The Engineering Society For Advancing Mobility Land Sea Air and Space®
INTERNATIONAL
400 Commonwealth Drive, Warrendale, PA 15096-0001

SURFACE VEHICLE RECOMMENDED PRACTICE

SAE J2190

ISSUED JUN93

Issued 1993-06-25

Submitted for recognition as an American National Standard

ENHANCED E/E DIAGNOSTIC TEST MODES

TABLE OF CONTENTS

SAENORM.COM : Click to view the full PDF of j2190_199306

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

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

QUESTIONS REGARDING THIS DOCUMENT: (412) 772-8512 FAX: (412) 776-0243
TO PLACE A DOCUMENT ORDER; (412) 776-4970 FAX: (412) 776-0790
SAE WEB ADDRESS http://www.sae.org

Copyright 1997 Society of Automotive Engineers, Inc.
All rights reserved.

Printed in U.S.A.

SAENORM.COM : Click to view the full PDF of j2190_199306

1. ***Scope—***This SAE Recommended Practice describes the implementation of Enhanced Diagnostic Test Modes, which are intended to supplement the legislated Diagnostic Test Modes defined in SAE J1979. Modes are defined for access to emission related test data beyond what is included in SAE J1979, and for non-emission related data. This document describes the data byte values for diagnostic messages transmitted between diagnostic test equipment, either on-vehicle or off-vehicle, and vehicle electronic control modules. No distinction is made between test modes for emission related and non-emission related diagnostics. These messages can be used with a diagnostic serial data link such as described in SAE J1850 or ISO 9141-2.

For each test mode, this document includes a functional description of the test mode, request and report message data byte content, and an example if useful for clarification.

2. ***References***

2.1 **Applicable Publications—**The following publications form a part of this specification to the extent specified herein. The latest issue of SAE publications shall apply.

2.1.1 SAE PUBLICATIONS—Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001.

SAE J1850—Class B Data Communication Network Interface
SAE J1930—E/E Systems Diagnostic Terms, Definitions, Abbreviations and Acronyms
SAE J1962—Diagnostic Connector
SAE J1978—OBD II Scan Tool
SAE J2012—Recommended Format and Messages for Diagnostic Trouble Codes
SAE J2178—Class B Data Communication Network Messages
SAE J2186—E/E Data Link Security

SAENORM.COM : Click to view the full PDF of j2190_199306

2.1.2 ISO PUBLICATIONS—Available from ANSI, 1 West 42nd Street, New York, NY 10036-8002.

ISO 9141-2—Road vehicles—Diagnostic systems—CARB requirements for interchange of digital information

2.1.3 CALIFORNIA ARB DOCUMENTS

Mail out #93-27—Title 13, California Code of Regulations, Section 1968.1 Malfunction and Diagnostic System

May 21, 1993 Requirements—1994 and Subsequent Model-Year Passenger Cars, Light-Duty Trucks, and Medium-Duty Vehicles and Engines

2.1.4 FEDERAL EPA DOCUMENTS

40 CFR Part 86—Control of Air Pollution From New Motor Vehicles and New Motor Vehicle Engines; Federal Register Regulations Requiring On-board Diagnostics

3. Definitions

3.1 Data Bytes—Bytes between header bytes and error detection byte.

3.2 Diagnostic Test Mode—See SAE J1930.

3.3 Offset—A number used to refer to a data value by specifying its relative position in a list of data values.

3.4 Pid—Parameter Identification Number—A unique identifier used to refer to a specific data value within a module.

3.5 Seed / Key—See SAE J2186

3.6 Dpid—data Packet Identification Number—An identifier used to refer to a **set** of data values within a module.

3.7 Byte—A group of eight bits of data, bits 0 through bit 7, where bit 7 is the most significant bit and bit 0 is the least significant bit.

3.8 $—Prefix defining a hexadecimal number

3.9 Nibble—Four bits of data. A byte may be split into two nibbles.

3.10 EPA—Environmental Protection Agency (Federal Agency)

3.11 Carb—California Air Resources Board (State Agency)

3.12 Obd II—Second generation of On-Board Diagnostic regulations required by the California Air Resources Board.

SAENORM.COM : Click to view the full PDF of j2190_199306

4. *Technical Requirements*

4.1 Test Mode Values—Figure 1 indicates the assignment of test mode values, and indicates the type of message and where they are defined, either in SAE J1979, this document, or by the manufacturer.

Test mode values $00 - $0F and $40 - $4F are reserved to be defined in SAE J1979, which currently only includes functionally addressed messages. Usage of shaded test mode values will be defined in this document. Appendix A indicates the test mode values currently defined by both SAE J1979 and SAE J2190.

There is a one-to-one correspondence between request messages and response messages, with bit 6 of the test mode value indicating the message type.

Test Mode Value	Message type (bit 6)	Where defined
00 - 0F	Request (bit 6 = 0)	SAE J1979
10 - 1F		SAE J2190
20 - 2F		
30 - 3F		
40 - 4F	Response to Modes $00-$3F (bit 6 = 1)	SAE J1979
50 - 5F		SAE J2190
60 - 6F		
70 - 7F		
80 - 8F	Request (bit 6 = 0)	Reserved for future expansion as needed
90 - 9F		
A0 - AF		Defined by vehicle manufacturer
B0 - BF		
C0 - CF	Response to Modes $80-$BF (bit 6 = 1)	Reserved for future expansion as needed
D0 - DF		
E0 - EF		Defined by vehicle manufacturer
F0 - FF		

FIGURE 1—TEST MODE VALUES

4.2 Physical Addressing—Physical addressing is used for all diagnostic test mode messages defined in this document. Typically when using SAE J1850, this type of addressing requires that the target address of the device for which the message is intended be included in the header of the message. In ISO 9141, the addressing is typically done during initialization. Only that device being addressed will respond to the request. Each device will need to be assigned a unique address to be used for communication purposes.

SAENORM.COM : Click to view the full PDF of j2190_199306

4.3 Miscellaneous Requirements

4.3.1 MESSAGE RESPONSE TIME—The vehicle controllers should respond to a diagnostic request within 100 ms of a request. If there is no response within this time period, the tool can assume no response will be received.

4.3.2 AUTOMATIC RETURN TO NORMAL OPERATION—During a diagnostic procedure, the on-board controllers will often be put into an abnormal mode of operation to aid in diagnostics. Examples are to report diagnostic data periodically, to disable normal message transmission, or substitute an input or output parameter. In practice, the test procedure should return all on-board controllers to a normal mode of operation at the end of the procedure.

If a test device is disconnected from the vehicle before the on-board controllers have returned to normal operation, the controllers should automatically detect that the tool is disconnected and return to a normal mode of operation. This should be accomplished by the on-board controller looking for a diagnostic message from the test device, which should be apparent from the message header. If a diagnostic message is not received for a 5 s period of time, the on-board controller should automatically return to normal operation.

Some diagnostic procedures may require more than 5 s without a required diagnostic message. For these cases, this document specifies a "test tool present" message to be transmitted by the test device at least once every 5 s if there is no other message. On-board controllers should consider this as a diagnostic message and continue in the current mode of operation.

On-board controllers should also return to normal operation at power up or after a controller reset.

4.3.3 DIAGNOSTIC MESSAGE LENGTH—This document only defines the data bytes to be used in a diagnostic message. The actual number of data bytes in a message, referred to as "n" in the message data byte descriptions in this document, depends on the requirements of the test mode and the maximum number of bytes available in a diagnostic message. The maximum number of bytes available depends on the maximum allowable message length and the number of bytes used as a message header, error detection byte(s), or for other protocol specific purposes.

For example, the SAE J1850 data link effectively limits the total number of bytes in a message to 12. If the message strategy includes a 3 byte header, 1 byte CRC, and 1 byte in-message response, then 7 data bytes are available for SAE J2190 functionality. If the message strategy includes a 3 byte header and 1 byte CRC, then 8 data bytes are available.

If other data links are used as the diagnostic data link, such as ISO 9141, there may be other message length limitations and, therefore, different limits on the number of data bytes available for SAE J2190 functionality.

4.3.4 MESSAGE RESPONSE—All diagnostic request messages, except "Test Device Present (Mode $3F)," should receive a response from the target module. A response is optional for the "Test Device Present" message. There are two types of response messages, specific and general. Specific response messages are uniquely paired with each request message, and repeat enough data bytes of the request message to identity the specific request. The "General Response Message (Mode $7F)" has a single test mode value. The response message to be used for a request depends on the type of the request and whether or not the module can perform the request.

