data* would map into bits 9 to 12 of 10367801_{16}.

5.7.14.3.2 *HANDLING OF POINTER OFFSET*—When the starting address created by concatenating the Pointer Extension and the Pointer does not represent the beginning of an OBJECT, such as a memory block or memory word, the Device shall be free to reject the requested memory access operation. If used by the manufacturer, then the appropriate Error Indicator/EDC Parameter may be returned (see Table 9 and 5.7.15.3).

5.7.14.4 *Key/User_Level*—This is a 2-byte parameter which is used by the Tool to primarily send a Key to the Device, but which can also be used by the Tool to provide a Password or a User_Level to the Device if desired (see (R) Appendix C). This Key/User_Level parameter can be used to send these independent variables since they will never be transmitted within the same message (a Password or User_Level parameter would be sent at the beginning of an operation, while a Key CANNOT be sent until after the receipt of a Seed).

Data Length:	16 bits
Resolution:	Per definitions in this section
Data Range:	0 to 65535 (0 to $FFFF_{16}$)
Type:	Status
Suspect Parameter Number:	1645
Reference:	5.7.14

5.7.14.4.1 *KEY*—The result of a set of mathematical operations performed upon a Seed to provide a device with a means of authenticating a Tool's request (see Table 7).

TABLE 7 KEY STATES

(i.e. ONLY for Key and NOT User Level)

Bit States	Key States
0000_{16}	Use Long Seed or Key from Data Security Message
0001_{16} to $FFFE_{16}$	Key Values
$FFFF_{16}$	No Key Available

5.7.14.4.1.1 *USE LONG SEED OR KEY FROM DATA SECURITY MESSAGE*—The actual Seed or Key is in the Data Security message and this is simply a flag.

5.7.14.4.1.2 *KEY VALUES*—The actual values of the Key.

5.7.14.4.1.3 *NO KEY AVAILABLE*—There is no Key at this time.

5.7.14.4.2 *SEED*—A number sent by a device to a Tool to obtain authentication of the Tool's right to access the device. The Tool must return a Key, which is a function of the Seed, and the Key matches the device's expectations to obtain access.

5.7.14.4.3 *PASSWORD*—The number sent when using a simple authentication technique wherein both the Device and Tool have a prior knowledge of the specific number and usually use equality as the verification.

5.7.14.4.4 *USER_LEVEL*—A number sent by a Tool to a device along with an initial request to inform the device of some specific level of access that the Tool wishes to gain. In such a case there is probably a following Seed and Key exchange. Usually the Seed and the mathematical operations to calculate the Key from it would be a function of the User_Level requested.

5.7.14.4.5 *HANDLING OF KEYS LARGER THAN 16 BITS*— If a manufacturer feels a Seed/Key structure requires a Key or Seed longer than 16 bits for a particular Device, this can be handled with the Data Security message (see (R) 5.7.18) and the setting of the Key/User_Level and the Seed parameters appropriately (see Table 7 and Table 10). Also see Figure E.9 and Figure E.10

5.7.14.4.6 *ACCEPTANCE RULES*—The Device will establish the set of rules governing acceptance of memory access requests (such as operational mode, User_Level versus Memory (OBJECT) Location versus requested operation, etc.). These rules may be manufacturer specific to prevent unauthorized modification of a Device's memory. The Command and the Length/Number Requested with the Memory Access Request message(s) should be constant throughout an entire sequence or the Device should reject the operation.

5.7.14.5 *Length/Number Requested*—This is an 11-bit parameter which identifies the amount of memory (i.e. the range within the memory) over which the Tool desires an operation to be carried out when the Command of the Memory Access Request message is a Read, Write, Boot Load, or Error Detection and/or Correction Parameter (EDCP) Generation. For these cases the length is in bytes (cross-reference to 5.7.14.3.1 Memory Parsing) when the Pointer Type is '0' and in objects when the Pointer Type is '1' (example: a length of 2 when referencing the SPN Space with a Pointer of 1648 would imply that you wanted the data of the two parameters with SPNs 1648 and 1649). When the Command is Erase, the length is the number of 'blocks' of memory to be erased. Block size being specific to the device. When the Command of the Memory Access Request is Operation Failed, Operation Completed, or Status Request the length is meaningless. The Tool should therefore send it as '0' and the Device should treat it as ‘DO NOT CARE'.

Data Length:	11 bits
Resolution:	Pointer Type 0 = 1 byte/bit Pointer Type 1 = 1 object/bit
Data Range:	0 to 1784
Type:	Status
Suspect Parameter Number:	1640
Reference:	5.7.14

5.7.14.6 *Command*—This is a 3-bit parameter which allows the Tool to send Commands to the Device. All Memory Access Requests originate at a Tool and are considered Commands. Some of the values within the Command have been overlaid with the same values in Status so perhaps a similar variable can be used.

Data Length:	3 bits
Resolution:	1 bit/command
Data Range:	0 to 7
Type:	Status
Suspect Parameter Number:	1642
Reference:	5.7.14

The following Command values for Memory Access Requests (i.e. Tool to Device) are defined:

0 - Erase	see 5.7.14.6.1
1 - Read	see 5.7.14.6.2
2 - Write	see 5.7.14.6.3
3 - Status Request	see 5.7.14.6.4
4 - Operation Completed	see 5.7.14.6.5
5 - Operation Failed	see 5.7.14.6.6
6 - Boot Load	see 5.7.14.6.7
7 - EDCP Generation	see 5.7.14.6.8

5.7.14.6.1 *ERASE*—This command is a block erase where the length is defined as the number of blocks to erase. The length and width of a block are to be defined within the Device and must be known by the Tool. The Pointer is the address where erase should start. If the Pointer is not on a block boundary, when corrected for memory width, then the erase is not to be allowed (i.e. the device is to respond with a ‘busy’ with or without data in the Error Indicator/EDC Parameter per the manufacturer’s choice, see C.2.2.1).

5.7.14.6.2 *READ* —This command allows the transfer of the contents from a Device memory to a Tool. When this command is accepted, the Device transfers the appropriate memory contents to the Tool, including initiating a transport protocol session if necessary. See Figure E.1 for an example message sequence used to accomplish a memory read operation with security (short form of security). See Figure E.2 for an example message sequence used to accomplish a memory read operation without security. See Figure E.3 for an example message sequence used to accomplish a multiple memory read operation with security (short form of security). Note that the operation complete message from the Tool indicates the session is over from the Tool’s perspective. See Figure E.7 for an example of a failed Memory Access read operation.

5.7.14.6.3 *WRITE*—This command allows the transfer of the new memory contents from a Tool to a Device. The Device may use a smart write, which will force an erase before write if it is going to be necessary, due to the value being written and the memory type, and if any other memory which will be altered by the erase can be buffered and rewritten to its original value, effecting a transparent write operation. See Figure E.5 for an example message sequence used to accomplish a memory write operation with security (short form of security).

5.7.14.6.4 *STATUS REQUEST*—This command allows a Tool to interrogate the Device to determine the current status of operation. This enables a Tool to determine what a Device may currently be doing and/or why it has not heard a message indicating Operation Completed/Failed from the Device, when the Tool itself believes sufficient time has elapsed for the operation. The Device responds to this command with either Operation Completed, Operation Failed, Proceed or Busy with a code indicating the current status or error condition within the Feedback parameter.

5.7.14.6.5 *OPERATION COMPLETED*—This command is sent during a close sequence. Operation Completed is sent by the Tool during the close sequence of an Erase, Read, Write, Boot Load, or EDCP Generation command to indicate to the Device that the Tool has heard the Device's close and that the close sequence is completed. At the end of a Read command, it further indicates that all of the expected memory contents were received. A Device's receipt of an Operation Completed from a Tool enables the Device to consider the Memory Access finished. The Device should have a time-out function such that on the failure to hear the expected Operation Completed (or Operation Failed) from the Tool it assumes the Memory Access operation with the Tool is complete. See Figure E.4. The value for this time-out should be 100 ms with no worse than ±25 ms error. (See also the section entitled Operation Completed under Memory Access Response - 5.7.15.1.3.)

5.7.14.6.6 *OPERATION FAILED*—This command is sent by a Tool only during the close sequence of a Read command to indicate to the Device that the expected memory contents were not received. (This initiates no further action from the Device.)

5.7.14.6.7 *BOOT LOAD*—This command allows a Tool to transfer the Execution of a Device to some address and if needed, write new values into this Executable memory prior to transfer of execution. When NO data is written, the Device considers the operation simply a transfer of execution and continues operation with no change in network communications, but with whatever other software changes the manufacturer has chosen to implement. When writing new data any values must be executable and upon a successful close, the Device transfers execution to the address specified by the request initiating this operation. Once execution has transferred at the close of a Boot Load, the Device is no longer required (it may do so if its designers choose) to operate upon any messages from nodes other than the specific Tool that initiated the Boot Load. If the device no longer operates upon other messages, the Tool is be required to protect the address of the Device from any Address Claim messages sent during the boot load process. If the close sequence of Boot Load, with data, indicates successful completion, then the Tool sends data to the Device using the Boot Load Data PGN until the Tool determines that the Boot Load sequence has been completed. The Tool then notifies the operator that the operation is complete so that the Device (and probably the system and network) can be restarted at its Power On Self-Test. There is no predefined close sequence for the end of the boot load data transfer provided by this standard. It is at the manufacturer's discretion to choose to have such a sequence.

5.7.14.6.8 *EDCP GENERATION* —This command allows a Tool to request a device to generate a checksum or other form of memory error detection and correction parameter over some range of memory. It is expected that the Tool must have a prior knowledge of the length and generation procedure used by the device. Parameters greater than 24 bits in length are handled by a looping concatenation structure (see also "EDCP Extension" on page47). The memory involved in this operation is the same as that defined within the read operation.

(R) 5.7.15 MEMORY ACCESS RESPONSE (DM15)—The main use for the Memory_Access_ Response is for a device to answer a Tool which has attempted to access the memory within the device. With this message the device can request further security responses from the requestor (see Figure E.6) as well as tell the requestor what is or is not allowed. The completion status of a memory operation may also be transferred with this message. A Memory Access State Transition Diagram (DM14 through DM18) is in (R) Appendix C.

Transmission Rate:	As needed
Data Length:	8
Data Page:	0
PDU Format:	216
PDU Specific:	DA
Default Priority:	6
Parameter Group Number:	55296 ($00D800_{16}$)

MEMORY_ACCESS_RESPONSE

Byte: 1	bits 8-1	Length/Number Allowed (Least significant 8 bits) (Bit 1 is least significant bit)	see 5.7.15.5
Byte: 2	bits 8-6	Length/Number Allowed (Most significant 3 bits) (Bit 8 is most significant bit)	see 5.7.15.5
	bit 5	SAE Reserved	
	bits 4-2	Status	see 5.7.15.1
	bit 1	SAE Reserved	
Byte: 3-5		Error Indicator/EDC Parameter (Byte 3 is least significant byte) (Bit 1 is least significant bit)	see 5.7.15.3
Byte: 6		EDCP Extension (When used as an EDCP extension, this is the most significant byte) (Bit 8 is most significant bit)	see 5.7.15.2
Byte: 7-8		Seed	see 5.7.15.4

5.7.15.1 *Status*—This is a 3-bit parameter which allows the Device to return its Status. All Memory Access Requests originate at a Tool and are considered Commands. All Memory Access Responses originate at a Device and are considered Status. The device may choose to send further information on its status within the Error Indicator/EDC Parameter (see 5.7.15.3 and 5.7.15.2).

Data Length:	3 bits
Resolution:	1 status value/bit
Data Range:	0 to 7
Type:	Status
Suspect Parameter Number:	1646
Reference:	(R) 5.7.15

The following Status values for Memory Access Responses (i.e. Device to Tool) are defined:

0 - Proceed	see 5.7.15.1.1
1 - Busy	see 5.7.15.1.2
2 - Reserved	
3 - Reserved	
4 - Operation Completed	see 5.7.15.1.3
5 - Operation Failed	see 5.7.15.1.4
6 - Reserved	
7 - Reserved	

5.7.15.1.1 *PROCEED*—This Status is sent from a Device to indicate that a specific Tool may continue with the sequence of a memory access operation the Tool had requested. When sent as a response to a Status Request command, this means the Device is not presently engaged in any Memory Access Operation (i.e. is not Busy). Optionally, at the manufacturer's preference, the Error Indicator/EDC Parameter may contain the Error Indicator for the previous operation the Device had performed (should be $FFFFFF_{16}$ otherwise).

5.7.15.1.2 *BUSY*—This Status is sent from the Device to indicate to a Tool that there is a condition which prevents the sequence from continuing. The Length/Number Allowed parameter will be zero, the Error Indicator/ EDC Parameter will contain a value indicating the condition which is preventing the Memory Access from continuing, with the manufacturer having a choice of how detailed the Error Indicator is (see 5.7.15.3), and the value of the other parameters will be treated as Do Not Care. When issued as a response to a Memory Access Status Request command, this means the Device may still be busy and in the process of completing a requested operation (this includes but is not limited to: transmitting/ receiving Data required for an operation, erasing memory, or programming memory). See Figure E.8 for an example use of the busy indication.

5.7.15.1.3 *OPERATION COMPLETED*—This Status is sent during a close sequence or in response to a Status Request command. Operation Completed is sent as status from the Device during the close sequence of an Erase, Read, Write, Boot Load, or EDCP Generation command to indicate that the request was successfully completed, there may be an EDC value contained within the Error Indicator/EDC Parameter. This Status is the start of the close sequence for all successful Commands which operate upon a Device's memory. A Device's receipt of an Operation Completed from a Tool enables the Device to consider the Memory Access finished. (See also Operation Completed under Memory Access Request - 5.7.14.6.5.) The Device should have a time-out such that on failure to hear the expected Operation Completed (or Operation Failed) from the Tool it closes the session. The value for this time-out should be 100 ms with no worse than ±25 ms error. See Figure E.4 for an example where the Tool does not send the required operation complete message. When the Operation Completed message is sent by a Device in response to a Status Request, it indicates that the last operation was successfully completed only if the close sequence has not been completed. Once the close sequence is completed for an operation, a Device no longer needs to maintain any data about that operation and may send a Status of Proceed, with or without the Error Indicator value from the previous operation, in response to a Status Request.

5.7.15.1.4 *OPERATION FAILED*—This Status is sent during a close sequence or in response to a Status Request command. Operation Failed is sent as status from the Device during the close sequence of an Erase, Write, Boot Load or EDCP Generation command to indicate that the request was unsuccessful, the Error Indicator/EDC Parameter should contain an Error Indicator. When sent in response to a Status Request, it indicates that the last operation failed only if the close sequence has not been completed. Once the close sequence is completed for an unsuccessful operation, the Device sends the Proceed status in response to a Status Request.

5.7.15.2 *EDCP Extension*—This is an 8-bit parameter used to identify how to handle the data in the Error Indicator/ EDC Parameter. This EDCP Extension parameter is used within the Memory Access Response message (Device to Tool). Meaning must be determined from a table of predefined values (Table 8). If there is no Error Indicator/EDC Parameter being sent then this (EDCP Extension) parameter must be properly set (11111111_{02}). The use of the Error Indicator/EDC Parameter is at the manufacturer's discretion, but it must be properly set relative to this parameter. For example: Suppose the unit is not willing to reveal the current cause of an error for security reasons, then if this EDCP Extension is set to 00000110_{02} then the Error Indicator/EDC Parameter must be set to 000001_{16} to indicate the error is not identified (see Table 8).

Data Length:	8 bits
Resolution:	1 state/bit
Data Range:	0 to 255 (0 to FF_{16})

Type: Status
Suspect Parameter Number: 1647
Reference: (R) 5.7.15

TABLE 8 EDCP EXTENSION STATES

Bit States	EDCP Extension States
00000000_{02}	Completed - all of the EDC Parameter has been sent
00000001_{02}	Reserved for SAE Assignment
00000010_{02}	More - Concatenate the following data as Higher order EDC Parameter
00000011_{02}	More - Concatenate the following data as Lower order EDC Parameter
00000100_{02} - 00000101_{02}	Reserved for SAE Assignment
00000110_{02}	Data in Error Indicator/EDC Parameter is an Error Indicator
00000111_{02}	Data in Error Indicator/EDC Parameter is an Error Indicator and Data in Seed is an expected time to completion
00001000_{02} - 11111110_{02}	Reserved for SAE Assignment
11111111_{02}	No Error Indicator/EDC Parameter Available

5.7.15.2.1 *COMPLETED*—An EDCP Extension value of '0' implies all of the EDC Parameter has been sent within the Error Indicator/EDC Parameter.

5.7.15.2.2 *MORE - CONCATENATE AS HIGHER*—Implies the following EDCP components should be concatenated as the next HIGHER order with those previously received.