When the request is for data and the module can perform the request (positive response), only the specific form of the response can be used. The response will either be a single message, multiple messages with additional data, or periodic multiple messages with updated data values.

SAENORM.COM : Click to view the full PDF of j2190_199306

If the request is for action, such as a request for entering a mode of operation or command to perform a function, and the module can perform the request (positive response), either the specific response message or general response message can be used. Those requests for which either response type is a valid response have both response types indicated in this document.

If the request for either data or action cannot be performed by the module (negative response), only the general response message can be used.

Figure 2 summarizes the valid response types for different types of requests.

	Positive response	Negative response
Request for data	Specific response only	General response only
Request for action	Specific or general response allowed	General response only

FIGURE 2—VALID RESPONSE TYPES

With the General Response Message a response code is returned. Response codes can indicate a negative or positive response. Standard response codes are included in the section with the definition of the General Response Message.

5. Test Modes

5.1 Modes $00 through $0F—Physically Addressed SAE J1979 Messages—Modes $00 through $0F are reserved to be defined in SAE J1979. The data byte content of the response to an SAE J1979 request when the device is addressed physically should be identical to the data byte content of the response when the device is addressed functionally. The only device that will respond to this request is the device to which the request is directed.

The response to one of these requests will be a Mode $40 through $4F message.

The same PID list as specified in SAE J1979 will apply when the device is physically addressed using the same test mode value. Although SAE J1979 reserves 256 PID values to be defined in that document, SAE J1979 specifies that PID values $21 to $3F are reserved to be defined in this document. Possible values which may be considered to be included are:

Number of header bytes and format
Use of CRC versus checksum for error detection
Use of in-message response
Maximum number of data bytes allowed
(based on header and in-message response)
Use of manufacturer specific pins in the SAE J1962 connector

SAENORM.COM : Click to view the full PDF of j2190_199306

5.2 Mode $10—Initiate Diagnostic Operation

5.2.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to inform devices on the serial data link that a diagnostic tool is ready to start diagnostic procedures using the data link. The on-board devices may need to alter their normal operation in order for the diagnostic procedures to be effective. One example is that the system may need to reduce the amount of data being sent on the data link to make more time available for the diagnostic messages. Another possibility is that the system may need to degrade its normal operation in order to be able to process the diagnostic requests. Systems that rely on data from modules being diagnosed may not receive data from those modules as frequently as normal. Use of this mode could inform the receiving modules that diagnostic procedures are in progress and they should not set diagnostic trouble codes based on not receiving data at the normal frequency.

An optional "level of diagnostics" byte may be used to indicate how much the normal operation must be altered in order to accommodate the extra diagnostic procedures. These levels and their corresponding amount of change must be predefined in the module by the module manufacturer. The level of diagnostics can be changed by sending a subsequent Mode $10 command.

5.2.2 MESSAGE DATA BYTES—(See Figure 3.)

	Data Bytes (Hex)			
	#1	#2	#3 to #n-1	#n
Request from Tool to Vehicle				
Request diagnostic operation	10	Level (opt.)		
Positive Response from Vehicle to Tool (either of the following is a valid response)				
Confirm diagnostic operation	50	Level (opt.)		
General response	7F	Optional data bytes may be included (see Mode $7F description).		Resp. Code - 00

FIGURE 3—MESSAGE DATA BYTES FOR MODE $10

5.3 Mode $11—Request Module Reset

5.3.1 FUNCTIONAL DESCRIPTION—This mode requests the module to effectively perform a module power on reset. The response message may be sent either before or after the module is reset. The response message may also be sent by modules whenever they perform a power on reset, whether requested by this test mode or not.

SAENORM.COM : Click to view the full PDF of j2190_199306

An optional data byte may be included in the request to indicate different levels of reset. Examples are:

First time ever connected to vehicle
First time after battery disconnect
After a full normal power down / power up cycle
After minor power supply interruption

5.3.2 MESSAGE DATA BYTES—(See Figure 4.)

	Data Bytes (Hex)			
	#1	#2	#3 to #n-1	#n
Request from Tool to Vehicle				
Request module reset	11	Level (opt.)		
Positive Response from Vehicle to Tool (either of the following is a valid response)				
Confirm module reset	51	Level (opt.)		
General response	7F	Optional data bytes may be included (see Mode $7F description).		Resp. Code - 00

FIGURE 4—MESSAGE DATA BYTES FOR MODE $11

SAENORM.COM : Click to view the full PDF of j2190_199306

5.4 Mode $12—Request Diagnostic Freeze Frame Data

5.4.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to allow access to data values which were stored during freeze frame conditions specified by the vehicle manufacturer. Data content, data format, and method of retrieval are specified by the vehicle manufacturer. Typical uses for this mode are to report data stored upon detection of a system malfunction. Multiple frames of data may be stored before and/or after the malfunction is detected. The request for information includes a frame number followed by an optional indication of the data requested.

If the optional data byte is not used, or is $00, then all data for the requested freeze frame will be reported. If the optional byte is used, then the vehicle manufacturer will define the different methods used to specify which data is requested.

The on-board module will respond to this message by transmitting the requested data.

5.4.2 MESSAGE DATA BYTES—(See Figure 5.)

	Data Bytes (Hex)			
	#1	#2	#3	#4 to #n
Request from Tool to Vehicle				
Request diagnostic freeze frame data	12	Frame No.	Method to request data (optional) examples are: Request all data (not included or 00) Request by offset (1 byte) Request by PID (2 bytes) Request by memory address (3 bytes) Request by Data Packet ID Request starting address for data DTC which caused data to be stored Other manufacturer specific method	
Positive Response from Vehicle to Tool				
Report diagnostic freeze frame data	52	Frame No.	Data byte	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.

FIGURE 5—MESSAGE DATA BYTES FOR MODE $12

SAENORM.COM : Click to view the full PDF of j2190_199306

5.5 Mode $13—Request Diagnostic Trouble Code Information

5.5.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to enable a diagnostic test tool to obtain stored emission and non-emission related diagnostic trouble code (DTC) information.

This mode includes the option to request DTC information by function, where function is either powertrain body, chassis, or undefined. These are the functions used in SAE J2012 to group trouble codes. The first two bits of the first nibble for requests by function are encoded using the same convention as when codes are reported:

00 - powertrain
01 - chassis
10 - body
11 - undefined

For requests by function using this test mode, the second two bits of the first nibble are 00, and the second nibble is all zeroes. This translates to the following function groups for DTC requests:

$00 - powertrain codes
$40 - chassis codes
$80 - body codes
$C0 - undefined

In addition, this document defines $FF as the function group to request DTC information for all functions.

This test mode can be used to either request diagnostic trouble codes, or request the number of diagnostic trouble codes. There are two ways this mode can request DTCs.

All codes can be requested by including only the test mode value in the request.

Codes can be requested by function group (Powertrain, Chassis, Body, Undefined, or All) by sending the function group followed by $00 as data bytes 2 and 3 ($00 00, $40 00, $80 00, $C0 00, or $FF 00). Requesting codes for all function groups by including $FF 00 yields the same response as not including function group in the request, but may be desired by a manufacturer for consistency with other messages.

There are also two ways this mode can request the number of DTCs.

Number of codes can be requested by including the function group as data byte 2, and not including data byte 3 ($FF as data byte 2 must be supported as a minimum to request the total number of DTC).

Number of codes can be requested by function group (Powertrain, Chassis, Body, Undefined, or All) by sending the function group followed by $FF as data bytes 2 and 3 ($00 FF, $40 FF, $80 FF, $C0 FF, or $FF FF). Requesting number of codes for all function groups by including $FF FF yields the same response as including only $FF as data byte 2 in the request, but may be desired by a manufacturer for consistency with other messages.

The response to a Mode $13 request for DTC will be one or more Mode $53 messages. If no codes are stored in the module, then the module will respond with one of the following:

Mode $53 with no additional data bytes

Mode $53 padded with $00 00 to fill response

Diagnostic trouble codes are transmitted as two bytes. The first two bits (high order) of the first byte for each code will be zeroes to indicate a powertrain code (refer to SAE J2012 - "Recommended Format and Messages for Diagnostic Trouble Codes" for additional interpretation of this structure). The second two bits will indicate the first digit of the diagnostic code (0 through 3). The second nibble of the first byte and the entire second byte are the next three digits of the actual code reported as Binary Coded Decimal (BCD). A powertrain trouble code transmitted as $0143 should be interpreted as P0143, as shown in Figure 6.