5.7.15.2.3 *MORE - CONCATENATE AS LOWER*—Implies the following EDCP components should be concatenated as the next LOWER order with those previously received.

5.7.15.2.4 *ERROR INDICATOR/EDC PARAMETER DATA IS AN ERROR INDICATOR*—Implies the following data is an Error Indicator value and not a component of an EDCP.

5.7.15.2.5 *ERROR INDICATOR/EDC PARAMETER IS AN ERROR INDICATOR AND DATA IN SEED IS AN EXPECTED TIME TO COMPLETION*—Implies the following data is an Error Indicator value and not a component of an EDCP, as well as the Seed parameter contains an expected time to completion. Time value shall have a resolution of 0.1 seconds per bit.

5.7.15.2.6 *NO EDCP AVAILABLE*—Implies there are no EDCP components available in this system - and could easily imply that an EDCP is not even used.

5.7.15.3 *Error Indicator/EDC Parameter*—This is a 24-bit parameter which has two uses. One is to transfer a checksum, CRC or other type of EDC parameter (or any segment thereof) from a device to a Tool within the Memory Access Response message. The second use is to send an Error Indicator any time the Device is not able to complete or act upon a Tool's request. Some Error Indicator States are predefined (see Table 9) although it is up to the manufacturer to decide if a particular error will be identified. (If identified, the pre-defined value is to be used.) The Error Indicator is only valid when EDCP Extension is equal to a value of

6 or 7, see Table 8. The Tool is responsible for knowing the EDC Parameter generation techniques used by the device. The Tool is also responsible for the verification that the EDCP is correct. The EDCP is sent within the Close sequence (see (R) Appendix C) at the completion of each operation. Since some users may wish an EDCP greater than 24 bits there is provision to form a larger value by concatenation. In such cases the EDCP Extension parameter is used to determine the direction of concatenation and the completion of the concatenation sequence. An EDCP Extension value of "all 1's" implies that the EDCP, as well as the extension, is not available and is not really being used by the device. In such cases the value in the EDCP has no meaning (see Table 9).

Data Length:	24 bits
Resolution:	Per definitions in this section
Data Range:	0 to 16,777,215 (0 to $FFFFFF_{16}$)
Type:	Status
Suspect Parameter Number:	1648
Reference:	(R) 5.7.15

TABLE 9 ERROR INDICATOR STATES

Bit States	Error Indicator States
000000_{16}	No Error
000001_{16}	Error NOT identified
000002_{16}	Currently processing for someone else
000003_{16} - $00000F_{16}$	Reserved for SAE Assignment
000010_{16}	Currently processing Erase Request
000011_{16}	Currently processing Read Request
000012_{16}	Currently processing Write Request
000013_{16}	Currently processing Status Request
000014_{16}	Reserved for SAE Assignment
000015_{16}	Reserved for SAE Assignment
000016_{16}	Currently processing Boot Load Request
000017_{16}	Currently processing EDCP Generation Request
000018_{16} - $00001E_{16}$	Reserved for SAE Assignment
$00001F_{16}$	Currently processing unspecified request from this address
000020_{16}	EDC parameter not correct for data stream
000021_{16}	RAM did not verify on Write
000022_{16}	FLASH did not verify on Write
000023_{16}	PROM did not verify on Write

000024_{16}	Internal failure preventing request (i.e. within the ECU)
000025_{16} - $0000FF_{16}$	Reserved for SAE Assignment
000100_{16}	Addressing or DATA General Error
000101_{16}	Addressing Error - Address not on a valid boundary (Block, Word, Object, etc.)
000102_{16}	Addressing Error - Length not valid for memory structure and operation
000103_{16}	Addressing Error - required memory exceeded available memory
000104_{16}	Addressing Error - requested operation requires prior erase of DATA memory
000105_{16}	Addressing Error - requested operation requires prior erase of PROGRAM memory
000106_{16}	Addressing Error - requested operation requires prior execution transfer and erase of PROGRAM memory
000107_{16}	Addressing Error - requested address for Boot Loader execution transfer is NOT within executable memory
000108_{16}	Addressing Error - requested address for Boot Loader execution transfer is NOT on valid boundary
000109_{16}	DATA Error - data does NOT conform to expected or allowed value ranges
$00010A_{16}$	DATA Error - NAME does NOT conform to expected value
$00010B_{16}$ - $0001FF_{16}$	Reserved for SAE Assignment
001000_{16}	Security Error General
001001_{16}	Security Error - Invalid Password
001002_{16}	Security Error - Invalid User Level
001003_{16}	Security Error - Invalid Key {Seed}
001004_{16}	Security Error - NOT in Diagnostic mode
001005_{16}	Security Error - NOT in Engineering or Development mode
001006_{16}	Security Error - Engine running
001007_{16}	Security Error - Vehicle NOT in “Park” or otherwise NOT stationary
001008_{16} - $0FFFFE_{16}$	Reserved for SAE Assignment
010000_{16}	Abort from external to normal software process
010001_{16}	Too Many Retries - module exceeding a set number of retries
010002_{16}	NO response in the time allowed
010003_{16}	Transport of data NOT initiated within the time allowed

010004_{16}	Transport of data NOT completed within the time allowed
010005_{16} - $FFFFFE_{16}$	Reserved for SAE Assignment
$FFFFFF_{16}$	No Error Indicator Available

5.7.15.3.1 *NO ERROR*—An Error Indicator value of '0' implies no error was detected by the Device.

5.7.15.3.2 *ERROR NOT IDENTIFIED*—Implies the Device could (or would) NOT identify the specific error preventing continued operation. This value is to be used by the manufacturer when the Device is unable (or unwilling) whether by design or failure to generate a more detailed summary of the fault or condition preventing continued operation on the given Memory Access Request.

5.7.15.3.3 *CURRENTLY PROCESSING FOR SOMEONE ELSE*—Implies that the Device is processing a Memory Access for some other address than the one which just requested.

5.7.15.3.4 *CURRENTLY PROCESSING SOME REQUEST FOR THIS DEVICE*—There are several errors which could be the result of the Device already being busy processing a Memory Access Request from this address. Since it was thought by some that it would be nice to know the specific operation underway, several errors have been assigned. It is planned that these errors will be grouped between the values 10_{16} and $1F_{16}$. The specific request can then be identified by the lower nibble of lower byte of Error Indicator/ EDC parameter as:

5.7.15.3.4.1 *CURRENTLY PROCESSING ERASE REQUEST*—Implies that the Device is processing a Memory Access Erase from this address already.

5.7.15.3.4.2 *CURRENTLY PROCESSING READ REQUEST*—Implies that the Device is processing a Memory Access Read from this address already.

5.7.15.3.4.3 *CURRENTLY PROCESSING WRITE REQUEST*—Implies that the Device is processing a Memory Access Write from this address already.

5.7.15.3.4.4 *CURRENTLY PROCESSING STATUS REQUEST*—Implies that the Device is processing a Memory Access Status Request from this address already.

5.7.15.3.4.5 *CURRENTLY PROCESSING BOOT LOAD REQUEST*—Implies that the Device is processing a Memory Access Boot Load from this address already.

5.7.15.3.4.6 *CURRENTLY PROCESSING EDCP GENERATION REQUEST*—Implies that the Device is processing a Memory Access EDCP Generation Request from this address already.

5.7.15.3.4.7 *CURRENTLY PROCESSING UNSPECIFIED REQUEST*—Implies that the Device is not identifying the specific Request it is presently processing, but is identifying that it is from this address.

5.7.15.3.5 *SOME FAILURE*—There a several errors which can be sent to indicate that an operation has failed. Some errors do not indicate the exact problem but more the event. Members of this form of error will be grouped between values 20_{16} and FF_{16}. The presently assigned errors are:

5.7.15.3.5.1 *EDC PARAMETER NOT CORRECT FOR DATA STREAM*—Implies that EDC was not correct for the data.

5.7.15.3.5.2 *RAM DID NOT VERIFY ON WRITE*—Identifies that some failure has caused RAM not to verify following a write.

5.7.15.3.5.3 *FLASH DID NOT VERIFY ON WRITE*—Identifies that some failure has caused a FLASH memory not to verify following a write.

5.7.15.3.5.4 *PROM DID NOT VERIFY ON WRITE*—Identifies that some failure has caused a PROM memory not to verify following a write.

5.7.15.3.6 *ADDRESSING OR DATA ERRORS*—There are several errors which imply that something was wrong with the addressing of the request or the data sent for the request. These errors have been grouped with a lower value of 100_{16} and the specific error can be parsed on the value of the lower byte as:

5.7.15.3.6.1 *ADDRESSING OR DATA GENERAL ERROR*—Identifies that the failure has been within the addressing or data but that it can not be identified further.

5.7.15.3.6.2 *ADDRESSING ERROR - LENGTH NOT VALID FOR MEMORY STRUCTURE AND OPERATION*—Identifies that the failure has been a length which is not compatible with the memory and/or the particular operation attempted upon said memory.

5.7.15.3.6.3 *ADDRESSING ERROR - REQUIRED MEMORY EXCEEDED AVAILABLE MEMORY*—Identifies that the failure has been a request for which there is not sufficient memory available.

5.7.15.3.6.4 *ADDRESSING ERROR - REQUESTED OPERATION REQUIRES PRIOR ERASE OF DATA MEMORY*—Identifies that the failure has been a request for which there needed to be an erase of some DATA memory prior to the requested operation.

5.7.15.3.6.5 *ADDRESSING ERROR - REQUESTED OPERATION REQUIRES PRIOR ERASE OF PROGRAM MEMORY*—Identifies that the failure has been a request for which there needed to be an erase of some PROGRAM memory prior to the requested operation.

5.7.15.3.6.6 *ADDRESSING ERROR - REQUESTED OPERATION REQUIRES PRIOR EXECUTION TRANSFER AND ERASE OF PROGRAM MEMORY*—Identifies that the failure has been a request for which there needed to be a transfer of execution to some other program segment and an erase of some PROGRAM memory prior to the requested operation.

5.7.15.3.6.7 *ADDRESSING ERROR - REQUESTED ADDRESS FOR BOOT LOADER EXECUTION TRANSFER IS NOT WITHIN EXECUTABLE MEMORY*—Identifies that the failure has been a request to transfer execution to some address not in an executable memory.

5.7.15.3.6.8 *ADDRESSING ERROR - REQUESTED ADDRESS FOR BOOT LOADER EXECUTION TRANSFER IS NOT ON A VALID BOUNDARY*—Identifies that the failure has been a request to transfer execution to some address not on a valid boundary within executable memory.

5.7.15.3.7 *SECURITY ERROR*—There are several errors which imply that something was wrong with the security used within the request. These errors have been grouped with a lower value of 1000_{16} and the specific error can be parsed on the value of the lower byte as:

5.7.15.3.7.1 *SECURITY ERROR GENERAL*—Identifies that the failure has been within the security but that it is not (or can not be) identified any further.

5.7.15.3.7.2 *SECURITY ERROR - INVALID PASSWORD*—Identifies that the failure has been an invalid Password for the requested operation.

5.7.15.3.7.3 *SECURITY ERROR - INVALID USER LEVEL*—Identifies that the failure has been an invalid User Level for the requested operation.

5.7.15.3.7.4 *Security Error - Invalid Key{Seed}*—Identifies that the failure has been an invalid Key returned for the Seed that was provided for the requested operation.

5.7.15.3.7.5 *Security Error - NOT in Diagnostic mode*—Identifies that the failure has been that the unit is not in some Diagnostic mode prior to the requested operation. This is an allowable manufacturer additional requirement.

5.7.15.3.7.6 *Security Error - NOT in Engineering or Development mode*—Identifies that the requested operation requires that the unit be in an Engineering or Development mode prior to the requested operation. This is an allowable manufacturer additional requirement.

5.7.15.3.7.7 *Security Error - Engine running*—Identifies that the requested operation requires the engine to be stopped prior to the requested operation. This is an allowable manufacturer additional requirement.

5.7.15.3.7.8 *Security Error - Vehicle NOT in PARK or otherwise NOT stationary*—Identifies that the requested operation requires the vehicle to be in Park or otherwise not able to move prior to the requested operation. This is an allowable manufacturer additional requirement.

5.7.15.3.8 *Time-out Errors*—There are several errors which imply that something has taken too long or too many tries and the unit has given up. These errors have been grouped with a lower value of 10000_{16} and the specific error can be parsed on the value of the lower byte as:

5.7.15.3.8.1 *Abort from external to normal software process*—Identifies that some event within the unit has caused an abort of this software process. Hence this Memory Access operation has also been terminated.

5.7.15.3.8.2 *Too Many Retries*—Identifies that the failure has been an excessive number of attempts were made without the desired event occurring.

5.7.15.3.8.3 *No Response in time allowed*—Identifies that there has been a time-out within the process, although no further identification of the time-out is possible.

5.7.15.3.8.4 *Transport of data NOT initiated within the time allowed*—Identifies that there has been a time-out within the process, and that it was in waiting for the establishment of the transport session to send the data.

5.7.15.3.8.5 *Transport of data NOT completed within the time allowed*—Identifies that there has been a time-out within the process, and that it has taken too long for the transport session to complete the sending the data.

5.7.15.3.9 *No Error Indicator Available*—Implies there is no Error Indicator AVAILABLE at this time.

5.7.15.4 *Seed*—This is a 16-bit parameter which is used by the Device primarily to send a Seed to a Tool, when using a Seed/Key type security system. It is also used by the Device to signal the Tool that the Device is satisfied that a complete Key has been received or that the Data Security message is expected to contain the Seed data. This parameter can also contain an expected time to completion when the EDCP Extension is 7 (see "Error Indicator/EDC Parameter Is an Error Indicator and Data in Seed Is an Expected Time to Completion" on page 48). The Seed is to be the mathematical basis upon which any Key is calculated. The Device verifies the validity of the Key {Seed} from the Tool and enable memory access operations appropriately. See also "Key/User_Level" on page42 (see Table 10).

Data Length:	16 bits
Resolution:	Per definitions in this section
Data Range:	0 to 65535 (0 to $FFFF_{16}$)

Type: Status
Suspect Parameter Number: 1599
Reference: (R) 5.7.15

TABLE 10 SEED STATES

Bit States	Seed States
0	Seed Completed - begin sending key
1	Use Long Seed or Key from Data Security Message
2 - $FFFE_{16}$	Seed values
$FFFF_{16}$	No Further Key required of Tool

5.7.15.4.1 *SEED COMPLETED*—A specific value to indicate that no further Seed or Seed segments is going to be provided by the Device, see also "Handling of Keys Larger than 16 Bits" on page43 and "No Further Key Required of Tool" on page54. The Tool should understand it is to begin the operation supposing that the operation is allowed within the Status parameter and that a non-zero length has been allowed.

5.7.15.4.2 *USE LONG SEED OR KEY FROM DATA SECURITY MESSAGE*—The actual Seed or Key is in the Data Security message.

5.7.15.4.3 *SEED VALUES*—The actual numeric values which can be used for Seeds.

5.7.15.4.4 *NO FURTHER KEY REQUIRED OF TOOL*—A specific value to indicate that no Key or further Key segments is required of the Tool to begin this operation, see also "Handling of Keys Larger than 16 Bits" on page43 and "Seed Completed" on page54.

5.7.15.4.5 *HANDLING OF KEYS LARGER THAN 16 BITS*—If a manufacturer feels a Seed/Key structure requires a Key or Seed longer than 16 bits for a particular Device, this can be handled with the Data Security message (see "Data Security (DM18)" on page58) and the setting of the Key/User_Level and the Seed parameters appropriately (see Table 7 and Table 10).

5.7.15.4.6 *ACCEPTANCE RULES*—The Device establishes the set of rules governing acceptance of memory access requests (such as operational mode, User_Level versus Memory (OBJECT) Location versus requested operation, etc.). These rules may be manufacturer specific to prevent unauthorized modification of a Device's memory. The manufacturer also has the option to allow an initial operation by a Tool to establish a security level and then let the device honor multiple requests from the same Tool (i.e. the network node whose Source Address matches the Source Address originally used by the Tool opening the original memory access operation, the manufacturer has the option to further check the NAME to address association) without further security operations. This optional re-entrant security is to be considered completely ended when the Tool sends a 'close' or the device time-outs waiting for the Tool's 'close'.

5.7.15.4.7 *EXPECTED TIME TO COMPLETION VALUES*—The expected time to completion of an operation when the device is already processing a request. Numeric values are in milliseconds.