SAENORM.COM : Click to view the full PDF of j2190_199306

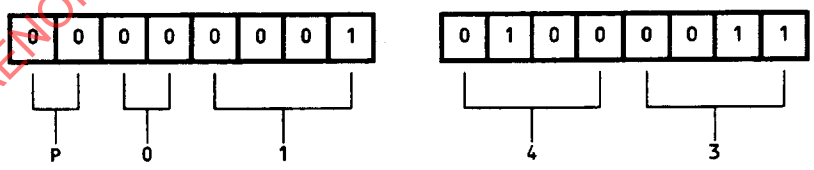

FIGURE 6—DIAGNOSTIC TROUBLE CODE EXAMPLE

5.5.2 MESSAGE DATA BYTES—(See Figure 7.)

	Data Bytes (Hex)			
	#1	#2	#3	#4 to #n
Request from Tool to Vehicle (either of the following messages can be used to request DTC)				
Request all DTC	13			
Request DTC by function group where group is: 00 - powertrain 40 - chassis 80 - body C0 - undefined FF - all	13	Group	00	
Request from Tool to Vehicle (either of the following messages can be used to request number of DTC)				
Request number of DTC by function where group is: 00 - powertrain 40 - chassis 80 - body C0 - undefined FF - all	13	Group		
FF in byte 3 is an optional byte used to maintain a fixed length request	13	Group	FF	
Positive Response from Vehicle to Tool (Multiple messages may be required to report the DTCs)				
Report stored DTC - response to request for DTC	53	Diagnostic trouble code or 00 00 (optional)		Additional optional diagnostic trouble codes may be added to fill the message up to the maximum number of available data bytes.
Report # of stored DTC - response to request for number of DTC	53	Number of codes		

NOTE—Refer to SAE J2012 for encoding method for trouble codes.

FIGURE 7—MESSAGE DATA BYTES FOR MODE $13

SAENORM.COM : Click to view the full PDF of j2190_199306

5.6 Mode $14—Clear Diagnostic Information

5.6.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to provide a means for the external test device to command on-board modules to clear all diagnostic information. This information includes primarily diagnostic trouble codes, but can also include freeze frame data or other on-board test results that may be stored as a result of the trouble code being set. This extra information is device dependent.

There are three ways this mode can be used. If only the test mode value is included in the request, then all diagnostic information stored in the module is to be cleared.

Diagnostic information can optionally be cleared by function (Powertrain, Chassis, or Body) by sending P0000, C0000, or B0000 as data bytes 2 and 3. These must be encoded using the same convention as when codes are reported (see description for Mode $13). The first two bits of the first nibble are:

00 - powertrain
01 - chassis
10 - body
11 - undefined

This translates to the following values for data bytes #2 and #3:

$00 00 - powertrain codes
$40 00 - chassis codes
$80 00 - body codes
$C0 00 - undefined

Diagnostic information can also optionally be cleared for a single trouble code by including the trouble code to clear as optional bytes in the request. The information stored with an individual trouble code would need to be clearly identified within the module and cleared at the same time.

Caution when clearing by single trouble code: This capability should not be provided and used unless consideration is given to the possible consequences of this option. If a single fault caused multiple trouble codes to be set, and a service technician found a defective component, repaired it, and cleared a single trouble code associated with that fault, then the device would still contain trouble codes that were set based on that fault. The next technician to check codes would read those codes, and if he did not know the vehicle history, he may try to repair the vehicle based only on the remaining trouble codes, which could waste time and result in replacement of good components.

Allowing use of this feature to clear emission related trouble codes should be considered very carefully to guard against improper diagnosis. Difficult interpretation can result if the system allowed clearing a single emission related trouble code. CARB required information could report that trouble codes are stored, but they would not be reported when an SAE J1979 Mode $04 message is sent to request those codes. Freeze frame data could be stored for a given trouble code, but that code would not be reported as stored.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.6.2 MESSAGE DATA BYTES—(See Figure 8.)

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Clear stored diagnostic information	14	Data bytes 2 and 3 are optional if not used - clear all information if 00 00 - clear all powertrain information if 40 00 - clear all chassis information if 80 00 - clear all body information if C0 00 - clear all Cx xx DTC information if FF 00 - clear all information any other value - trouble code to clear			
Positive Response from Vehicle to Tool (either of the following is a valid response)					
Diagnostic information cleared	54	repeat data bytes 2 and 3 of request, if used			
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. Code - 00

FIGURE 8—MESSAGE DATA BYTES FOR MODE $14

SAENORM.COM : Click to view the full PDF of j2190_199306

5.7 Mode $17—Request Status of Diagnostic Trouble Codes

5.7.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to provide a means to determine the status of diagnostic trouble codes.

This mode is nearly identical in operation to Mode $13, except that only a single DTC, followed by the status of that DTC, is reported in a single response. This mode also allows a request for the status of a single DTC. The discussion of function groups and the method of reporting DTCs is identical to the description in Mode $13.

This test mode can be used to either request status of diagnostic trouble codes, or request the number of diagnostic trouble codes with a status bit set. There are three ways this mode can request DTCs.

All codes can be requested by only including the test mode value in the request.
Codes can be requested by function group (Powertrain, Chassis, Body, Undefined, or All) by sending the function group followed by $00 as data bytes 2 and 3 ($00 00, $40 00, $80 00, $C0 00, or $FF 00). Requesting codes for all function groups by including $FF 00 yields the same response as not including function group in the request, but may be desired by a manufacturer for consistency with other messages.
Status of a single trouble code can be requested by including the trouble code as data bytes 2 and 3 in the request.

There are two ways this mode can request the number of DTCs with a status bit set.

Number of codes can be requested by including the function group as data byte 2, and not including data byte 3 ($FF as data byte 2 must be supported as a minimum to request the total number of DTC).
Number of codes can be requested by function group (Powertrain, Chassis, Body, Undefined, or All) by sending the function group fy $FF as data bytes 2 and 3 ($00 FF, $40 FF, $80 FF, $Co FF, or $FF FF). Requesting number of codes for all function groups by including $FF FF yields the same response as including only $FF as data byte 2 in the request, but may be desired by a manufacturer for consistency with other messages.

Multiple Mode $57 response messages may be reported due to a single Mode $17 request message, depending on the type of request and the number of diagnostic trouble codes stored in the module. Each response message will report the status of a single DTC. If no codes are stored in the module, then the module will respond with one of the following:

Mode $57 with no additional data bytes
Mode $57 with $00 00 for DTC and $00 for status

SAENORM.COM : Click to view the full PDF of j2190_199306

5.7.2 MESSAGE DATA BYTES—(See Figure 9.)

	Data Bytes (Hex)			
	#1	#2	#3	#4
Request from Tool to Vehicle (any of the following messages can be used to request DTC)				
Request status of all DTC	17			
Request status of DTC by function group where group: 00 - powertrain 40 - chassis 80 - body C0 - undefined FF - all	17	Group	00	
Request status of specific DTC	17	Diagnostic Trouble Code		
Request from Tool to Vehicle (either of the following messages can be used to request number of DTC)				
Request number of DTC with a status bit set by function where group is: 00 - powertrain 40 - chassis 80 - body C0 - undefined FF - all	17	Group		
FF in byte 3 is an optional byte used to maintain a fixed length request	17	Group	FF	
Positive Response from Vehicle to Tool (Multiple messages may be required to report the DTCs)				
Report status of stored DTC	57	Diagnostic trouble code or 00 00 (optional)		Status of DTC or 00 (optional if DTC not reported): same definition as Data Byte #2 for Mode $18
Report # of stored DTC with any status bit set	57	Number of codes		

FIGURE 9—MESSAGE DATA BYTES FOR MODE $17

SAENORM.COM : Click to view the full PDF of j2190_199306

5.8 Mode $18—Request Diagnostic Trouble Codes by Status

5.8.1 FUNCTIONAL DESCRIPTION—This mode is used to retrieve diagnostic trouble codes based on the status by which they have been stored. There are various conditions under which codes are generated, for example, under normal customer driving conditions versus "on demand" codes which are generated by tests performed by the service technician. Another difference is that some codes illuminate a warning lamp while other codes do not. Some codes are retained in memory for at least 40 warmup cycles while some codes are retained only for one ignition/power up cycle.

This test mode is similar in operation to Mode $17, except that an additional data byte is inserted as data byte #2 to indicate the specific status bits of interest. Data bytes #3 and #4 in Mode $18 requests have the same meaning as data bytes #2 and #3 in Mode $17 requests.

There are many philosophies among the various manufacturers for storing codes. Bit definitions have been assigned to the more commonly accepted methods. Codes stored under that definition can be requested by setting the appropriate bit(s) to "1" to retrieve the set of desired codes.

Table 1 shows the interpretation of each bit of the status byte. This definition is used for both Modes $17 and $18. Multiple bits can be set to "1". When requesting codes using Mode $18, codes with any of the requested status bits will be reported.

SAENORM.COM : Click to view the full PDF of j2190_199306

TABLE 1—INTERPRETATION OF EACH BIT OF THE STATUS BYTE

bit	status
7	Warning lamp illuminated for this code
6	Warning lamp pending for this code, not illuminate but malfunction was detected
5	Warning lamp was previously illuminated for this code, malfunction not currently detected, code not yet erased
4	Stored trouble code
3	Manufacturer specific status
2	Manufacturer specific status
1	Current code - present at time of request
0	Maturing/intermittent code - insufficient data to consider as a malfunction

Multiple Mode $58 response messages may be reported to a single request, depending on the number of diagnostic trouble codes stored in the module. Each response message will report up to three DTCs for which at least one of the requested status bits is set. If no codes are stored in the module that meet the requested status, then the module will respond with one of the following:

Mode $58 with Status Requested and no additional data bytes
Mode $58 with $00 00 as data bytes 3 and 4
Mode $58 with $00 00 repeated to fill response

5.8.2 MESSAGE DATA BYTES—(See Figure 10.)

SAENORM.COM : Click to view the full PDF of j2190_199306

	Data Bytes (Hex)				
	#1	#2	#3	#4	#5 to #n
Request from Tool to Vehicle (either of the following messages can be used to request DTC)					
Request all DTC with status	18	Status			
Request all DTC with status and function group where group is: 00 - powertrain 40 - chassis 80 - body C0 - undefined FF - all	18	Status	Group	00	
Request from Tool to Vehicle (either of the following messages can be used to request number of DTC)					
Request number of DTC with any specific status bits set by function group 00 - powertrain 40 - chassis 80 - body C0 - undefined FF - all	18	Status	Group		
FF in byte 4 is an optional byte used to maintain a fixed length request	18	Status	Group	FF	
Positive Response from Vehicle to Tool					
Report DTC by status - response to request with test mode and status	58	Status req'd	Diagnostic trouble code or 00 00 (optional)	Additional optional diagnostic trouble codes may be added to fill the message up to the maximum number of available data bytes.	
Report DTC by status - response to request with test mode, status, and data byte #3	58	Status req'd	number of DTC with req'd status		

FIGURE 10—MESSAGE DATA BYTES FOR MODE $18

SAENORM.COM : Click to view the full PDF of j2190_199306

5.9 Mode $20—Return to Normal Operation

5.9.1 FUNCTIONAL DESCRIPTION—The on-board device will return to the normal mode of operation when this message is received. All normal algorithms and normal communications will be resumed. All active diagnostic modes will be terminated without sending additional request or response messages. If the module had been unlocked using security access, then the module should also be locked.

5.9.2 MESSAGE DATA BYTES—(See Figure 11.)

	Data Bytes (Hex)		
	#1	#2 to #n-1	#n
Request from Tool to Vehicle			
Request return to normal operation	20		
Positive Response from Vehicle to Tool (either of the following is a valid response)			
Confirm return to normal operation	60		
General response	7F	Optional data bytes may be included (see Mode $7F description).	Resp. Code - 00

FIGURE 11—MESSAGE DATA BYTES FOR MODE $20

SAENORM.COM : Click to view the full PDF of j2190_199306

5.10 Modes $21 to $23—Request Diagnostic Data

5.10.1 FUNCTIONAL DESCRIPTION—The purpose of these modes is to request data values, such as analog inputs and outputs, digital inputs and outputs, freeze frame data, calculated values, bit mapped fault code data, and system status information. The request for information can be by one of three methods:

Mode $21 - Offset (1 byte)
Mode $22 - Parameter Identification (PID) value (2 bytes)
Mode $23 – Memory Address (3 bytes)

System designers have the flexibility to maintain a table including PID numbers and data values, without having consecutive PID numbers. In this case, the tool can request either by PID number or offset. A request by offset is generally an easier method for the software to retrieve data because it does not need to search for the PID in the table. Mode $24 allows use of the PID number one time to get the offset, with successive requests by offset for better system performance.

Figure 12 is an example of a table that could be included in the system module to utilize all of these modes. The offset is the row number of the table and is not a table entry. The location of the information within this table can be calculated by multiplying the offset times the number of bytes in each row. The location in memory can then be calculated by adding the location within the table to the base location of the table in memory.

Offset	PID1	PID2	ADDR1	ADDR2	ADDR3	SCALING1	SCALING2
00	00	01	00	1B	FE		
01	00	0A	00	2C	12		
02	01	FE	00	BA	1F		
--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--
xx	04	2C	00	65	8A		

FIGURE 12—OFFSET/PID/ADDRESS TABLE EXAMPLE

A Mode $21 request by offset would calculate the location of the offset information in the table, find the address and scaling for that value, and return the appropriate number of bytes of information starting at that address.

A Mode $22 request by PID would search through the table for the requested PID, find the address and scaling for that value, and return the appropriate number of bytes of information starting at that address.

A Mode $23 request by address would ignore this table and report data starting at the address specified in the request.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.10.2 MODE $21 MESSAGE DATA BYTES—(See Figure 13.)

	Data Bytes (Hex)						
	#1	#2	#3	#4	#5	#6 to #n-1	#n
Request from Tool to Vehicle							
Request data by offset	21	offset	Value for Byte 3 (optional): not included - 1 response 00 - stop sending data 01 - send 1 response 02 - repeat at slow rate 03 - repeat at medium rate 04 - repeat at fast rate 05 to FF - manufacturer defined conditional response, where the condition is pre-defined in the module - multiple conditions may be defined Value for Byte 4 (optional if Byte 3 supported): 00 - send data until requested to stop nn - maximum number of responses to send					
Vehicle Response to Request for Data (Multiple response messages will be sent if data rate requested periodic data reporting)							
Report data by offset	61	offset	data byte	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.			
Vehicle Response to Request to stop sending data (either of the following is a valid response)							
Report stop sending data by offset	61	offset					
General response	7F	Optional data bytes may be included (see Mode $7F description).					Resp. Code - 00

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 13—MESSAGE DATA BYTES FOR MODE $21

5.10.3 MODE $22 MESSAGE DATA BYTES—(See Figure 14.)

	Data Bytes (Hex)						
	#1	#2	#3	#4	#5	#6 to #n-1	#n
Request from Tool to Vehicle							
Request data by PID	22	PID (high byte)	PID (low byte)	Value for Byte 4 (optional): same definition as Data Byte #3 for Mode $21 Value for Byte 5 (optional if Byte 4 supported): 00 - send data until requested to stop nn - max. number of responses to send			
Vehicle Response to Request for Data (Multiple response messages will be sent if data rate requested periodic data reporting)							
Report data by PID	62	PID (high byte)	PID (low byte)	data byte	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.		
Vehicle Response to Request to stop sending data (either of the following is a valid response)							
Report stop sending data by PID	62	PID (high byte)	PID (low byte)				
General response	7F	Optional data bytes may be included (see Mode $7F description).					Resp. Code - 00

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 14—MESSAGE DATA BYTES FOR MODE $22

5.10.4 MODE $23 MESSAGE DATA BYTES—(See Figure 15.)

	Data Bytes (Hex)						
	#1	#2	#3	#4	#5	#6 to #n-1	#n
Request from Tool to Vehicle							
Request data by memory address	23	addr (high byte)	addr (mid byte)	addr (low byte)	Value for Byte 5 (optional): same definition as Data Byte #3 for Mode $21 Value for Byte 6 (optional if Byte 5 supported): 00 - send data until requested to stop nn - max. number of responses to send		
Vehicle Response to Request for Data (Multiple response messages will be sent if data rate requested periodic data reporting)							
Report data by memory address	63	addr (mid byte)	addr (low byte)	data byte	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.		
Vehicle Response to Request to stop sending data (either of the following is a valid response)							
Report stop sending data by memory address	63	addr (mid byte)	addr (low byte)				
General response	7F	Optional data bytes may be included (see Mode $7F description).					Resp. Code - 00

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 15—MESSAGE DATA BYTES FOR MODE $23

5.11 Mode $24—Request Scaling and Offset / PID

5.11.1 FUNCTIONAL DESCRIPTION—The purpose of Mode $24 is to request either scaling and PID information by specifying offset, or request scaling and offset information by specifying PID. $FF should be included in the request for the data byte value not being specified. If both offset and Parameter ID are specified, and the parameter ID is not correct for the offset specified, unexpected data values can be returned. Operation of this mode is similar to either Mode $21 or $22, depending on which value was specified in the request. The response would include offset, PID, and scaling for the requested entry. Any data byte values not supported by the module would report $FF.