5.7.15.5 *Length/Number Allowed*—When the Status of the Memory Access Response message is a Proceed, an 11-bit parameter identifies the amount of memory (i.e. the range of memory) over which the Device is willing to allow a particular operation to be carried out. For 'Proceed' the length value is either in bytes or objects (see Length/Number Requested in Section 5.7.14.5). When the Status of the Memory Access

Request is Busy, Operation Failed, or Operation Completed the length is meaningless. The Device should therefore send it as '0' and the Tool should treat it as 'DO NOT CARE'.

Data Length:	11 bits
Resolution:	1 byte/bit or object/bit or other
Data Range:	0 to 1784
Type:	Status
Suspect Parameter Number:	1649
Reference:	(R) 5.7.15

(R) 5.7.16 BINARY DATA TRANSFER (DM16)—Used primarily to transfer data for the memory access commands. A Memory Access State Transition Diagram (DM14 through DM18) is in (R) Appendix C.

Transmission Rate:	As needed
Data Length:	Variable (8 to *n*)
Data Page:	0
PDU Format:	215
PDU Specific:	DA
Default Priority:	6
Parameter Group Number:	55040 ($00D700_{16}$)

BINARY_DATA_TRANSFER

Byte: 1	Number of Occurrences of Raw Binary Data	see 5.7.16.1
Bytes: 2-8	Raw Binary Data	see 5.7.16.2
Bytes: 9-*n*	Raw Binary Data - when multipacketed	see 5.7.16.2

5.7.16.1 Number of Occurrences of Raw Binary Data—This is an 8-bit (1-byte) parameter to be sent within the Binary Data Transfer PGN to provide information on the number of Raw Binary Data parameters which will follow when the message is single packet. Its value is between 1 and 7 when the Binary Data Transfer PGN is not multipacketed. If the message is multipacketed then the value of this parameter is to be 255 (FF_{16}). When the message is multipacketed the number of occurrences of the Raw Binary Data parameter must be determined from the Total Message Size parameter (see J1939-21) sent in the connection request message.

Data Length:	8 bits
Resolution:	1 byte/bit
Data Range:	1 to 7 or 255 Values 0 or 8 to 254 are not used
Type:	Status
Suspect Parameter Number:	1650
Reference:	(R) 5.7.16

5.7.16.2 Raw Binary Data—This is a 1-byte parameter representing the value for 1 byte of memory. It can have any value between 0 and 255 (0 and FF_{16}) with no reserved values. The number of occurrences of this parameter within a message can range from 1 to 1784 (1 less than the 1785 transport limit to account for the parameter - Number of Occurrences of Raw Binary Data). When more than 7 occurrences are to be sent a transport protocol session must be used (remember 1 of the 8 message data bytes was used for the Number of Occurrences of Raw Binary Data parameter). The Number of Occurrences of Raw Binary Data parameter must be used to determine the message length when single packeted. In this case the Number of Occurrences of Raw Binary Data parameter provides the number of Raw Binary Data parameters being sent. This value plus 1 is the number of data bytes within the single packet message. When there are greater than 7 occurrences of the Raw Binary Data parameter to be sent, transport protocol will be needed and it will be necessary to send the Sequence Number (J1939-21 section 3.10.12) from the transport session. Hence the first transport packet will have the Sequence Number, along with the ‘Number of Occurrences of Raw Binary Data’ parameter and 6 occurrences of this (Raw Binary Data) parameter. In each subsequent transport packet there will be the Sequence Number and 7 occurrences of this (Raw Binary Data) parameter. The Sequence Number must be used to calculate the occurrence number of each of the Raw Binary Data parameters. Also as outlined in J1939-21 the last packet, although 8 bytes in length, may contain fewer than 7 occurrences of this parameter and the Total Message Size parameter (sent in the session connection message) must be used to identify when the end of data is reached. An example of the positioning of the Raw Binary Data within the messages is shown in Table 11 through Table 13. For an example of parsing memory with widths other than 8 bits, see 5.7.14.3.1

Data Length:	8 bits
Resolution:	Not applicable
Data Range:	0 to 255 (0 to FF_{16})
Type:	Status
Suspect Parameter Number:	1651
Reference:	5.7.16

TABLE 11 MESSAGE APPEARANCE WHEN MULTIPACKETED

Message	CAN ID	CAN DB1	CAN DB2	CAN DB3	CAN DB4	CAN DB5	CAN DB6	CAN DB7	CAN DB8
First Packet of a transport session	Transport Protocol - Data Transfer Message	Sequence Number J1939-21 3.10.12	Number of Occurrences of Raw Binary Data (value = FF_{16})	Raw Binary Data - #1	Raw Binary Data - #2	Raw Binary Data - #3	Raw Binary Data - #4	Raw Binary Data - #5	Raw Binary Data - #6
Second Packet of a transport session	Transport Protocol - Data Transfer Message	Sequence Number J1939-21 3.10.12	Raw Binary Data - #7	Raw Binary Data - #8	Raw Binary Data - #9	Raw Binary Data - #10	Raw Binary Data - #11	Raw Binary Data - #12	Raw Binary Data - #13
Last Packet of a transport session	Transport Protocol - Data Transfer Message	Sequence Number J1939-21 3.10.12	Raw Binary Data - #(Total Message Length - 2)	Raw Binary Data - #(Total Message Length - 1)	FF_{16}	FF_{16}	FF_{16}	FF_{16}	FF_{16}

TABLE 12 MESSAGE APPEARANCE - 7 OCCURRENCES OF RAW BINARY DATA (I.E. WITHOUT TRANSPORT)

Message	CAN ID	CAN DB1	CAN DB2	CAN DB3	CAN DB4	CAN DB5	CAN DB6	CAN DB7	CAN DB8
When not a transport session	Binary Data Transfer Message	Number of Occurrences of Raw Binary Data (value = 07_{16})	Raw Binary Data - #1	Raw Binary Data - #2	Raw Binary Data - #3	Raw Binary Data - #4	Raw Binary Data - #5	Raw Binary Data - #6	Raw Binary Data - #7

TABLE 13 MESSAGE APPEARANCE - 4 OCCURRENCES OF RAW BINARY DATA (I.E. WITHOUT TRANSPORT)

Message	CAN ID	CAN DB1	CAN DB2	CAN DB3	CAN DB4	CAN DB5	CAN DB6	CAN DB7	CAN DB8
When not a transport session with less than 7 occurrences of Raw Binary Data	Binary Data Transfer Message	Number of Occurrences of Raw Binary Data (value = 04_{16})	Raw Binary Data - #1	Raw Binary Data - #2	Raw Binary Data - #3	Raw Binary Data - #4	FF_{16}	FF_{16}	FF_{16}

(R) 5.7.17 BOOT LOAD DATA (DM17)—Used primarily to load boot data/program into a device when a memory access boot load command has been issued. A Memory Access State Transition Diagram (DM14 through DM18) is in (R) Appendix C. The CAN Data Length Code of the message is set to 8 bytes to deliberately avoid the use of transport protocol and thereby reduce the program overhead that would need to be functional within a Device while its program is being reloaded.

Transmission Rate:	As needed
Data Length:	8
Data Page:	0
PDU Format:	214
PDU Specific:	DA
Default Priority:	6
Parameter Group Number:	54784 ($00D600_{16}$)

BOOT_LOAD_DATA

Byte: 1-8	Boot Load Data	see 5.7.17.1

5.7.17.1 *Boot Load Data*—This is a 1-byte parameter using the same SLOT as the Raw Binary Data. There shall be 8 occurrences of this parameter in the message. The meaning of this parameter is proprietary. The structure used to reference the program and verify the data is also proprietary.

Data Length:	8 bits
Resolution:	Not applicable
Data Range:	0 to 255 (0 to FF_{16})
Type:	Status

Suspect Parameter Number:	1652
Reference:	5.7.17

(R) 5.7.18 DATA SECURITY (DM18)—The Data Security parameter group is used to send Security Entities of a given type and length. These entities are data produced by or used for applications of cryptography and supporting procedures to ensure data security. Also included is the provision to provide a Long Seed and Long Key to be used with Memory Access functions. The capabilities of the Long Seed and Key are outlined in (R) Appendix D. A Memory Access State Transition Diagram (DM14 through DM18) is in (R) Appendix C.

Transmission Rate:	As needed
Data Length:	Variable (8 to *n*)
Data Page:	0
PDU Format:	212
PDU Specific:	DA
Default Priority:	6
Parameter Group Number:	54272 ($00D500_{16}$)

DATA_SECURITY

Byte: 1	bits 8-1	Security Entity Length (Least significant 8 bits) (Bit 1 is least significant bit)	see 5.7.18.2
Byte: 2	bits 8-5	Security Entity Length (Most significant 4 bits) (Bit 8 is most significant bit)	see 5.7.18.2
	bits 4-1	Security Entity Type	see 5.7.18.1
Byte: 3-n		Data Security Parameter (Least significant byte is Byte 3) (Bit 1 is least significant bit) (Most significant byte is Byte *n*) (Bit 8 is most significant bit)	see 5.7.18.3

5.7.18.1 Security Entity Type—This 4-bit parameter that indicates whether the data in the following Security Entity parameter is to be used as a Long Seed, Long Key, Session Key, or Certificate (see Table 14.

Data Length:	4 bits
Resolution:	1 type/bit
Data Range:	0 to 15 (see Table 14)
Type:	Status
Suspect Parameter Number:	1479
Reference:	5.7.18

TABLE 14 SECURITY ENTITY TYPES

Bit States	Security Entity Type
0000_{02}	Data is Long Seed
0001_{02}	Data is Long Key
0010_{02}	Data is Session Key
0011_{02}	Data is Certificate
0100_{02} - 1111_{02}	Reserved for SAE Assignment

5.7.18.1.1 *DATA IS LONG SEED*—A Security Entity Type value of 0000_{02} implies that the data in the following Data Security parameter is to be used as a Long Seed. It is most likely then going from a Device to a Tool.

5.7.18.1.2 *DATA IS LONG KEY*—A Security Entity Type value of 0001_{02} implies that the data in the following Data Security parameter is to be used as a Long Key. In general this would imply previous receipt of a Long Seed upon which to base the Long Key. Also the direction would typically be from a Tool to a Device.

5.7.18.1.3 *DATA IS SESSION KEY*—A Security Entity Type value of 0010_{02} implies that the data in the following Data Security parameter is to be used as a Session Key. The Session Key is sent encrypted by using a secret key (symmetric encryption) or the public key of the addressed ECU (asymmetric encryption). The addressed ECU has to decrypt the Session Key before it can be used. The length of the decrypted Session Key is 8 bytes. In the case of using asymmetric encryption the Session Key is put into the first 8 bytes of the data string to be encrypted, followed by 8 bytes, each filled with FF_{16}, and arbitrary numbers for the remaining bytes. This provides a mechanism for the receiving ECU to check if its decryption was successful.

5.7.18.1.4 *DATA IS CERTIFICATE*—A Security Entity Type value of 0011_{02} implies that the data in the following Data Security parameter is to be used as a Certificate.

5.7.18.2 *Security Entity Length*—This 12-bit parameter contains the length, in bytes, of the Data Security Parameter.

Data Length:	12 bits
Resolution:	1 byte/bit
Data Range:	0 to 1785
Type:	Status
Suspect Parameter Number:	1596
Reference:	5.7.18

5.7.18.3 *Data Security Parameter*—This Parameter is used to send the data for the Data Security message. There are presently four different items defined. The Data Security Parameter shall be sent least significant byte first.

Data Length:	Variable (length given in the Security Entity Length parameter)
Resolution:	1 byte/bit
Data Range:	0 to 1785
Type:	Status
Suspect Parameter Number:	1597
Reference:	5.7.18

5.7.18.3.1 *LONG SEED*—When the Security Entity Type value is 0000_{02} the data is a Long Seed. The long seed is a number. The number is sent (sometimes randomly) when requesting message or application authentication to rule out replay attacks. (See (R) Appendix D.)

5.7.18.3.2 *LONG KEY*—When the Security Entity Type value is 0001_{02} the data is a Long Key. The long key is a number. This number represents a mathematical function of a previously received Long Seed sent when attempting to justify one's request for a message or application. (See (R) Appendix D.)

5.7.18.3.3 *SESSION KEY*—When the Security Entity Type value is 0010_{02} the data is a Session Key. In this application the Long Seed/Key Data Parameter must be interpreted only if it contains a Session Key.

5.7.18.3.4 *CERTIFICATE*—When the Security Entity Type value is 0011_{02} the data is a Certificate. Parameter group to be sent on request from an ECU authorized by a Certification Authority to send authentication messages. Acceptance of the certificate is a prerequisite for the receiving unit to send a Session Key. The certificate is only needed when the establishment of a Session Key is based on an asymmetric encryption procedure. For symmetric encryption the installation of the secret key and the algorithm used is not specified here. The content of the certificate is given by ISO/IEC 9594-8 with the subject being the sender of the message. The certificate contains the Public Key of the sender.

(R) 5.7.19 CALIBRATION INFORMATION (DM19)—Provides information about the calibration to an interrogating Tool (see Figure 4).

Transmission Rate:	On request using PGN 59904 (See SAE J1939-21 PGN 59904). A NACK is required if PG is not supported. (See SAE J1939-21 PGN 59392)	
Data Length:	20 bytes	
Function:	Provide information about the calibration to scan Tool	
Data page:	0	
PDU Format:	211	
PDU Specific:	DA	
Default Priority:	7	
Parameter Group Number:	54016 ($00D300_{16}$)	
Bytes 1-4:	Calibration Verification Number (Byte 1 is least significant byte)	see 5.7.19.1
Bytes 5-20:	Calibration ID (Byte 5 is least significant byte)	see 5.7.19.2

5.7.19.1 *Calibration Verification Number*—Four-byte checksum of the entire calibration. Includes code and data. Excludes parameters that exist only in RAM, nonvolatile parameters that change values during the life cycle of the module (hours of operation, miles, number of on/off cycles, freeze frame data, etc.), or nonemissions-related parameters that may be changed by the operator (offsets for real-time clocks, user selectable preferences, etc.). If the checksum is less than 4 bytes, it must be padded with 00_{16} (the 00_{16} pad is placed in the most significant byte(s) when needed). The checksum algorithm shall be more robust than a two's complement checksum.

Data Length:	4 bytes
Resolution:	Not applicable
Data Range:	0 to 4,294,967,295 (00 00 00 00_{16} to FF FF FF FF_{16})
Type:	Hexadecimal
Suspect Parameter Number:	1634
Reference:	5.7.19

5.7.19.2 *Calibration Identification*—Sixteen-byte calibration identification number. Uniquely identifies the software installed in the control module. The calibration ID must be unique, but does not need to be 16 bytes long. If the Calibration ID is less than 16 bytes, those unused bytes are reported at the end of the calibration ID as 00_{16} (the 00_{16} pad is placed in the least significant bytes of the Calibration Identifier when needed). The 00_{16} if needed is added to the end of the ASCII character string for Calibration Identification.