Mode $24 can also be used to determine the PID table by incrementing the offset in repeated Mode $24 requests.

5.11.2 MESSAGE DATA BYTES—(See Figure 16.)

	Data Bytes (Hex)				
	#1	#2	#3	#4	#5 to #n
Request from Tool to Vehicle					
Request scaling and offset	24	offset ($FF if N/A)	PID (high byte - $FF if N/A)	PID (low byte - $FF if N/A)	
Positive Response from Vehicle to Tool					
Report scaling and offset	64	offset	PID (high byte - $FF if N/A)	PID (low byte - $FF if N/A)	Scaling bytes may be added to fill the message up to the maximum number of available data bytes.

FIGURE 16—MESSAGE DATA BYTES FOR MODE $24

SAENORM.COM : Click to view the full PDF of j2190_199306

5.11.3 SCALING BYTE 1—High nibble (bits 7 to 4) data type

0000 - Unsigned numeric (1 to 4 bytes)
0001 - Signed numeric (1 to 4 bytes)
0010 - Bit mapped, reported without mask
0011 - Bit mapped, reported with mask
0100 - BCD (1 to 4 bytes)
0101 - State encoded
0110 - ASCII
0111 - Signed floating point (ANSI/IEEE Std 754-1985)
1000 - Packet
1001 - Formula
1010 thru 1111 - Reserved

Low nibble (bits 3 to 0) number of bytes of value

0000 - Reserved
0001 - 1 byte
0010 - 2 bytes
0011 - 3 bytes
0100 - 4 bytes
0101 thru 1111 - Reserved - may be greater than 4 for non-numeric data (ASCII or bit-mapped)

The high order nibble of the first byte defines the type of information encoding which is used to represent the parameter. The low order nibble can be read directly to determine the number of bytes used to represent the parameter.

5.11.3.1 *Unsigned Numeric*—This encoding uses a common binary weighting scheme to represent a value by means of discrete incremental steps. One byte affords 256 steps; two bytes yields 65 536 steps, etc. For example, vehicle speed may be encoded with a single unsigned numeric parameter ranging from 0 to 255 km/h. This unsigned numeric value may be up to 4 bytes in length.

5.11.3.2 *Signed Numeric*—This encoding uses a two's complement binary weighting scheme to represent a value by means of discrete incremental steps. One byte affords 256 steps; two bytes yields 65,536 steps, etc. A value can be encoded with a single unsigned numeric parameter ranging from –127 to 128. This signed numeric value may be up to 4 bytes in length.

5.11.3.3 *Bit Mapped*—Bit mapped encoding uses individual bits or small groups of bits to represent status. For every bit which represents status, a corresponding mask bit is required as part of the parameter definition. The mask indicates the validity of the bit for particular applications. A bit mapped parameter may contain 2 bytes; one representing status and one containing the validity mask, or 4 bytes; the first two representing status and the second two containing validity masks. Reference J2178, Part 2 for a discussion of bit mapped data values.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.11.3.4 *BCD*—Conventional BCD encoding is used to represent two numeric digits per byte. The upper nibble is used to represent the most significant digit (0 – 9), and the lower nibble the least significant digit (0 – 9). Up to 8 characters (4 bytes) may comprise a single parameter.

5.11.3.5 *State Encoded*—This encoding uses a common binary weighting scheme to represent up to 256 distinct states. An example is a parameter which represents the status of the ignition switch. Codes "00," "01," "02," and "03" may indicate ignition off, locked, run, and start, respectively. The representation is always limited to 1 byte.

5.11.3.6 *ASCII*—Conventional ASCII encoding is used to represent up to 128 standard characters (MSB = logic 0). An additional 128 custom characters may be represented with MSB = logic 1. Up to four ASCII characters may comprise a single parameter.

5.11.3.7 *Signed Floating Point*—Floating point encoding is used for data that needs to be represented in floating point or scientific notation. Standard IEEE formats shall be used.

5.11.3.8 *Packet*—Packets contain multiple data values, usually related, each with unique scaling. Scaling information is not included for the individual values.

5.11.3.9 *Formula*—Different formulas are currently being discussed to be included. The formula type and constant values used in those formulas will be specified in scaling bytes, or reported as a different test mode.

5.11.4 SCALING BYTE 2—This byte is only relevant for signed and unsigned numbers and for bit mapped parameters. For other parameter types, this field is "padded" with $00. When used with signed and unsigned numbers, this byte specifies scaling by indicating the binary point location. The binary point may be moved 127 places to the right ($FF) or 128 places to the left ($00). The binary point is initially assumed to be to the right of the LSB without this information.

When this byte is used for bit mapped parameters that are reported without a mask, this byte indicates which bits of the PID are supported for the current application. When bit mapped data is reported with a mask, this byte is not used.

When the data type is "formula," this byte is used to indicate the formula type, where:

$00 - $7F are reserved for SAE to define
$80 - $FF are reserved for the manufacturer to define

5.11.5 SCALING BYTES 3 TO N—Optional bytes to be included if multiple bytes of bit mapped data are reported. This mode is more limiting in the number of masks that can be reported than the modes that allow retrieval of the data.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.12 Mode $25—Stop Transmitting Requested Data

5.12.1 FUNCTIONAL DESCRIPTION—The purpose of Mode $25 is to stop all data transmission that was started by any test mode that can request repetitive data. If only an individual data request is to be stopped, then that value can be stopped by sending a request message for the mode that requested the data with a "0" for the data rate. In practice, if a message is already in an outgoing message queue ready to be sent, this would probably not remove that message from the queue, but it would prevent additional messages from being added to the queue. Response is either Mode $65 or general response.

5.12.2 MESSAGE DATA BYTES—(See Figure 17.)

	Data Bytes (Hex)		
	#1	#2 to #n-1	#n
Request from Tool to Vehicle			
Request all data stop	25		
Positive Response from Vehicle to Tool (either of the following is a valid response)			
Confirm all data stopped	65		
General response	7F	Optional data bytes may be included (see Mode $7F description).	Resp. Code - 00

FIGURE 17—MESSAGE DATA BYTES FOR MODE $25

SAENORM.COM : Click to view the full PDF of j2190_199306

5.13 Mode $26—Specify Data Rates

5.13.1 FUNCTIONAL DESCRIPTION—The purpose of Mode $26 is to specify the setting of data rates for data to be transmitted by request messages that can request repetitive data. Those modes include "Data Rate" as a data byte in the request message. Modules will normally have default rates associated with slow, mid, and fast. The slow rate should be about one or two samples per second, the mid rate should be normal handheld scan tool rates of about four to ten samples per second, and the fast rate 20 or more samples per second. Default rates are defined within each module and should be based on the frequency of change typical for the value reported, uses for that data, and the capability of the processor.

5.13.2 MESSAGE DATA BYTES—(See Figure 18.)

	Data Bytes (Hex)					
	#1	#2	#3	#4	#5 to #n-1	#n
Request from Tool to Vehicle						
Request data rate values	26	Slow Rate * - 40 msec per bit	Mid Rate * - 2 msec per bit	Fast Rate * - .1 msec per bit		
Positive Response from Vehicle to Tool (either of the following is a valid response)						
Confirm data rate values set	66					
General response	7F	Optional data bytes may be included (see Mode $7F description).				Resp. Code - 00

* NOTE: A value of $00 should cause the data to be reported as fast as possible. A value of $FF will cause no change in the current data rate.

FIGURE 18—MESSAGE DATA BYTES FOR MODE $26

SAENORM.COM : Click to view the full PDF of j2190_199306

5.14 Mode $27—Security Access Mode

5.14.1 FUNCTIONAL DESCRIPTION—The primary purpose of this mode is to restrict unauthorized intrusion into the on-board controller. Improper programs could potentially damage the electronics or other vehicle components or risk the vehicle's compliance to emission or safety standards. This mode is intended to be used to implement the data link security measures defined in SAE J2186.

The external device will request the controller to "Unlock" itself by sending Request #1. The controller will respond by sending a "Seed" using Response #1. The external device will respond by returning a "Key" number back to the controller using Request #2. The controller would compare this "Key" to one internally stored. If the two numbers agree, then the controller will enable ("Unlock") the external device's access to specific test modes and indicate that with Mode $67, Response #2. If upon two attempts of a Request #2 where the two keys do not compare, then the controller will insert a 10 s time delay before allowing further attempts. This time delay will also be required before responding to a Mode $27 request #1 for each controller power-on.

If a device supports security, but is already unlocked when a Request #1 is received, that device should respond with a Response #1 message with a seed of $00 00. A test device could use this method to determine if a device is locked by checking for a non-zero seed.

The security system will not prevent normal diagnostic or vehicle communications between external devices and the on-board controller. Proper "Unlocking" of the controller is a prerequisite to the external device's ability to perform some of the more critical functions such as reading specific memory locations within the controller, downloading information to specific locations, or downloading routines for execution by the controller. In other words, the only access to the controller permitted while in a "locked" mode is through the product specific software. This permits the product specific software to protect itself from unauthorized intrusion.

Devices that provide security should support reject messages if a secure mode is requested while the device is locked. The reject message to be returned is a general response message Mode $7F, with a response code $33 to indicate the product is secured.

Some devices could support multiple levels of security, either for different functions controlled by the device, or to allow different capabilities to be exercised. These additional levels of security can be accessed by using requests #3 and #4, etc. The second data byte of the request for seed should always be an odd number, and the second data byte of the message to send the key should be the next even number.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.14.2 MESSAGE DATA BYTES—(See Figure 19.)

	Data Bytes (Hex)					
	#1	#2	#3	#4	#5 to #n-1	#n
Request #1 from Tool to Vehicle (data byte #2 can be an odd number greater than $01 if additional levels of security are supported)						
Request for "Seed"	27	01 (03, etc.)				
Positive Response #1 from Vehicle to Tool						
Return "Seed"	67	01 (03, etc.)	Seed (00 00 if module is not locked)		Optional data bytes may be specified by the vehicle manufacturer	
Request #2 from Tool to Vehicle (data byte #2 can be an even number one greater than data byte #2 of request #1 if additional levels of security are supported)						
Send "Key"	27	02 (04, etc.)	Key			
Positive Response #2 from Vehicle to Tool						
Security access	67	02 (04, etc.)	Optional data bytes may be specified by the vehicle manufacturer			
General response	7F	Optional data bytes may be included (see Mode $7F description).				Resp. Code - 00

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 19—MESSAGE DATA BYTES FOR MODE $27

5.15 Mode $28—Disable Normal Message Transmission

5.15.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to inhibit the on-board device from transmitting normal operating data on the link, while still performing other functions normally. The device will continue to operate in whatever diagnostic mode it was operating in prior to the Mode $28 command.

If an unsafe or undesirable vehicle operating condition would result from the lack of normal messages, then this mode could cause all nonessential messages to be inhibited. The optional manufacturer specific "level" data byte could be used to indicate which normal mode messages to disable. Defining which messages to disable is determined by the system designer to ensure safe vehicle operation. When using this test mode, some provision must be made to allow for other devices that rely on information from the silenced device so that they do not set diagnostic trouble codes due to the lack of required information.

One use for this test mode is to reduce message traffic on the data link. This would make more time available for diagnostic messages. Another use for this mode is to allow the test equipment to emulate the remote device for diagnostic purposes. In this scenario, the test device would send a Mode $28 request to the device to be emulated. The test device would then respond to all normal communication request messages directed to that device, most likely with data intended to cause a known response by a system that uses the information in the response. The test device can then observe the actions of those systems.

5.15.2 MESSAGE DATA BYTES—(See Figure 20.)

SAENORM.COM : Click to view the full PDF of j2190_199306

	Data Bytes (Hex)			
	#1	#2	#3 to #n-1	#n
Request from Tool to Vehicle				
Request disable normal message transmission	28	level (opt)		
Positive Response from Vehicle to Tool (either of the following is a valid response)				
Confirm normal message transmission disabled	68	level (opt)		
General response	7F	Optional data bytes may be included (see Mode $7F description).		Resp. Code - 00

FIGURE 20—MESSAGE DATA BYTES FOR MODE $28

5.16 Mode $29—Enable Normal Message Transmission

5.16.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to cause an on-board device to resume normal communications after previously disabling these messages by a Mode $28 command. The device will continue to operate in whatever diagnostic modes were set by previous messages, such as executing on-board routines.

5.16.2 MESSAGE DATA BYTES—(See Figure 21.)

	Data Bytes (Hex)		
	#1	#2 to #n-1	#n
Request from Tool to Vehicle			
Request enable normal message transmission	29		
Positive Response from Vehicle to Tool (either of the following is a valid response)			
Confirm normal message transmission enabled	69		
General response	7F	Optional data bytes may be included (see Mode $7F description).	Resp. Code - 00

FIGURE 21—MESSAGE DATA BYTES FOR MODE $29

SAENORM.COM : Click to view the full PDF of j2190_199306

5.17 Mode $2A—Request Diagnostic Data Packet(s)

5.17.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to request diagnostic data packets that contain data values, such as analog inputs and outputs, digital inputs and outputs, freeze frame data, calculated values, bit mapped fault code data, and system status information. This Mode differs from Modes $21 through $23 in that multiple data packets can be requested, each containing multiple data values. These data packets can either be predefined in the on-board control module or dynamically defined using Mode $2B or $2C.

The on-board module will respond to this message by transmitting multiple response messages specific to that device. The data packet ID Bytes in the request allow multiple Mode $2A data packets to be requested. Data packets should be defined which include data that is commonly used together, or changes together.

Uses for these different data packets include data such as wheel speeds for an ABS system. Data can be returned quickly with a minimal length request and response. Another use for these different data packets is to return values that do not change, such as VIN and option content. These values need to be known one time only during testing. Other data bytes may contain present values of analog and discrete device I/O, device software flags and status words, and failure codes.

5.17.2 MESSAGE DATA BYTES—(See Figure 22.)

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Request diagnostic data packets	2A	Data rate *	DPID #	Additional optional data packet IDs may be requested to fill the message up to the maximum number of available data bytes.	
Positive Response from Vehicle to Tool (Multiple response messages will be sent if multiple DPIDs requested, or if data rate requested periodic data reporting)					
Report of diagnostic data packet - repeated for each requested data packet	6A	DPID #	Data Byte	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.	
Vehicle Response to Request to stop sending data (either multiple $6A messages or a single $7F message is a valid response)					
Report stop sending diagnostic data packets	6A	DPID# of DPID req.	(any additional DPID #s in this message could appear to be data)		
Repeat report stop sending diagnostic data packets for each DPID # requested	6A	DPID # of DPID req.			
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. code - 00

* same data rate definition as Data Byte #3 for Mode $21

FIGURE 22—MESSAGE DATA BYTES FOR MODE $2A

SAENORM.COM : Click to view the full PDF of j2190_199306

5.18 Mode $2B—Dynamically Define Data Packet by Single Byte Offsets

5.18.1 FUNCTIONAL DESCRIPTION—This test mode can be used to dynamically define data packets that can subsequently be requested using a Mode $2A request. This mode is a special case alternative to using multiple Mode $2C messages, because a single request message can be used to completely define the contents of a data packet. All other methods to specify the data values to be included in data packets using Mode $2C would also require multiple messages, therefore, special modes to build data packets by PID or memory address are not included.

If the parameter for any offset value is longer than 1 byte, then offset data bytes in the request should be filled with $FF for each additional byte in the data value requested (see message example). This forces the data value byte locations in the response message to mirror the offset value byte locations in the request message.

5.18.2 MESSAGE DATA BYTES—(See Figure 23.)

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Define diagnostic data packet	2B	DPID #	Offset	Additional optional data offset values may be added to fill the message up to the maximum number of available data bytes.	
Positive Response from Vehicle to Tool (either of the following is a valid response)					
Confirm diagnostic data packet defined	6B	DPID #			
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. Code - 00

FIGURE 23—MESSAGE DATA BYTES FOR MODE $2B

SAENORM.COM : Click to view the full PDF of j2190_199306

5.18.3 MESSAGE EXAMPLE—Figure 24 is an example of messages required to define a data packet by offset and then request that data packet. The data packet can be defined with a single message and requested with another message.