Data Length:	16 bytes
Resolution:	Not applicable
Data Range:	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00_{16} to FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF_{16}
Type:	ASCII
Suspect Parameter Number:	1635
Reference:	5.7.19

EXAMPLE: The following example shows how a 3-byte Calibration Verification Number $ABCDEF_{16}$ and a 10-byte Calibration ID “CONTENDER1” would be sent. The hexadecimal representation for the ASCII “CONTENDER1” is

ASCII:	C	O	N	T	E	N	D	E	R	1
Hex::	43	4F	4E	54	45	4E	44	45	52	31

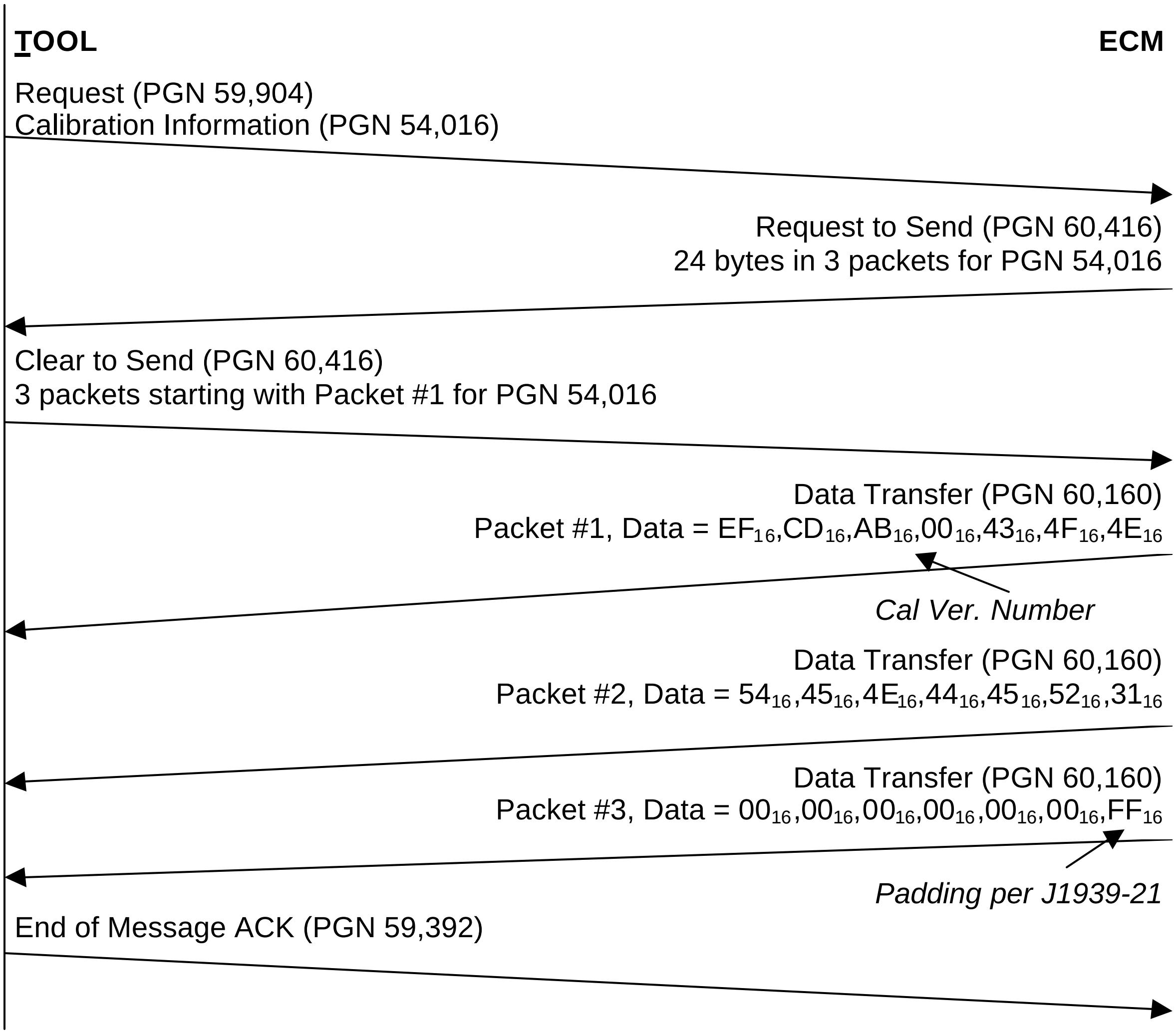


FIGURE 4 EXAMPLE OF TOOL REQUESTING THE CALIBRATION ID AND CVN FROM AN ECM

NOTE—The Calibration Verification Number is sent LSB-MSB per J1939-71 practice for numeric values so the last byte is the padding per J1979. Note also that Calibration ID is sent per J1939-71 standard practice for ASCII values. Finally, note that the entire calibration information PGN is 20 bytes long so the last byte in the data transfer is FF_{16} per J1939-21.

PREPARED BY THE SAE TRUCK AND BUS CONTROL AND COMMUNICATIONS SUBCOMMITTEE OF THE SAE TRUCK AND BUS ELECTRICAL AND ELECTRONICS COMMITTEE

Appendix A Failure Mode Identifier Codes

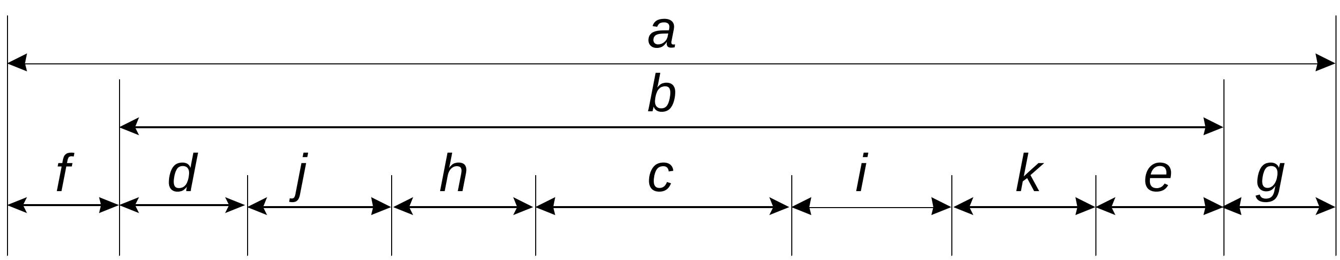
A.1 The following definitions shall be applicable when using FMIs. Examples have been included to help achieve consistent usage of the failure mode identifiers.

A.1.1 Assumptions and Definitions Used for the FMI Definitions

Data—Any information pertaining to physical conditions that is communicated to an electronic module in the form of voltage, current, PWM signals, or data streams.

Real World—Mechanical parameters or operating conditions that can be measured in the form of voltage, current, PWM signals, data streams, etc.

Signal Range Definitions—(See Figure A.1.)



Region a   Total signal input range possible that can be seen by an electronic module.

Region b   Total signal range physically possible as defined by an application.

Region c   Range defined as normal for a given real world measurement.

Region d   Range defined as below normal, Most Severe Level, of what is considered normal for the given real world measurement.

Region e   Range defined as above normal, Most Severe Level, of what is considered normal for the given real world measurement.

Region f   Range which is low outside the range of what is considered physically possible for a given system, indicating a short to a low source has occurred.

Region g   Range which is high outside the range of what is considered physically possible for a given system, indicating a short to a high source has occurred.

Region h   Range defined as below normal, Least Severe Level, of what is considered normal for a given real-world measurement.

Region i   Range defined as above normal, Least Severe Level, of what is considered normal for a given real-world measurement.

Region j   Range defined as below normal, Moderately Severe Level, of what is considered normal for a given real-world measurement.

Region k   Range defined as above normal, Moderately Severe Level, of what is considered normal for a given real-world measurement.

FIGURE A.1 SIGNAL RANGES

A.1.2 FMI and Description

A.1.2.1 FMI=0—DATA VALID BUT ABOVE NORMAL OPERATIONAL RANGE - MOST SEVERE LEVEL

The signal communicating information is within a defined acceptable and valid range, but the real world condition is above what would be considered normal as determined by the predefined most severe level limits for that particular measure of the real world condition (*Region e* of the signal range definition). Broadcast of data values is continued as normal.

A.1.2.2 FMI=1—DATA VALID BUT BELOW NORMAL OPERATIONAL RANGE - MOST SEVERE LEVEL

The signal communicating information is within a defined acceptable and valid range, but the real world condition is below what would be considered normal as determined by the predefined least severe level limits for that particular measure of the real world condition (*Region d* of signal range definition). Broadcast of data values is continued as normal.

A.1.2.3 FMI=2—DATA ERRATIC, INTERMITTENT OR INCORRECT

Erratic or intermittent data includes all measurements that change at a rate that is not considered possible in the real world condition and must be caused by improper operation of the measuring device or its connection to the module. Broadcast of data value is substituted with the “error indicator” value.

Incorrect data includes any data not received and any data that is exclusive of the situations covered by FMIs 3, 4, 5 and 6 as follows in A.1.2.4 through A.1.2.7. Data may also be considered incorrect if it is inconsistent with other information collected or known about the system.

A.1.2.4 FMI=3—VOLTAGE ABOVE NORMAL, OR SHORTED TO HIGH SOURCE

a. A voltage signal, data or otherwise, is above the predefined limits that bound the range *(Region g* of the signal range definition). Broadcast of data value is substituted with the “error indicator” value.

b. Any signal external to an electronic control module whose voltage remains at a high level when the ECM commands it to low. Broadcast of data value is substituted with the “error indicator” value.

A.1.2.5 FMI=4—VOLTAGE BELOW NORMAL, OR SHORTED TO LOW SOURCE

a. A voltage signal, data or otherwise, is below the predefined limits that bound the range *(Region f* of the signal range definition). Broadcast of data value is substituted with the “error indicator” value.

b. Any signal external to an electronic control module whose voltage remains at a low level when the ECM commands it to high. Broadcast of data value is substituted with the “error indicator” value.

A.1.2.6 FMI=5—CURRENT BELOW NORMAL OR OPEN CIRCUIT

a. A current signal, data or otherwise, is below the predefined limits that bound the range (*Region g* of the signal range definition). Broadcast of data value is substituted with the “error indicator” value.

b. Any signal external to an electronic control module whose current remains off when the ECM commands it on. Broadcast of data value is substituted with the “error indicator” value.

A.1.2.7 FMI=6—CURRENT ABOVE NORMAL OR GROUNDED CIRCUIT

a. A current signal, data or otherwise, is above the predefined limits that bound the range (*Region g* of the signal range definition). Broadcast of data value is substituted with the “error indicator” value.

b. Any signal external to an electronic control module whose current remains on when the ECM commands it off. Broadcast of data value is substituted with the “error indicator” value.

A.1.2.8 FMI=7—MECHANICAL SYSTEM NOT RESPONDING OR OUT OF ADJUSTMENT

Any fault that is detected as the result of an improper mechanical adjustment or an improper response or action of a mechanical system that, with a reasonable confidence level, is not caused by an electronic or electrical system failure. This type of fault may or may not be directly associated with the value of general broadcast information.

A.1.2.9 FMI=8—ABNORMAL FREQUENCY OR PULSE WIDTH OR PERIOD

To be considered in cases of FMI 4 and 5. Any frequency or PWM signal that is outside the predefined limits which bound the signal range for frequency or duty cycle (outside *Region b* or the signal definition). Also if the signal is an ECM output, any signal whose frequency or duty cycle is not consistent with the signal which is emitted. Broadcast of data value is substituted with the “error indicator” value.

A.1.2.10 FMI=9—ABNORMAL UPDATE RATE

Any failure that is detected when receipt of data via the data link or as input from a smart actuator or smart sensor is not at the update rate expected or required by the ECM (outside *Region c* of the signal range definition). Also any error that causes the ECM not to send information at the rate required by the system. This type of fault may or may not be directly associated with the value of general broadcast information.

A.1.2.11 FMI=10—ABNORMAL RATE OF CHANGE

Any data, exclusive of the abnormalities covered by FMI 2, that is considered valid but whose data is changing at a rate that is outside the predefined limits that bound the rate of change for a properly functioning system (outside *Region c* of the signal range definition). Broadcast of data values is continued as normal.

A.1.2.12 FMI=11—ROOT CAUSE NOT KNOWN

It has been detected that a failure has occurred in a particular subsystem but the exact nature of the fault is not known. Broadcast of data value is substituted with the “error indicator” value.

A.1.2.13 FMI=12—BAD INTELLIGENT DEVICE OR COMPONENT

Inconsistency of data indicates that a device with some internal intelligence, such as a controller, module, smart sensor or smart actuator, is not properly functioning. This data may be internal to a module or external from a data link message or from various system responses. Broadcast of data value is substituted with the “error indicator” value. This error is to include all internal controller trouble codes that can not be caused by connections or systems external to the controller.

A.1.2.14 FMI=13—OUT OF CALIBRATION

A failure that can be identified to be the result of not being properly calibrated. This may be the case for a subsystem which can identify that the calibration attempting to be used by the controller is out of date. Or it may be the case that the mechanical subsystem is determined to be out of calibration. This failure mode does not relate to the signal range definition as do many of the FMIs.

A.1.2.15 FMI=14—SPECIAL INSTRUCTIONS

SPNs 611 through 615 are defined as “System Diagnostic Codes” and are used to identify failures that cannot be tied to a specific field replaceable component. Specific subsystem fault isolation is the goal of any diagnostic system, but for various reasons this cannot always be accomplished. These SPNs allow the manufacturer some flexibility to communicate non-“specific component” diagnostic information. Since SPNs 611-615 use the standard SPN/FMI format it allows the use of standard diagnostic tools, electronic dashboards, satellite systems and other advanced devices that scan Parameter Groups containing the SPN/FMI formats. Because manufacturer defined codes are not desirable in terms of standardization, the use of these codes should only occur when diagnostic information cannot be communicated as a specific component and failure mode.

Possible reasons for using a System Diagnostic Code include:

1. Cost of specific component fault isolation is not justified, or
2. New concepts in Total Vehicle Diagnostics are being developed, or
3. New diagnostic strategies that are not component specific are being developed.

Due to the fact that SPNs 611-615 are manufacturer defined and are not component specific, FMIs 0-13 and 15-31 have little meaning. Therefore, FMI 14, “Special Instructions”, is usually used. The goal is to refer the service personnel to the manufacturer's troubleshooting manual for more information on the particular diagnostic code. This failure mode does not relate to the signal range definition as do many of the FMIs. This type of fault may or may not be directly associated with the value of general broadcast information.

A.1.2.16 FMI=15—Data Valid but Above Normal Operating Range - Least Severe Level

The signal communicating information is within a defined acceptable and valid range, but the real world condition is above what would be considered normal as determined by the predefined least severe level limits for that particular measure of the real world condition (*Region i* of signal range definition). Broadcast of data values is continued as normal.

A.1.2.17 FMI=16—Data Valid but Above Normal Operating Range - Moderately Severe Level

The signal communicating information is within a defined acceptable and valid range, but the real world condition is above what would be considered normal as determined by the predefined moderately severe level limits for that particular measure of the real world condition (*Region k* of signal range definition). Broadcast of data values is continued as normal.

A.1.2.18 FMI=17—Data Valid but Below Normal Operating Range - Least Severe Level

The signal communicating information is within a defined acceptable and valid range, but the real world condition is below what would be considered normal as determined by the predefined least severe level limits for that particular measure of the real world condition (*Region h* of signal range definition). Broadcast of data values is continued as normal.

A.1.2.19 FMI=18—Data Valid but Below Normal Operating Range - Moderately Severe Level

The signal communicating information is within a defined acceptable and valid range, but the real world condition is below what would be considered normal as determined by the predefined moderately severe level limits for that particular measure of the real world condition (*Region j* of signal range definition). Broadcast of data values is continued as normal.

A.1.2.20 FMI=19—Received Network Data In Error

Any failure that is detected when the data received via the network is found substituted with the “error indicator” value (i.e. FE_{16}, see J1939-71). This type of failure is associated with received network data. The component used to measure the real world signal is wired directly to the module sourcing the data to the network and not to the module receiving the data via the network. This FMI is applicable to *Regions f* and *g* of the signal range definition. This type of fault may or may not be directly associated with the value of general broadcast information.

A.1.2.21 FMI=20-30—RESERVED FOR SAE ASSIGNMENT

A.1.2.22 FMI=31—NOT AVAILABLE OR CONDITION EXISTS

Used to indicate that the FMI is not available or that the condition that is identified by the SPN exists. When no applicable FMI exists for the reported SPN, FMI 31 can be used. Also in cases when the reported SPN name has the failure information in it, FMI 31 can be used to indicate that the condition reported by the SPN exists. This type of fault may or may not be directly associated with the value of general broadcast information.

(R) Appendix B Assumptions Used To Design Memory Access

B.1 Assumptions used in the design of Memory Access

B.1.1 Memory data is transferred in byte pieces and if the memory width is other than an integer number of bytes an extra full byte is used to contain the remaining bits.

B.1.2 It would be useful to have a direct address into memory, as well as, a spatial (object or symbolic) referencing address. (As an example: a single 24-bit address would suffice for the direct address while 256 16-bit addresses could divide space and the standard could predefine the meaning of the first 128 spaces, while allowing the users to define and use the other 128 spaces proprietarily. It appears a 5-bit space identifier and a 19-bit object identifier would work since it would allow referencing SPNs directly—although a different length may be ultimately chosen—in fact presently the pointer is 24 bits while the pointer extension is 8 bits.)

B.1.3 It is desired to generate a memory access function without adding another transport protocol capable of handling more than 1785 bytes to the standard; thereby data transfers are limited to lengths under 1785 bytes (J1939-21 section 3.10.1.1).

B.1.4 Several security types must be handled to satisfy all users. They are:

B.1.4.1 No security

B.1.4.2 Password form of security

B.1.4.3 Re-entrant security, which the manufacturer may optionally chose to implement, where the device allows multiple operations after a security level has been established

B.1.4.4 Some more elaborate scheme similar to Seed/Key

B.1.4.5 A User_Level request, which controls the User's privileges with the option for further security

B.1.4.6 A means of increasing the effective Seed/Key size by requiring multiple iterations and/or mathematically combining the Seeds and Keys

B.1.5 Minimum number of new PGNs would be preferred (so that filtering and software overhead are minimized), so items that are time/message independent are combined (overlaid) to reduce the message set. (Obviously the message set can be extended if the overlaying appears too complex or is desired for any other reason.)

B.1.6 Prefer single packet messages for the Memory Access invocation and control to reduce software overhead and improve speed of interchange, while need multipacketed messages for data transfer to provide reasonable lengths and improve transfer efficiency.