	Data Bytes (Hex)							
	#1	#2	#3	#4	#5	#6	#7	#8
Define Diagnostic Data Packet #22 consisting of: Value at offset $30 (1 byte value) Value at offset $12 (2 byte value) Value at offset $26 (2 byte value) Value at offset $05 (1 byte value)								
Define diagnostic data packet #22	2B	22	30	12	FF	26	FF	05
Positive Response from Vehicle to Tool (either of the following is a valid response)								
Confirm data packet defined #22	6B	22						
General response	7F	2B	22	Resp. Code - 00				
Request from Tool to Vehicle								
Request data packet #22	2A	Data rate	22					
Positive Response from Vehicle to Tool								
Report data packet #22	6A	22	1 byte value from offset $30	2 byte value from offset $12		2 byte value from offset $26		1 byte value from offset $05

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 24—DATA PACKET EXAMPLE

5.19 Mode $2C—Dynamically Define Diagnostic Data Packet

5.19.1 FUNCTIONAL DESCRIPTION—This test mode can be used to dynamically define data packets that can subsequently be requested using a Mode $2A request. This single test mode is an alternative to multiple modes to define data packets by offset, PID, and memory address.

This mode allows data packets to be defined in different ways, and also allows a single data packet to include data specified in different ways. Multiple messages must be sent to define a single data packet. Each message constructs a portion of the data packet, and includes an indication of how the data is being requested, the starting byte of the data packet, and the number of bytes to be specified for the data packet.

Data packets can be shortened by sending a data packet definition with the starting byte indicating the first data byte not to be included, and 0 as the length. This should be interpreted as 0 bytes at the starting location, which means that data byte is not in the response message.

Data packet definitions can be cleared by sending a message with 1 as the starting data byte and 0 as the number of bytes. Requesting this data packet would result in a response which would include the mode value, data packet ID, and no data.

Data packet definitions can be removed by sending a message with 0 as the starting data byte and 0 as the number of bytes. This would free memory space, if needed.

An optional feature of this mode is to request the present definition for the data packet. This is accomplished by sending a Mode $2B with the data packet ID as the only additional data byte. The result is multiple $6B responses, each including data bytes that resemble the $2B messages that created the data packet.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.19.2 MESSAGE DATA BYTES—(See Figure 25.)

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Define diagnostic data packet	2C	DPID #	Data byte #3 bits 7,6 if 00 - define by offset (1 byte) if 01 - define by PID (2 bytes) if 10 - define by memory address (3 bytes) if 11 - manufacturer defined bits 5,4,3 starting byte for data, where 001 is the first byte after the DPID # bits 2,1,0 number of data bytes for this parameter Data bytes starting at byte #4 contain either the offset, PID, memory address, or manufacturer defined value for the data to be included in the packet		
Positive Response from Vehicle to Tool (either of the following is a valid response)					
Confirm diagnostic data packet defined	6C	DPID #	Request data byte #3		
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. Code - 00

FIGURE 25—MESSAGE DATA BYTES FOR MODE $2C

SAENORM.COM : Click to view the full PDF of j2190_199306

5.19.3 MESSAGE EXAMPLE—Figure 26 shows the messages required to define a data packet and then request that data packet. Data packet #26 is to be defined as:

Value for PID $01 04 (1 byte data value)
Value at offset $32 (2 byte data value)
2 bytes of data at memory address $24 81 70

This definition requires three sets of request / report messages, and the data packet request requires one set of messages.

	Data Bytes (Hex)						
	#1	#2	#3	#4	#5	#6	#7
Request from Tool to Vehicle							
PID $01 04 is 1 byte value to be included at data byte 3 (1 byte after DPID #)	2C	26	49 --- 01 001 001 1 1	01	04		
Positive Response from Vehicle to Tool (either of the following is a valid response)							
PID $01 04 at data byte 3	6C	26	49				
General response	7F	2C	26	49	Resp. Code - 00		
Request from Tool to Vehicle							
Offset $32 is 2 byte value to be included at data byte 4 (2 bytes after DPID #)	2C	26	12 --- 00 010 010 2 2	32			
Positive Response from Vehicle to Tool (either of the following is a valid response)							
Offset $32 starting at data byte 4	6C	26	12				
General response	7F	2C	26	12	Resp. Code - 00		

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 26A—DATA PACKET DEFINITION EXAMPLE

	Data Bytes (Hex)						
	#1	#2	#3	#4	#5	#6	#7
Request from Tool to Vehicle							
2 bytes starting at address $24 81 70 to be included at data bytes 6 and 7	2C	26	A2 10 100 010 4 2	24	81	70	
Positive Response from Vehicle to Tool (either of the following is a valid response)							
Addresses $24 81 70 and $24 81 71 at data bytes 6 and 7	6C	26	A2				
General response	7F	2C	26	A2	Resp. Code - 00		
After the data packet is defined, the tool can request the data packet from the vehicle							
Request data packet #26	2A	26					
Positive Response from Vehicle to Tool							
Report data packet #26	6A	26	1 byte value for PID $01 04	2 byte value from offset $32		2 bytes at memory address $24 81 70	

FIGURE 26B—DATA PACKET DEFINITION EXAMPLE (CONTINUED)

SAENORM.COM : Click to view the full PDF of j2190_199306

5.20 Mode $2F—Input/Output Control by PID

5.20.1 FUNCTIONAL DESCRIPTION—This capability allows the tester to verify proper operation of external input and output components and circuitry by isolation techniques. Real world sensor inputs can be temporarily bypassed, and direct control of output devices can be achieved. Since substitution may cause the control module to operate in a manner which is unsuitable or unsafe for "on-the-road" operation, precautions must be taken to ensure safe operation. The substituted value is used only for the duration of the diagnostic procedure, and when the module is returned to normal operation, or control of the data value is returned to the vehicle, then the substituted value reverts back to the normal value determined by the control system.

Parameter IDs (PIDs) are assigned by the manufacturer for those input, output, and intermediate (calculated) values that are allowed to be substituted. The PID assignments can be the same PID values that are used with a Mode $22 request for data by PID.

5.20.2 MESSAGE DATA BYTES—(See Figure 27.)

	Data Bytes (Hex)					
	#1	#2	#3	#4	#5 to #n-1	#n
Request from Tool to Vehicle						
Request Input/Output control by PID	2F	PID (high byte)	PID (low byte)	Data Byte #4 (optional) if not specified - control of identified data value is disabled if specified - value is substituted for identified data value in the module Data Bytes after #4 may be used if needed to specify substitute value		
Positive Response from Vehicle to Tool (either of the following is a valid response)						
Confirm Input/Output control by PID	6F	PID (high byte)	PID (low byte)	Req. data byte 4 (opt)		
General response	7F	Optional data bytes may be included (see Mode $7F description).				Resp. Code - 00

FIGURE 27—MESSAGE DATA BYTES FOR MODE $2F

SAENORM.COM : Click to view the full PDF of j2190_199306

5.21 Mode $30—input/output Control By Data Value Id

5.21.1 FUNCTIONAL DESCRIPTION—This mode is identical in purpose and function to mode $2f, except that the parameter to be controlled is specified by a 1 byte data value ID instead of a 2 byte PID.

Data value IDs are assigned by the manufacturer for those input, output, and intermediate (calculated) values that are allowed to be substituted.

5.21.2 MESSAGE DATA BYTES—(See Figure 28.).

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Request Input/Output control by data value ID	30	Data value ID	Data Byte #3 (optional) if not specified - control of identified data value is disabled if specified - value is substituted for identified data value in the module Data Bytes after #3 may be used if needed to specify substitute value		
Positive Response from Vehicle to Tool (either of the following is a valid response)					
Confirm Input/Output control by data value ID	70	Data value ID	Req. data byte 3 (opt)		
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. Code - 00

FIGURE 28—MESSAGE DATA BYTES FOR MODE $30

SAENORM.COM : Click to view the full PDF of j2190_199306

5.22 Modes $31 to $33—Perform Diagnostic Routine by Test Number

5.22.1 FUNCTIONAL DESCRIPTION—The purpose of these modes is to execute diagnostic tests and obtain test results. The diagnostic routines are manufacturer defined and executed in the control module by referencing the test number. Each module may support up to 255 distinct diagnostic routines. Definition of the routines is defined by the module manufacturer.

Reported results may consist of any set of return information, such as measured test values or fault codes.

Three sets of messages are used to perform tests resident in an on-board control module. These messages perform three functions:

Mode $31 - Enter/start Diagnostic Routine by Test Number
Mode $32 - Exit/stop Diagnostic Routine by Test Number
Mode $33 - Request Diagnostic Routine Results by Test Number

The tests are known by the module and identified by a test number. They are usually permanently stored in the module, but may have been downloaded by Modes $34 and $36.

These test numbers could either be tests that run instead of normal operating code, or could be routines that are enabled in this mode and execute with the normal operating code. In the first case, normal system operation for the controller being tested is not possible. In the second case, multiple diagnostic routines can be enabled that run while all other parts of the system are functioning normally.

Any combination of the messages can be used, depending on the implementation in the module. Some examples are:

- Tester sends a Mode $31 message to start a test. The module reports that the test has started with a Mode $71 and runs the test until a Mode $32 is sent to stop the test. The module exits the test and informs the tester with a Mode $72 that the test has stopped. The tester then requests test results with a Mode $33, and the module reports the results with a Mode $73 message.
- Tester sends a Mode $31 message to start a test. The Module confirms the test has started, exits when the test is done and reports results using Mode $73.
- Tester sends a Mode $31 message to start a test. The Module starts the test and periodically reports results using Mode $73 until a Mode $32 is sent to stop the test.
- The on-board controller starts a test automatically, stops the test automatically and reports results using a Mode $73 message automatically. This requires no request messages from the tester.

These examples demonstrate both maximum and minimum use of the available message. Test equipment needs to know what tests are available, which messages are required, and how to interpret the results.

The general purpose response, Mode $7F, may also be used instead of a Mode $71 or Mode $72 message to acknowledge entry or exit from a diagnostic routine.

SAENORM.COM : Click to view the full PDF of j2190_199306

5.22.2 MESSAGE DATA BYTES—(See Figure 29A.)

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Request entry by test number	31	Test #	Additional optional data bytes may be added to the request to fill the message up to the maximum number of available data bytes.		
Positive Response from Vehicle to Tool (either of the following is a valid response)					
Report entry by test number	71	Test #	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.		
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. code - 00
Request from Tool to Vehicle					
Request exit by test number	32	Test #	Additional optional data bytes may be added to the request to fill the message up to the maximum number of available data bytes.		
Positive Response from Vehicle to Tool (either of the following is a valid response)					
Report exit by test number	72	Test #	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.		
General response	7F	Optional data bytes may be included (see Mode $7F description).			Resp. code - 00

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 29A—MESSAGE DATA BYTES FOR MODES $31 THROUGH $33

	Data Bytes (Hex)				
	#1	#2	#3	#4 to #n-1	#n
Request from Tool to Vehicle					
Request results by test number	33	Test #	Additional optional data bytes may be added to the request to fill the message up to the maximum number of available data bytes.		
Positive Response from Vehicle to Tool					
Report results by test number	73	Test #	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.		

Note: Mode $33 reports the results of a test that was run on request from a tool. This differs from Mode $05, which reports the results of the latest on-board monitoring test that was run to meet OBD regulations.

FIGURE 29B—MESSAGE DATA BYTES FOR MODES $31 THROUGH $33 (CONTINUED)

SAENORM.COM : Click to view the full PDF of j2190_199306

5.23 Modes $34 to $37—Data Transfer

5.23.1 FUNCTIONAL DESCRIPTION—These modes allow the transfer of data either from a tester to an on-board module or from an on-board module to a tester. Modes defined are:

Mode $34 - Request Download - tool to module
Mode $35 - Request Upload - module to tool
Mode $36 - Data Transfer
Mode $37 - Request Data Transfer Exit

Download is defined as data transfer from the tool to the controller in the vehicle. Upload is defined as data transfer from the controller in the vehicle to the tool.

5.23.2 MESSAGE DATA BYTES—(See Figure 30A.)

	Data Bytes (Hex)		
	#1	#2 to #n-1	#n
Request from Tool to Vehicle			
Request download	34	Messages used to request download are defined by the vehicle/module manufacturer. This Mode value is reserved for a download request message.	
Positive Response from Vehicle to Tool (either of the following is a valid response)			
Report download status	74	Messages used to request download are defined by the vehicle/module manufacturer. This Mode value is reserved for the response to a Mode $34.	
General response	7F	Optional data bytes may be included (see Mode $7F description).	Resp. code - 00

FIGURE 30A—MESSAGE DATA BYTES FOR MODES $34 THROUGH $37

SAENORM.COM : Click to view the full PDF of j2190_199306

	Data Bytes (Hex)		
	#1	#2 to #n-1	#n
Request from Tool to Vehicle			
Request upload	35	Messages used to request upload are defined by the vehicle/module manufacturer. This Mode value is reserved for an upload request message.	
Positive Response from Vehicle to Tool (either of the following is a valid response)			
Report upload status	75	Messages used to request download are defined by the vehicle/module manufacturer. This Mode value is reserved for the response to a Mode $35.	
General response	7F	Optional data bytes may be included (see Mode $7F description).	Resp. code - 00
Transfer Data			
Data transfer	36	Messages used to perform the data transfer are defined by the vehicle/module manufacturer. This Mode value is reserved for the data transfer request message.	
Data transfer	76	Messages used to perform the data transfer are defined by the vehicle/module manufacturer. This Mode value is reserved for the response to a Mode $36.	
Request from Tool to Vehicle			
Request exit	37	Messages used to exit data transfer are defined by the vehicle/module manufacturer. This Mode value is reserved for the exit data transfer request message.	
Positive Response from Vehicle to Tool (either of the following is a valid response)			
Report exit	77	Messages used to exit data transfer are defined by the vehicle/module manufacturer. This Mode value is reserved for the response to a Mode $37.	
General response	7F	Optional data bytes may be included (see Mode $7F description).	Resp. code - 00

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 30B—MESSAGE DATA BYTES FOR MODES $34 THROUGH $37 (CONTINUED)

5.24 Modes $38 to $3A—Perform Diagnostic Routine at a Specified Address

5.24.1 FUNCTIONAL DESCRIPTION—These test modes are used to execute code resident in the on-board controller at the specified address. This executable code may be permanently stored in the controller or may have been downloaded using Modes $34 and $36.

These three sets of messages are used to perform tests resident in an on-board control module. These messages perform three functions:

Mode $38 - Enter Diagnostic Routine by Address
Mode $39 - Exit Diagnostic Routine by Address
Mode $3A - Request Diagnostic Routine Results

5.24.2 MESSAGE DATA BYTES—(See Figure 31A.)

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 31A—MESSAGE DATA BYTES FOR MODES $38 THROUGH $3A

	Data Bytes (Hex)						
	#1	#2	#3	#4	#5	#6 to #n-1	#n
Request from Tool to Vehicle							
Request test exit by starting address	39	Starting Address			(opt)	Additional optional data bytes may be added to the request to fill the message up to the maximum number of available data bytes.	
		high byte	mid byte	low byte			
Positive Response from Vehicle to Tool (either of the following is a valid response)							
Report exit by starting address	79	Starting Address			Exit Status (opt)	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.	
		high byte	mid byte	low byte			
General response	7F	Optional data bytes may be included (see Mode $7F description).					Resp. code - 00
Request/Report Diagnostic Routine Results							
Request results	3A	(opt.)	Additional optional data bytes may be added to the request to fill the message up to the maximum number of available data bytes.				
Positive Response from Vehicle to Tool							
Report results	7A	Additional optional data bytes that are in response to the request may be added to fill the message up to the maximum number of available data bytes.					

SAENORM.COM : Click to view the full PDF of j2190_199306

FIGURE 31B—MESSAGE DATA BYTES FOR MODES $38 THROUGH 3A (CONTINUED)

5.25 Mode $3B—Write Data Block

5.25.1 FUNCTIONAL DESCRIPTION—The purpose of this mode is to provide a means for the external test device to change the contents of a data block. The data block numbers and associated memory locations need to be known by the on-board device. This mode does not allow off-board test equipment to change any memory locations other than for those data blocks predefined in the on-board device.

The number of data bytes included in the request message for each block number depends on the system design and intent of the usage for that block.

Possible uses for this mode are:

Clear non-volatile memory
Reset learned values in a single table
Set option content
Set Vehicle Identification Number (VIN)
Change calibration values

This mode can be used to clear or reset tables stored in non-volatile memory. A data block number could be defined as memory locations that need to be either cleared or reset to predefined initial values, such as learned values in a table. Sending the data block number with no data bytes could cause the on-board system to clear/reset all learned values associated with that data block number to known nominal values.

Another use is to allow the VIN to be input. A data block number could be defined to expect a portion of the VIN. Multiple messages would be required to send the entire 17-character VIN. Whatever data values were included in the request message would be stored in the memory locations known by the on-board system for VIN.

In some systems, this mode could be used for programming calibrations or vehicle specific information in the assembly plant or service without using secured methods for reprogramming. If desired that those values not be changed after production, the capability to modify those data blocks could be disabled after the initial change.

SAENORM.COM : Click to view the full PDF of j2190_199306