B.1.7 Reprogramming of 'program memory' could be handled by any one of 3 general choices:

B.1.7.1 Use of a Write operation in combination with some form of execution control table for enabling/disabling execution within sections of the program memory that are being modified in combination with a hardware configuration such that writing to these sections of program does not interfere with operation of other sections of program.

B.1.7.2 A Boot Loader approach where a proprietary program for reloading executable memory is loaded using the memory access operation of the standard and execution is then transferred to this proprietary reloading program. There is no need to standardize the data transfer utilized by or the operation of this proprietary reloading program, but only the memory access operation loading said program and transferring control to it.

B.1.7.3 A completely proprietary technique, which is already possible using other features of this network standard.

B.1.8 Memory need only be addressed in one direction. Assume start at the lowest address and operate toward a higher address for this proposal (i.e. only an incrementing pointer is provided).

B.1.9 Also assume that for multipacket data sets the transport packet number must be combined with the pointer provided in the original memory access to decode the address(es) for each packet.

B.1.10 All Memory Access Requests originate at a Tool and are considered commands to the device. The device however controls whether the request is handled.

B.1.11 Design to provide access for a single 'Tool' to access a single 'Device'. Then later if it is desired, one can allow any node to function as a 'Tool' communicating with any other node, functioning as a 'Device'. Also if an OEM desires to allow more than one Tool to access their Device simultaneously all they need additional is software to handle the different accesses.

B.2 Assumptions for Data Security

B.2.1 More of the committee members desired to use two messages over a single message, which at times was single-frame and at other times multiframe, necessitating transport session.

B.2.2 A single message containing either a Seed or a Key is better than a separate message for Seed and another for Key, since it uses fewer PGNs.

B.2.3 A length parameter, while not inherently required, simplifies software handling enough to warrant inclusion.

B.2.4 No need to pack these parameters, as it still takes a minimum of 5 frames to send any Seed or Key with a length between 8 and 13 bytes, so leave separate for ease of parsing.

(R) Appendix C Application Rules Regarding Memory Access Pgns

Memory Access State Transition Diagram

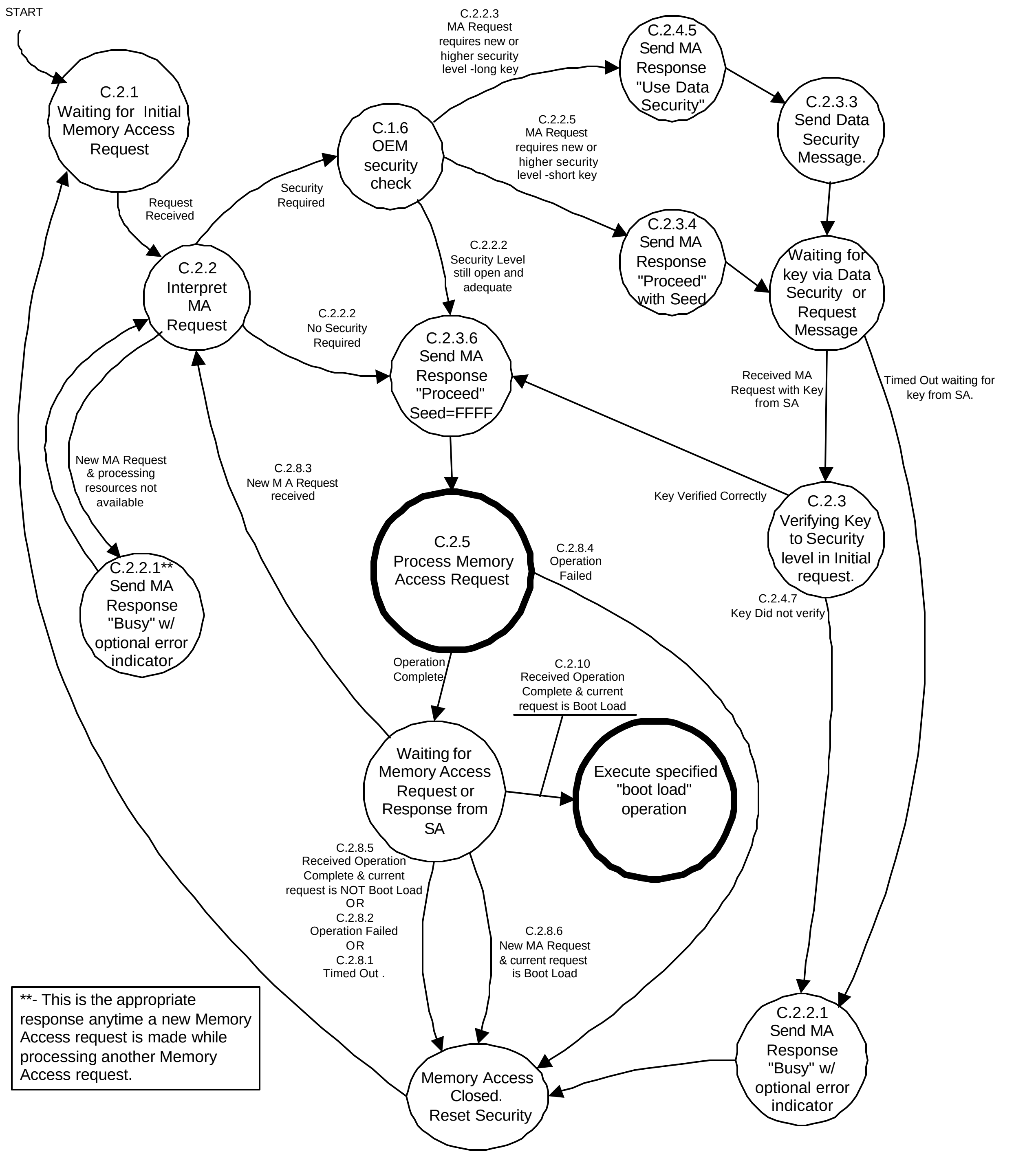


FIGURE C.1 MEMORY ACCESS STATE TRANSITION DIAGRAM

C.1 General Rules

The following general rules must be considered:

C.1.1 Only Memory Access Operations initiated by a Tool (using Memory Access Request) are required to be honored. However, the manufacturer may chose to allow any network node to operate as either a Tool or Device, as long as it meets the functions presented in Appendix C for Tool or Device.

C.1.2 A Device is required to support only one session of Memory Access at any one time (it may therefore reject all other requests with Status of Busy).

C.1.3 A Tool may be designed to initiate Memory Access Operations with more than one Device at any given time.

C.1.4 There will be no specific messages to:

C.1.4.1 “Undo” a write request.

C.1.4.2 Abort an operation (obviously, failure to transfer data, etc. will cause a failure which could be construed an abort).

C.1.5 A Device may impose any number of additional constraints on when memory access requests are honored (see 5.7.14.4).

C.1.6 A manufacturer may choose to allow their Device(s) to allow re-entrant security, wherein a Tool which has already made a Memory Access Request and established a security level may send additional Memory Access Requests following the successful completion of the present operation, using the established security.

C.1.7 The Device needs time-out functions for:

C.1.7.1 Failure to receive further security from a Tool when the Device has required same

C.1.7.2 Failure to receive a complete transfer of the data set once an operation was allowed

C.1.7.3 Failure to hear a close from a Tool

C.1.8 A Tool needs time-out functions for:

C.1.8.1 Failure to receive a Memory Access Response from a Device to which it has sent a Request

C.1.8.2 Failure to receive a complete transfer of the data set once a read operation was allowed

C.1.8.3 Failure to hear a close from a Device

C.2 The following is a narrative of a typical application of this protocol:

It is only required that Memory Access operations be available once a node has become operational upon the network and satisfied any manufacturer specific interlock requirements. Software functions which will need to be finished before Memory Access becomes available include Address claiming, updating of instance fields within the NAME and any other configuration matters that the manufacturer deems necessary as a precursor to allowing operation of the Memory Access software. A diagram showing the memory access state transitions for a device has been included along with message transmission diagrams for several cases ((R) Appendix E). These diagrams should be used along with the following text to generate the software modules for a device.

There is presently no diagram for a Tool and the text and message transition diagrams in (R) Appendix E should be used as the reference in designing the Tool's software.

C.2.1 Initial Memory Access Request - The Tool sends a Memory Access Request to the Device. This consists of the address of the memory within the Device to be accessed (Pointer, Pointer Extension, and Pointer Type), the length of the memory the Tool desires to operate upon (Length/Number Requested), the operation requested (Command = Erase, Read, Write, Boot Load, or EDCP Generation), and if utilized by the Device any necessary User_Level or Password information within the Key/User_Level parameter. If needed, based upon the device's particular requirements, it extracts from the Message Identifier (J1939-21 Section 3.1): the source (J1939-21 Section 3.2.6) and destination (J1939-21 Sections 3.2.4, 3.2.5, and 3.2.5.1).

C.2.2 Device response to initial Memory Access Request - The device responds to this request with a Memory Access Response as follows:

C.2.2.1 If the Device is busy or has identified an error within the request (such as the pointer is not on a memory boundary for the memory being selected, the space being Reserved, etc.), the Device transmits a Seed of all 1's ($FFFF_{16}$) and a Status of Busy with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension, and the Length/Number Allowed to be '0'. The Number Allowed needs not be interpreted by the Tool, as it has no specific meaning in the context of this message. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Tool needs to try again later. Go to C.2.3.1.

C.2.2.2 If the Device is not busy, and no security is required, or was established in a previous operation (as would occur when the manufacturer has allowed re-entry to the memory access operation for a Tool which has as yet not issued a 'close'), or the Password transmitted has been accepted, the Device transmits the allowed number of objects or memory length within Length/Number Allowed, a Seed of all 1's ($FFFF_{16}$) to indicate no further Key is required (5.7.15.4 and Table 10), and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter (most likely 00_{16} since no error) as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The requested operation can begin. (Remember that for the optional manufacturer re-entry the device may have other established rules regarding whether the re-entrant operation was truly at the security level previously established.) Go to C.2.5.

C.2.2.3 If the Device is not busy, and 'long' Seed/Key security is required and if a valid User_Level was provided, (when utilized by the device), the Device transmits a Length/Number Allowed of 0, a Seed equal to 1 (0001_{16}) and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter (most likely 00_{16} since no error) as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. Memory Access may continue. Go to C.2.3.3.

C.2.2.4 If the Device is not busy, but security was required and some security violation or error has occurred (such as: an invalid User_Level or Password), the Device transmits a Seed of all 1's ($FFFF_{16}$) and a Status of Busy, with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the

Error Indicator/EDC Parameter as appropriate, and the Length/Number Allowed to be '0'. The Number Allowed needs not be interpreted by the Tool, as it has no specific meaning in the context of this message. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Tool needs to try again later. Go to C.2.3.1.

C.2.3 Tools action on security response - The Tool responds to the Memory Access Response message(s) controlling security of a Memory Access Operation in one of several ways. (Remember that the Tool always has a choice of how to handle the EDCP Extension and Error Indicator/EDC Parameter. While the Tool is NOT required to assign any meaning to these items, it may optionally (at the manufacturer's discretion) make use of the assigned values and definitions (see 5.7.15.3) to imply specific meanings. This supposes that the device being communicated with has also chosen to use the EDCP Extension and Error Indicator/EDC Parameter to indicate error conditions. The responses are as follows:

C.2.3.1 If the Tool receives a Memory Access Response with a Status of Busy, it needs to try the request again later, unless the busy was really indicating an error in the Request. If the manufacturer has provided diagnostics of such errors this will be indicated by the EDCP Extension and the error identification will be within the Error Indicator/EDC Parameter. If there was an identified error the Tool could then chose to correct the 'problem' and issue another request. (It is felt that some manufacturers will wish to provide no further indication of invalid security, as this would only aide those trying to defeat the security. This is their choice - 5.7.15.3.) Go to C.2.1.

C.2.3.2 If the Tool sees a Memory Access Response with a Status of Proceed, a Length/Number Allowed of 0, and a Seed equal to all 0's (0000_{16}), then the Seed (5.7.15.4) has been sent previously by the device and the device is expecting the Tool to begin sending the Key corresponding to the Seed (using another Memory Access Request message). This request should contain the Key based upon the received Seed, plus all of the Memory Access Request parameters (Pointer Type, Pointer Extension, Pointer, Length/Number Requested, and Command) that were in the initial Request. Go to C.2.4.

C.2.3.3 If the Tool sees a Memory Access Response with a Status of Proceed, a Length/Number Allowed of 0, and a Seed equal to 1 (0001_{16}), then a 'Long' Seed and Key are to be used (see also Data Security message document). The Tool should now expect a Data Security message (with a Long Seed). Following the receipt of a Long Seed from a Data Security message, the Tool should reply with the corresponding Long Key using another Data Security message. The Device then answers the Tool with another of the messages identified here in section C.2.3. Go to C.2.3.

C.2.3.4 If the Tool sees a Memory Access Response with a Status of Proceed, a Length/Number Allowed of 0, and a Seed not equal to 0, 1, or all 1's (0000_{16}, 0001_{16}, or $FFFF_{16}$) then this is the Seed from the device. The Tool may now begin sending the Key corresponding to the Seed, using another Memory Access Request message. This request should contain the Key based upon the received Seed, plus all of the other Memory Access Request parameters (Pointer Type, Pointer Extension, Pointer, Length/Number Requested, and Command) that were in the initial Request. Go to C.2.4.

C.2.3.5 If the Tool sees a Memory Access Response with a Status of Proceed, a Length/Number Allowed of 0, and a Seed equal to all 1's ($FFFF_{16}$), then the device feels the Key transfer has been completed, but that the Key verification is not completed (or some other similar delay) and the operation can not yet begin. There may have been an Error Indicator in the Error Indicator/EDC Parameter, at the manufacturer's choice (5.7.15.3). The Tool must not begin data transfer yet (if there is to be one). The Tool should in general send another Memory Access Request to the Device, with a Key of all 1's ($FFFF_{16}$) plus all of the other Memory Access Request parameters (Pointer Type, Pointer Extension, Pointer, Length/Number Requested, and Command) that were in the initial Request. Go to C.2.4. However, if the Tool is waiting for data from the Device, it may chose simply to continue waiting instead of sending another Request. Go to C.2.5.

C.2.3.6 If the Tool sees a Memory Access Response with a Status of Proceed, a non-zero Length/Number Allowed, and a Seed equal to all 1's ($FFFF_{16}$), then the device feels the data transfer may begin. The Tool should consider the device is now ready to begin the requested operation. Go to C.2.5.

C.2.4 If the Device has not previously signaled that it was Busy, it responds to the next Memory Access Request with a Memory Access Response as follows:

C.2.4.1 If the Device has become busy, the Device transmits a Seed of all 1's ($FFFF_{16}$) and a Status of Busy with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension, and the Length/Number Allowed to be '0'. The Number Allowed needs not be interpreted by the Tool, as it has no specific meaning in the context of this message. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Tool needs to try again later. Go to C.2.3.1 to see Tool's action.

C.2.4.2 If the Device is still not busy, and security was required (including receipt of a valid User_Level, if it was required), and the device feels a complete Key has been received, thus requiring no additional Seed/Key combinations, but the device has as yet been unable to complete the verification of the Key, the Device transmits a zero for Length/Number Allowed, a Seed of all 1's ($FFFF_{16}$), and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. Go to C.2.3 to see Tool's action.

C.2.4.3 If the Device is still not busy, and security was required (including receipt of a valid User_Level, if it was required), and the device feels a complete Key has been received, thus requiring no additional Seed/Key combinations, and the device has validated (accepted) the Key, the Device transmits a nonzero Length/Number Allowed (with the value representing the actual length the device is willing to allow the Tool to operate upon), a Seed of all 1's ($FFFF_{16}$), and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The requested operation can begin. Go to C.2.3 to see Tool's action.

C.2.4.4 If the Device is still not busy, and security was required (including receipt of a valid User_Level, if it was required), but that the Seed has NOT been sent yet and the use of a Long Seed/Key (see (R) 5.7.18) is NOT required, the Device transmits a Length/Number Allowed of 0, a Seed not equal to either all 0's or all 1's (0000_{16} or $FFFF_{16}$) or '1' (implying use Long Seed see Table 10) and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. Memory Access may continue. Go to C.2.3 to see Tool's action.

C.2.4.5 If the Device is still not busy, and security was required (including receipt of a valid User_Level, if it was required), but the use of a LONG SEED/KEY (see (R) 5.7.18) is required, the Device transmits a Length/Number Allowed of 0, a Seed equal to 1 (0001_{16}) and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Device should also transmit a Data Security message with an appropriate Seed (outlined within the Data Security message documentation). Memory Access may continue. Go to C.2.3 to see Tool's action.

C.2.4.6 If the Device is still not busy, and security was required (including receipt of a valid User_Level if it was required), and the Seed has been sent but reception of the Key has NOT occurred, and the device has timed-out waiting for the Tool, the Device may transmit another Memory Access Response message with a Length/Number Allowed of 0, a Seed equal to all 0's (0000_{16}) and a Status of Proceed with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. Memory Access may continue. Go to C.2.3 to see Tool's action. Alternately the device may chose to discontinue the operation. Go to C.2.10.

C.2.4.7 If the Device is still not busy, and security was required, and an invalid Key was received, the Device transmits a Seed of all 1's ($FFFF_{16}$) and a Status of Busy with the EDCP Extension set to either No Error Indicator/EDC Parameter Available (FF_{16}) or Data in Error Indicator/EDC Parameter is an Error Indicator (06_{16} or 07_{16}) as desired by the manufacturer, the Error Indicator/EDC Parameter as appropriate for the EDCP Extension, and the Length/Number Allowed to be '0'. The Number Allowed needs not be interpreted by the Tool, as it has no specific meaning in the context of this message. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Tool must try again later if it desires to obtain the requested action. Go to C.2.3.1 to see Tool's action.

C.2.5 Begin a requested Memory Access Operation, when the Tool has seen a Status of Proceed and a Seed equal to all 1's ($FFFF_{16}$), then it recognizes that the Device is willing to allow the requested Memory Access Operation to begin. The device should have retained any internal state information indicating that it has signaled the Tool of its own readiness to allow said operation. The next step depends upon the type of Operation initiated with the Command parameter of the initial Memory Access Request. (Note: The Command and the Length/Number Requested within the Memory Access Request message(s) should have been either constant or changed to what the device was willing to allow (C.2.4.3), any other alteration during the sequence should cause the Device to reject the operation - see 5.7.14.4.6.)

C.2.5.1 If the Memory Access Request Command was an Erase, the Device should process the Erase command that it allowed and when completed initiate the Close Sequence. Go to C.2.6.

C.2.5.2 If the Memory Access Request Command was a Read, the Tool allows the Device to initiate a transfer using the Binary Data Transfer PGN, either as a single packet or as a multipacketed message within a transport session depending upon the Length involved (If a transport session is required it follows the rules in SAE J1939-21). When the transfer is completed, the Device initiates the Close Sequence. Go to C.2.6.

C.2.5.3 If the Memory Access Request Command was a Write or a Boot Load with data (non-zero Length / Number Requested), the Device allows the Tool to initiate a transfer using the Binary Data Transfer PGN, either as a single packet or as a multipacketed message within a transport session depending upon the Length involved (if a transport session is required, it follows the rules in SAE J1939-21). When the transfer is completed and when the write operation has finished (successfully or not), the Device initiates the Close Sequence. If the command was a Boot Load without data (zero Length / Number Requested), the Device should initiate the Close Sequence exactly as when a data transfer had completed. Go to C.2.6.

C.2.5.4 If the Memory Access Request Command was an EDCP Generation, the Device reads the data from the length of memory at the address it has allowed access to and then generate the requested checksum (or CRC, etc.) for these locations. When the checksum is generated, the Device initiates the Close Sequence. Go to C.2.6.

C.2.6 Memory Access Close Sequence is initiated upon completion of a Memory Access operation, as follows:

C.2.6.1 If the Memory Access Request Command was an Erase, Write, Boot Load, or EDCP Generation, the Device transmits a Memory Access Response with a Status of Operation Completed or Operation Failed depending upon the success/failure of the requested operation. The EDCP Extension identifies whether the Error Indicator/EDCP is used (remember this is at the manufacturer's discretion). It also identifies how to interpret itself and said Error Indicator/EDC Parameter. The Length/Number Allowed should be 0, the Seed should be equal to all 0's (0000_{16}) and the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to these items, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Tool may interpret the Length/Number Allowed and Seed parameters as having no meaning. This Memory Access Response is to be transmitted only after any internal processes invoked by the Memory Access have completed. Go to C.2.7.

C.2.6.2 If the Memory Access Request Command was a Read, the Device transmits a Memory Access Response with a Status of Operation Completed or Operation Failed depending upon the success/failure of the requested operation. The EDCP Extension identifies whether the Error Indicator/EDCP is used and/or how to handle itself and said Error Indicator/EDC Parameter. The Length/Number Allowed should be 0, the Seed should be equal to all 0's (0000_{16}) and the Error Indicator/EDC Parameter as appropriate for the EDCP Extension. The Tool has a choice with handling of the EDCP Extension and Error Indicator/EDC Parameter. The Tool is NOT required to assign any meaning to the EDCP Extension and Error Indicator/ EDC Parameter, but may optionally (at the manufacturer's discretion) use the assigned values and definitions to imply specific meanings. The Tool may interpret the Length/Number Allowed and Seed parameters as having no meaning (it may also chose to simply dispose of them). This Memory Access Response is transmitted immediately after the Binary Data Transfer has completed (Transport Session has closed if one was required for the data transfer). Go to C.2.7.

C.2.7 When the Tool receives the Memory Access Response from the Device indicating operation completed or failed, and the Tool wishes to end the memory access connection, it transmits a Memory Access Request indicating a status of either Operation Completed or Operation Failed, from its perspective. (The Tool should have checked the EDCP Extension and Error Indicator/EDC Parameter as a part of its decision process). The Tool should send another Memory Access Request to the Device, with a Key of all 1's ($FFFF_{16}$) plus all of the other Memory Access Request parameters (Pointer Type, Pointer Extension, Pointer, Command and Length/Number Allowed) that were in the initial Request (exception that the Length/Number Requested may also be the value it changed to when the device indicated it was willing to allow the operation (C.2.4.3)). The Device may treat all of these other parameters as having no meaning. In the case of Boot Load, a Memory Access Request of Operation Failed from the Tool shall prevent the Device from transferring execution to the address specified within the original Boot Load request.

C.2.8 After the Device initiates the Close Sequence, it waits for a Memory Access Request from the Tool. The action taken by the Device depends upon the original request, as well as the Response from the Tool. The following are the possible Device actions. Note: It is expected that only a successful completion of the execution of the Boot Load Command prevents the system from returning to the same operational mode it was in prior to the Memory Access Request which initiated a Boot Load operation (see C.2.9).

C.2.8.1 If there is no response from the Tool within 100 ms (tolerance of ±25 ms), plus any additional delay needed to account for the delay of any bridges within the system, of the Device transmission, the Device shall reset any optional re-entrant security levels and return to the initial state for memory access and to whatever operation mode it was in prior to the original Memory Access Request and may optionally consider this Operation Failed.

C.2.8.2 If the Memory Access Response from the Device to initiate the Close Sequence was Operation Failed then regardless of the response from the Tool, the Device shall reset any optional re-entrant security levels and return to the initial state for memory access and to whatever operation mode it was in prior to the original Memory Access Request and shall consider this Operation Failed.

C.2.8.3 If the response from the Tool (i.e. the Tool with the source address from which the initial memory access operation came) is another memory access request, and the manufacturer has allowed the optional re-entrant security and the initial request was other than Boot Load, the Device shall consider this Operation Completed and shall return to the internal state where it processes the memory access requests with re-entrant security.

C.2.8.4 If the response from the Tool is Operation Failed, the Device shall reset any optional re-entrant security levels and return to the initial state for memory access and to whatever operation mode it was in prior to the original Memory Access Request and shall consider this Operation Failed.

C.2.8.5 If the Memory Access Response from the Device to initiate the Close Sequence was Operation Completed and the Memory Access Request from the Tool to complete the Close Sequence was Operation Completed, then the Device resets any optional re-entrant security levels and returns to operational mode it was in prior to the Memory Access Request which initiated this sequence, unless the request was a Boot Load Command. When the request has been a Boot Load command the Device should transfer execution. Go to C.2.9.

C.2.8.6 If the response from the Tool is another memory access request, and the initial operation was Boot Load, the Device shall reset any optional re-entrant security levels and return to the initial state for memory access and to whatever operation mode it was in prior to the original Memory Access Request, thus effectively considering the Boot Load operation failed.

C.2.9 If a Boot Load Command is successfully completed (Both the Tool and the Device sent Operation Completed), then the Device transfers execution to the address that was determined from the Pointer, Pointer Extension, and Pointer Type of the initial Memory Access Request. At such time several results are possible, they are:

C.2.9.1 If there was no data to be sent (a zero Length/Number Requested in original request) the device will simply transfer execution to another location (may be used simply as a means to invoke a new mode of program operation within the device, such as ‘reset’, switch to an internal ‘loader’, etc.).

C.2.9.2 If there was new data transferred it may have been a new program which the device is simply to begin operating from. One possibility for this new program is that it is a reprogramming program designed to provide a more time efficient means of reprogramming the device’s executable memory. In such a case, the Tool and the Device may then intercommunicate by means of the Boot Load Data PGN. This Boot Load Data PGN can be transmitted from the Tool to the Device with the parameters in any format which meets the needs of the Device being programmed. The Boot Load Data PGN can be transmitted from the Device to the Tool as an Ack/Nak sequence or to control timing in any way deemed necessary to achieve the transfer between the Tool and the Device. It is expected that the device will no longer respond to any other PGNs transmitted to it; however, the Tool will be required to maintain communications to the remainder of the network devices. It will also become the Tool’s further responsibility to act on the Device’s behalf in any Network Management functions, in particularly to prevent an Address Claim by any other node of the address being used by the Device being programmed.

C.2.9.3 If there was new data transferred it may simply have been an addition to the existing program. This possibility would have required the manufacturer to have left space available for such an addition and have used a memory type that could have new data added without damage to the old. Should this have been the case execution would simply transfer to the new address just as when no new data had been added. Whether or not a reset or other operation would be necessary would be at the manufacturer's discretion.

C.2.10 If a Tool fails to hear the Memory Access Response message with operation completed or operation failed from a Device within what it considers to be appropriate time, it may send a Memory Access Request of Status Request to the Device. If the Tool receives no response within 0.25 seconds, it shall consider the Device is not going to respond and return to a mode of operation appropriate for this 'failure' (i.e. try to determine if the Device is still operating, if data has been damaged, etc.). If the Tool receives a Memory Access Response of Proceed from the Device, the Tool should recognize that the Device has already returned to the waiting for request state and considers the previous request completed and whether it was successful or failed can no longer be determined. The Tool may wish to attempt to determine why the device completed without it hearing the response (possible reasons are the response was not sent, bus communication is impaired, device had timed-out hearing the Tool at one of the interchange points, etc.).

(R) Appendix D Application Rules Regarding Data Security Message (DM18)

D.1 General Rules

The following general rules must be adhered to:

D.1.1 The message should only be sent to a specific destination, never to the global address or the unavailable address.

D.1.2 A Tool or a Device must have set the Key parameter in the Memory Access Request message or the Seed parameter in the Memory Access Response message (as appropriate) to identify that the Long Seed/Key is being used prior to transmission of the Data Security message by either. This enables the software in both to have a basis upon which to ‘flag’ that the Data Security message is going to be used to provide Long Seed/Key.

D.1.3 A Tool or a Device upon seeing either a Memory Access Request message or a Memory Access Response message identifying that the Long Seed/Key is being used shall set the appropriate ‘flags’ within the respective software such that they look for the Data Security message and process it.

D.2 Message exchange Rules

The following outlines the procedure for using the Data Security message to send/receive Long Seed/Key within a Memory Access sequence.

D.2.1 A Device that has received a Memory Access Request message (see Memory Access Request [section (R) 5.7.14] and Response [section (R) 5.7.15] documentation) for which it is going to send a Seed using the Data Security message should set the Seed parameter within the Memory Access Response message to indicate that the Seed will actually be sent using the Data_Security message. The Data Security message with the Long Seed should then be sent within 0.25 seconds of the Memory Access Response message. (A Tool may use a time-out of twice this value plus whatever bridge delays it has determined is appropriate for the particular system. This requires the Tool to have identified the system prior to this operation.)

D.2.2 A Tool that has received a Data_Security message containing a Long Seed from a Device should send the Long Key of that Long Seed back to the Device with the Data Security message within 0.25 seconds. (Note that bridges, when utilized, need to be accounted for in the time-out number. A Device may use a time-out of twice this value plus whatever bridge delays it has determined is appropriate for the particular system. This, of course, requires the Device to have identified the system prior to this operation.)

D.2.3 A Device that has received a Data Security message containing a Long Key (presumably of a Long Seed it had previously sent) from a Tool should verify the Seed and then continue with the Memory Access as outlined in Appendix C of the Memory Access Request and Response operation. (The two cases of Seed verified and failed are outlined separately. See Figure E.9 and Figure E.10)

(R) Appendix E Memory Access Information

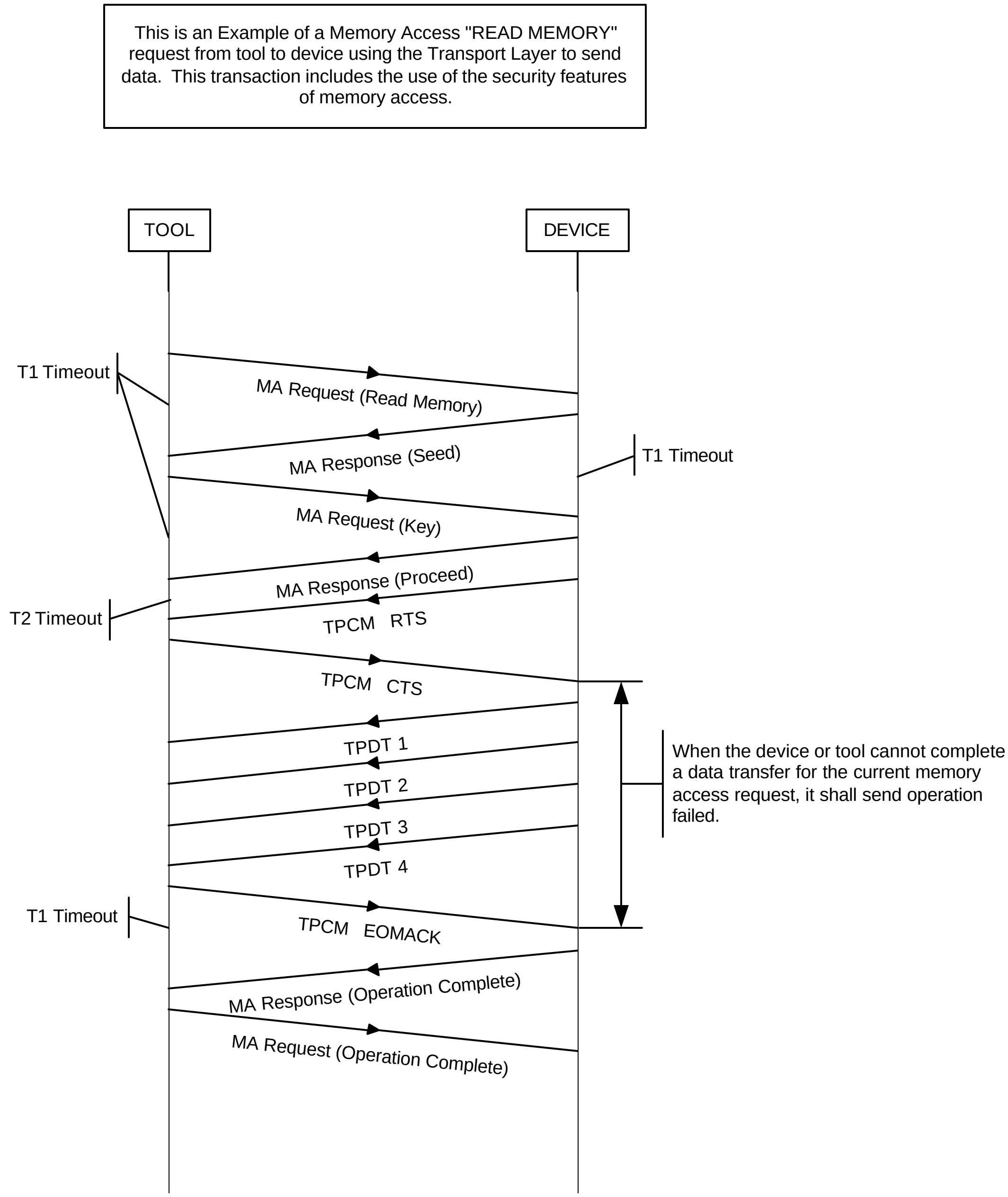


FIGURE E.1 EXAMPLE: MESSAGE SEQUENCE TO ACCOMPLISH MEMORY READ OPERATION WITH SECURITY (SHORT FORM OF SECURITY)

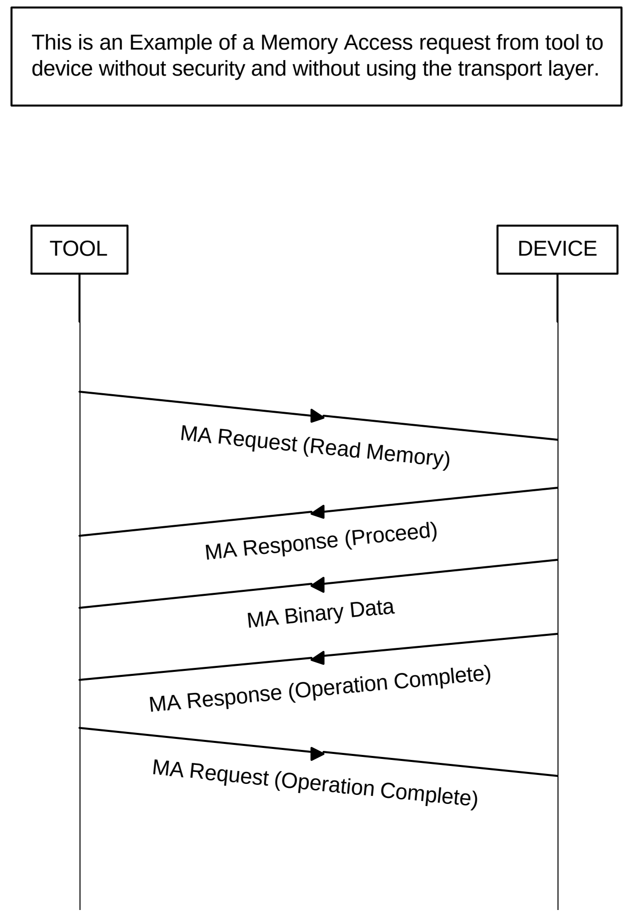


FIGURE E.2 EXAMPLE: MESSAGE SEQUENCE TO ACCOMPLISH MEMORY READ OPERATION WITHOUT SECURITY

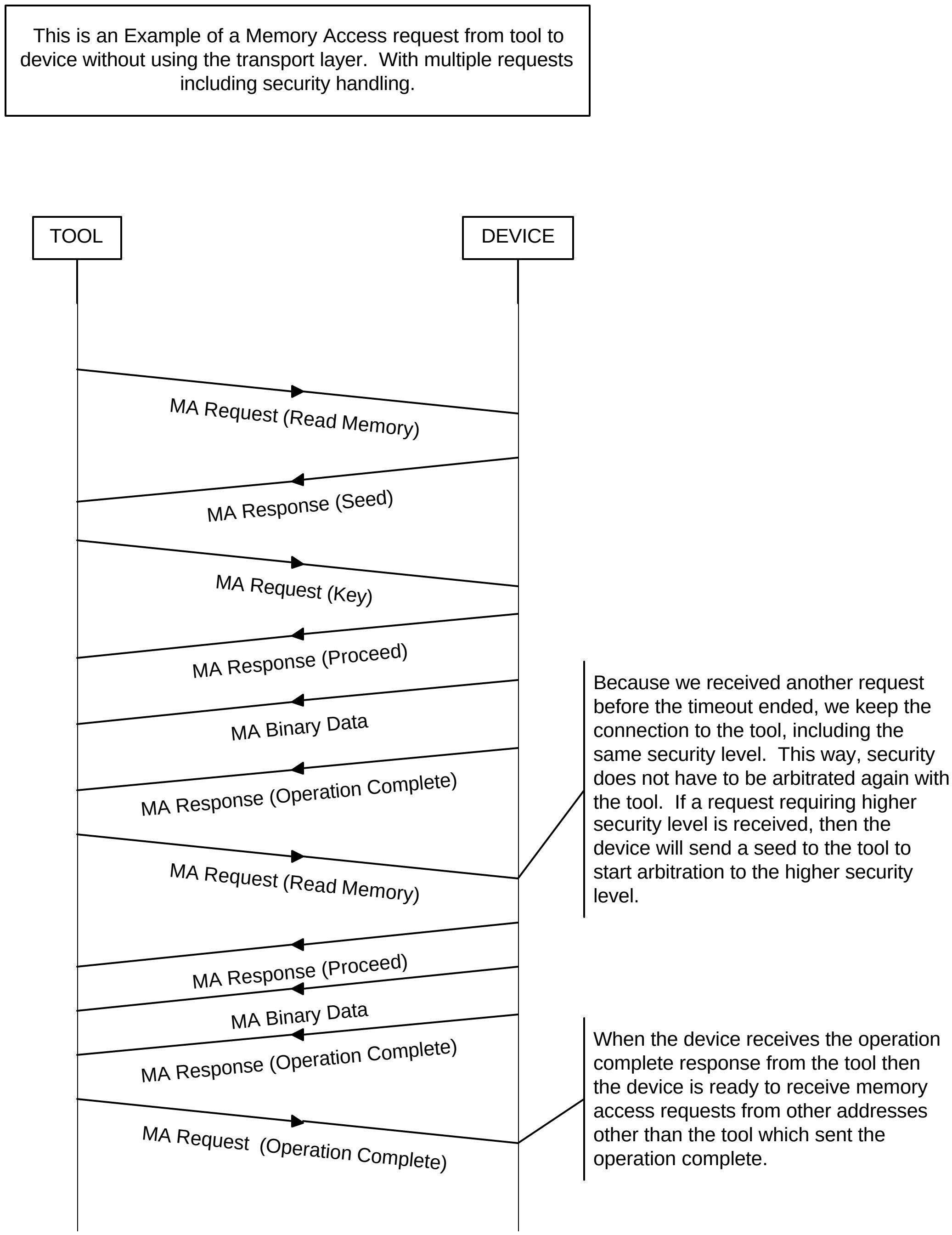


FIGURE E.3 EXAMPLE: MESSAGE SEQUENCE TO ACCOMPLISH MULTIPLE MEMORY READ OPERATION WITH SECURITY (SHORT FORM OF SECURITY)

This is an Example of a Memory Access request from tool to device when the tool does not send and operation complete.

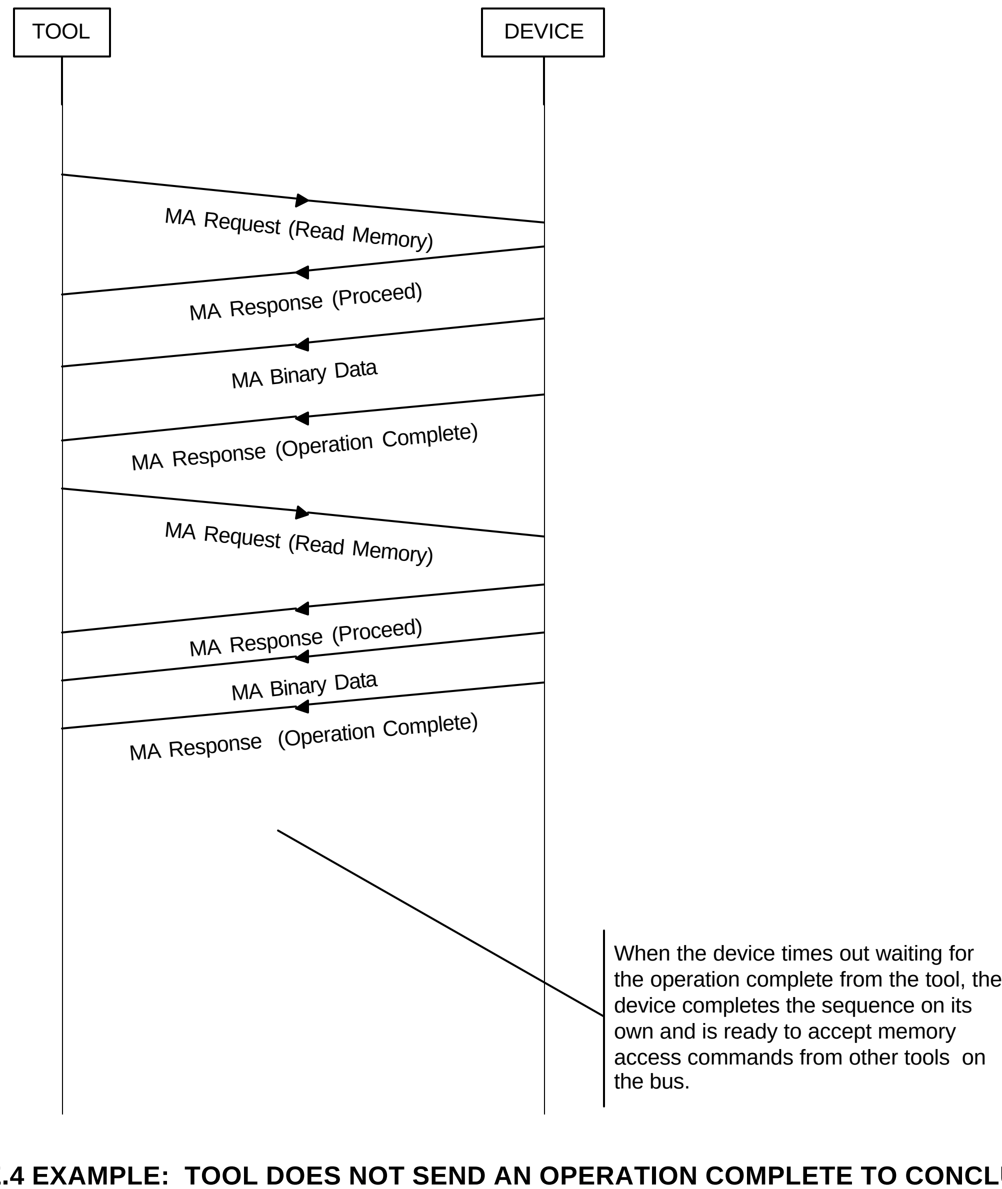


FIGURE E.4 EXAMPLE: TOOL DOES NOT SEND AN OPERATION COMPLETE TO CONCLUDE THE MEMORY ACCESS SESSION

This is an Example of a Memory Access "WRITE MEMORY" request from tool to device using the Transport Layer to send data. This transaction includes the use of the security features of memory access.

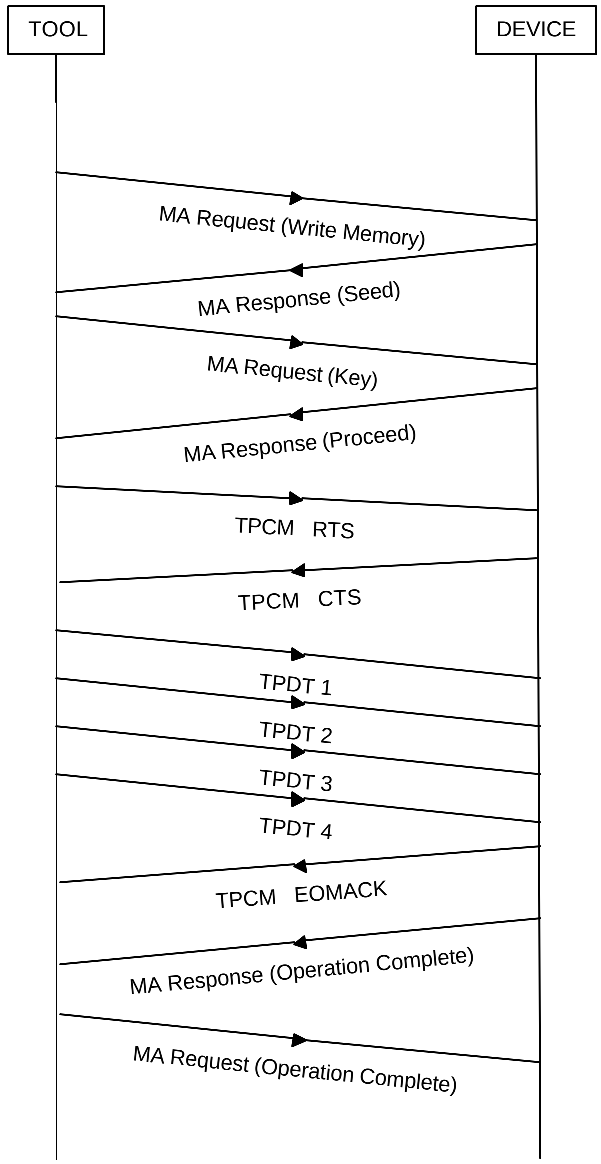


FIGURE E.5 EXAMPLE: WRITE MEMORY USING TRANSPORT PROTOCOL TO SEND THE DATA; ALSO USES THE SHORT FORM OF SECURITY

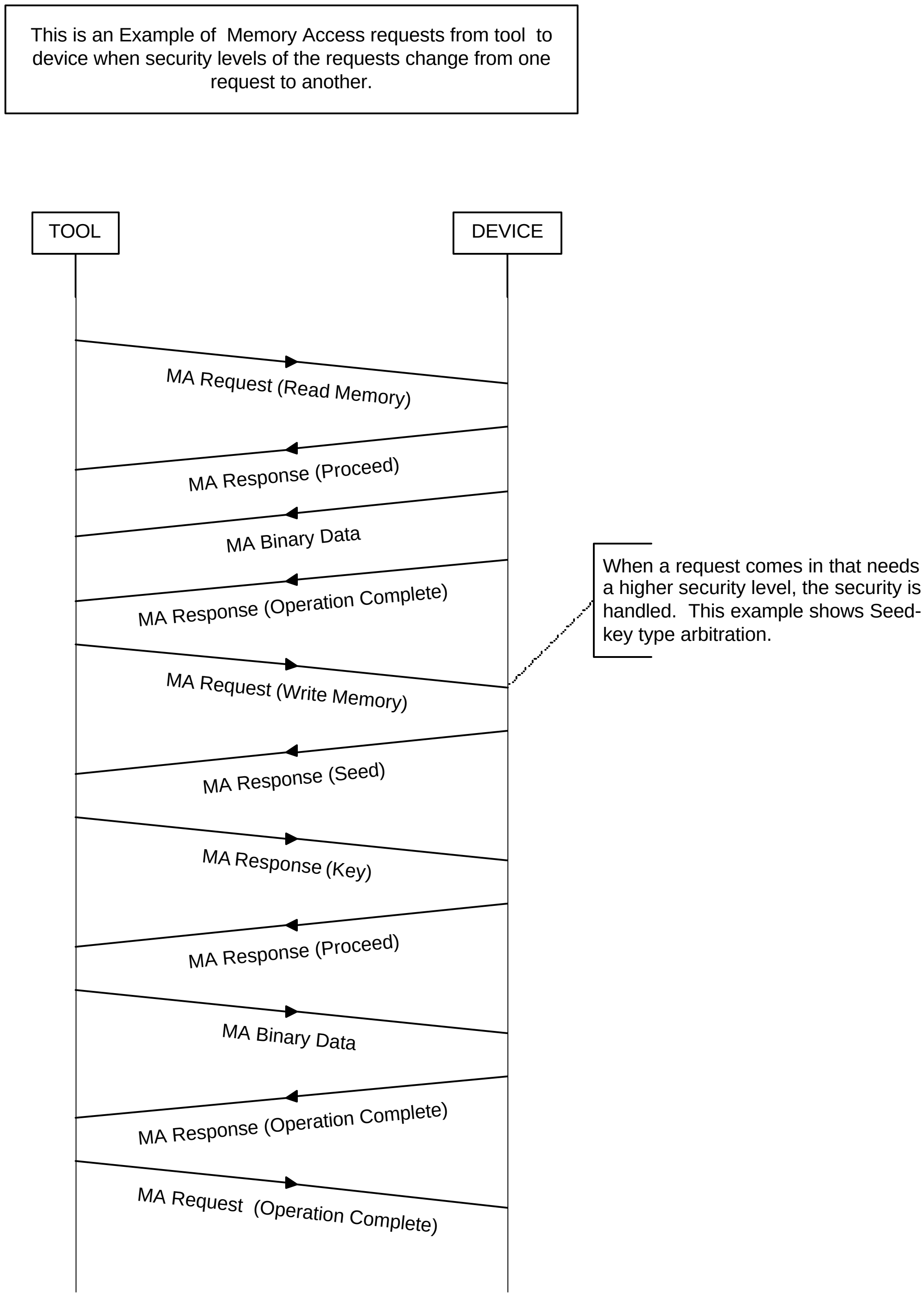


FIGURE E.6 EXAMPLE: MEMORY ACCESS TOOL TO DEVICE OPERATIONS REQUIRING DIFFERENT SECURITY LEVELS

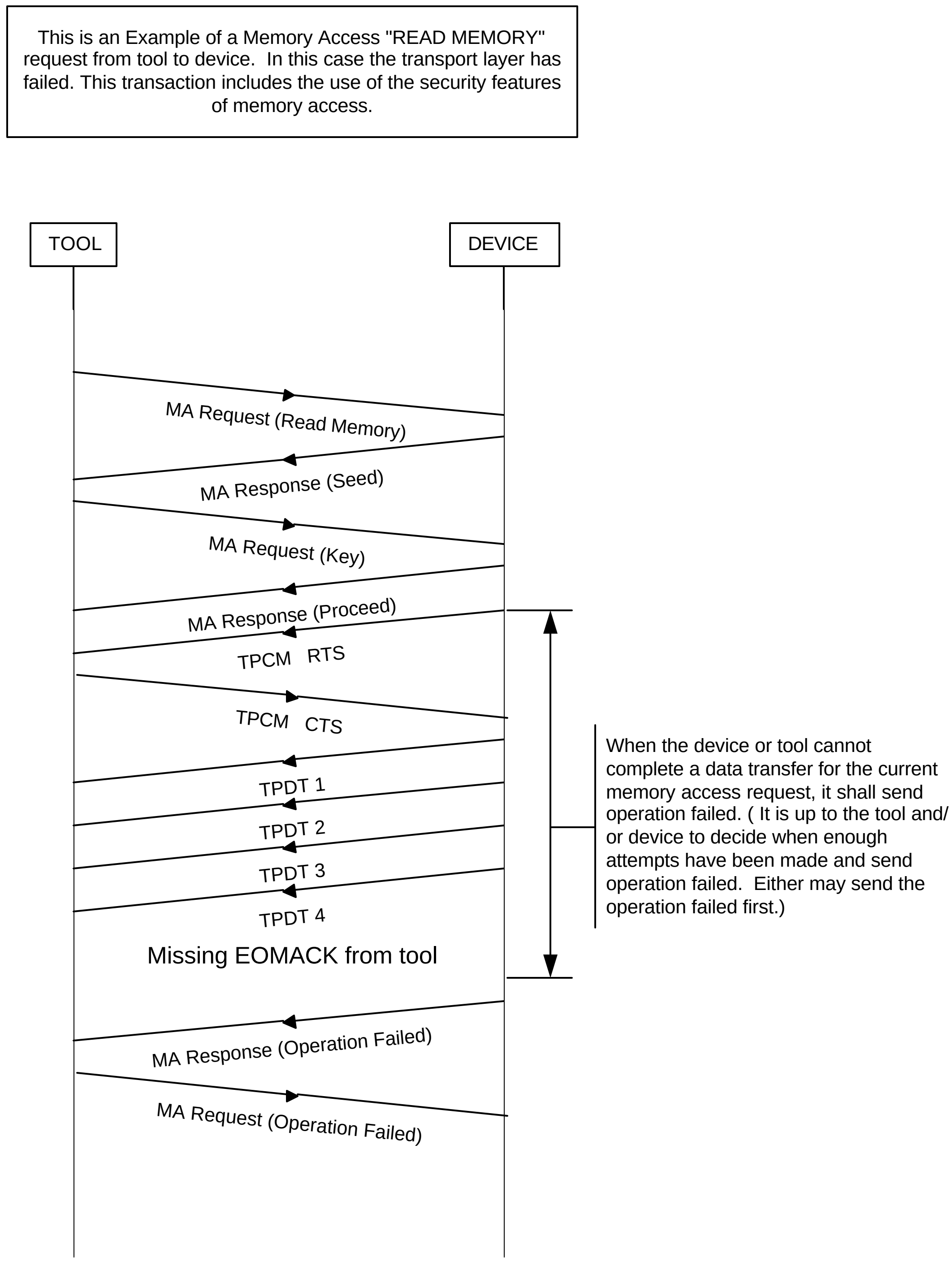


FIGURE E.7 EXAMPLE: MEMORY ACCESS OPERATION FAILED DUE TO TRANSPORT PROTOCOL SESSION FAILURE

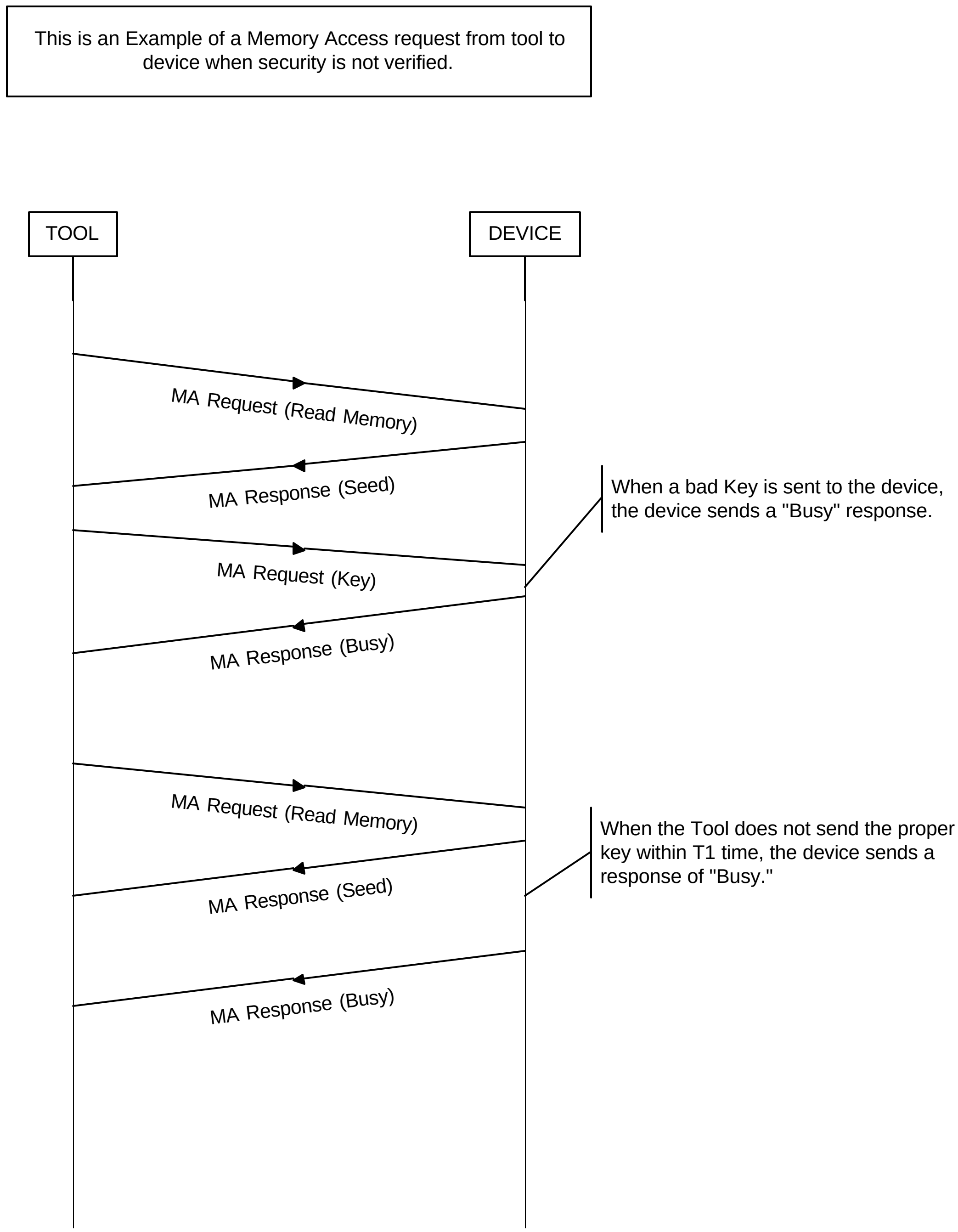


FIGURE E.8 EXAMPLE: MEMORY ACCESS OPERATION WHERE SECURITY IS NOT VERIFIED

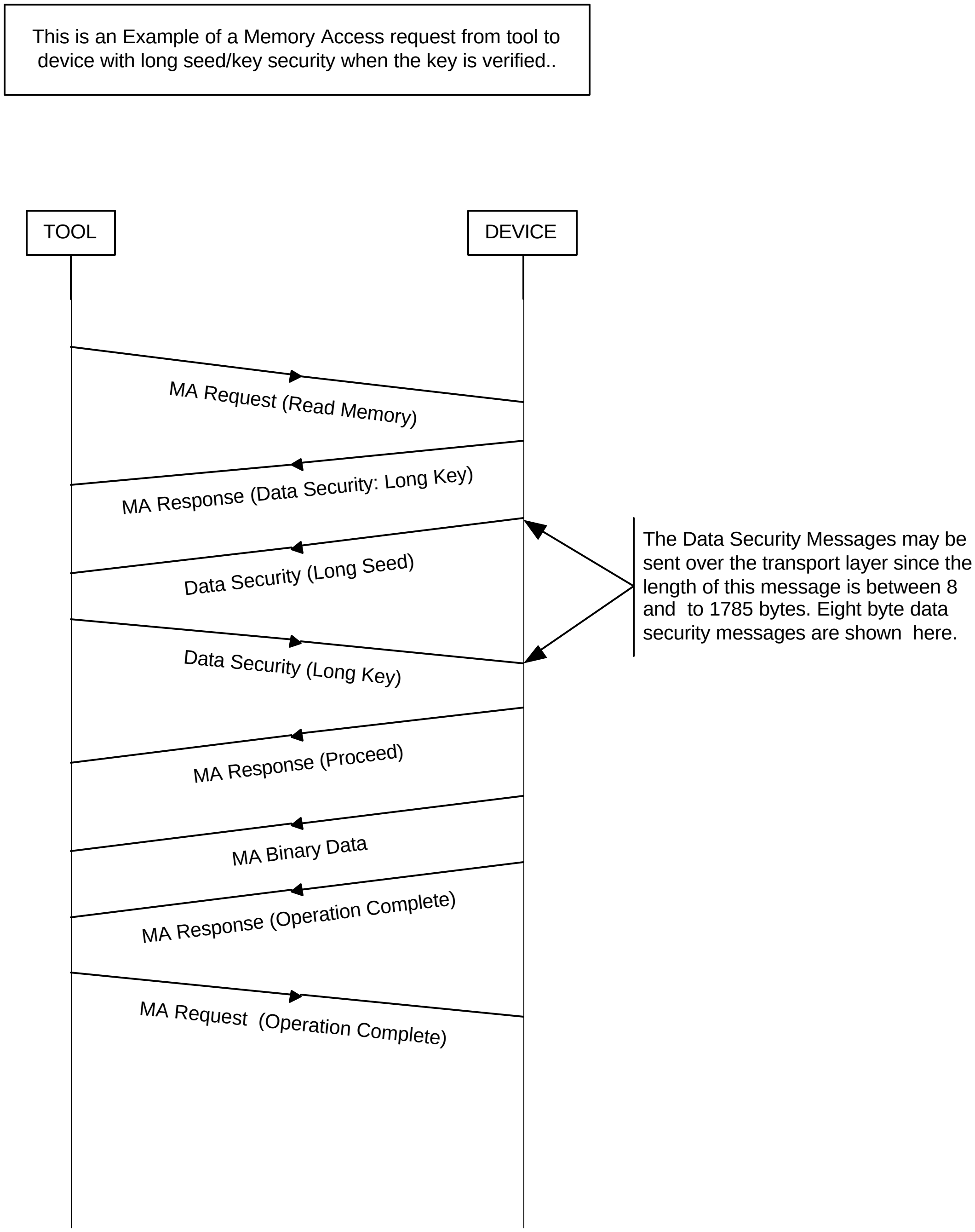


FIGURE E.9 EXAMPLE: MEMORY ACCESS OPERATION USING THE LONG SEED AND KEY

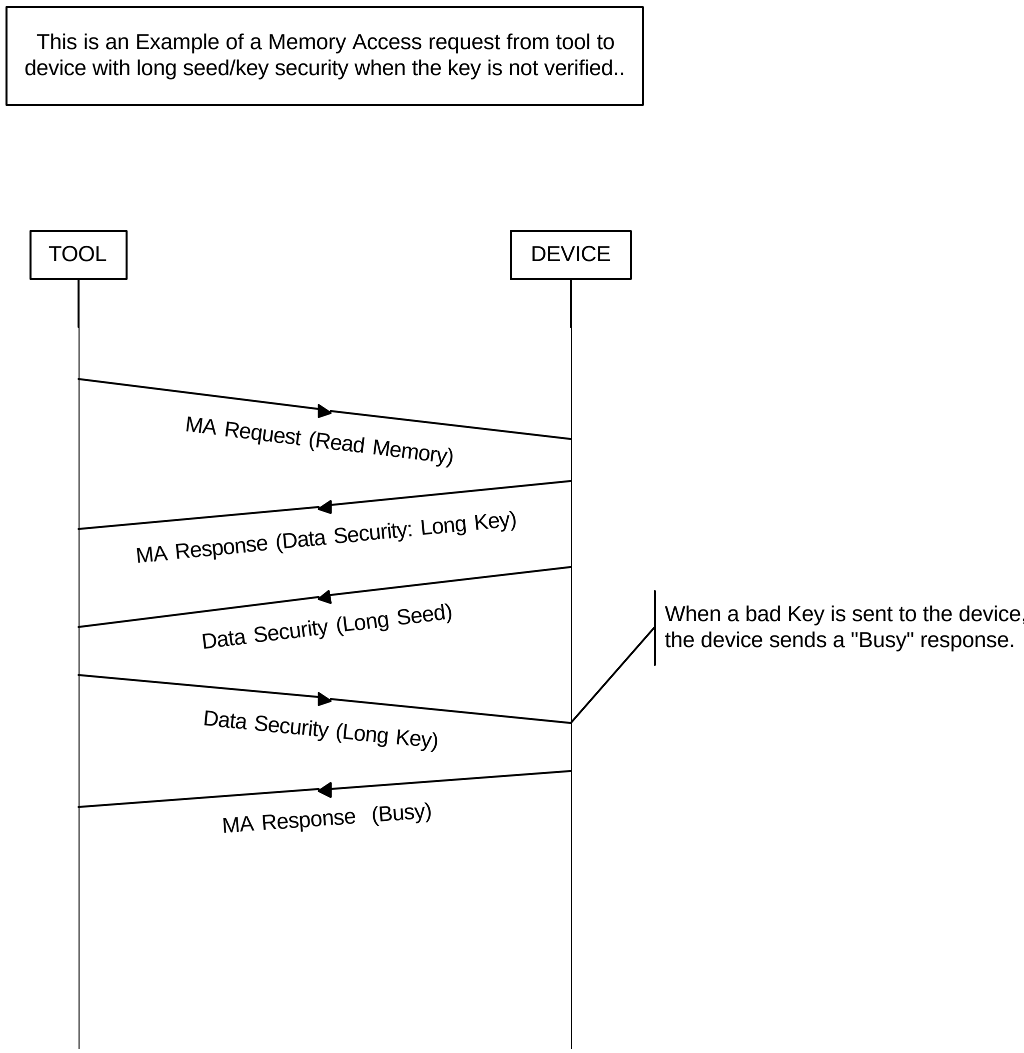


FIGURE E.10 EXAMPLE: TOOL DOES NOT SEND VALID KEY SO DEVICE RESPONDS WITH "BUSY"

Rationale—Not applicable.

Relationship of SAE Standard to ISO Standard—Not applicable.

Application—The SAE J1939 series of recommended practices are intended for light- and heavy-duty vehicle uses on- or-off road as well as appropriate stationary applications which use vehicle-derived components (e.g. generator sets). Vehicles of interest include, but are not limited to: on- and off-highway trucks and their trailers, construction equipment, and agriculture equipment and implements.

The purpose of these documents is to provide an open interconnect system for on-board electronic systems. It is the intention of these documents to allow electronic devices to communicate with each other by providing a standard architecture.

Reference Section

SAE J1587—Joint SAE/TMC Electronic Data Interchange Between Microcomputer Systems In Heavy-Duty Vehicle Applications

SAE J1939—Serial Control and Communications Vehicle Network

SAE J1939-13—Off-Board Diagnostic Connector

SAE J1939-21—Data Link Layer

SAE J1939-71—Vehicle Application Layer

SAE J1979—E/E Diagnostic Test Modes

Mail Out #95-03, January 19, 1995

OBD II, California code of regulations, Title 13, 1968.1: Malfunction and Diagnostics Systems Requirements, 1994 and subsequent model year passenger cars, light-duty trucks and medium-duty vehicles with feedback fuel control systems.

Developed by the SAE Truck and Bus Control and Communications Subcommittee

Sponsored by the SAE Truck and Bus Electrical and Electronics Committee