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

SURFACE VEHICLE INFORMATION REPORT

SAE J1924

CAN-CELLED AUG2000

Issued 1992-12
Cancelled 2000-08

Supserseding J1924 DEC1992

OEM/Vendor Interface Specification for Vehicle Electronic Programming Stations

Foreword—In our previous meeting, we tried to define an OEM to Vendor communications protocol. There were too many differences between what each vendor needed to establish a common RS-232 based protocol. We decided to have each vendor supply its own communications program. The OEM would host the vendor communications programs on the OEM production line computers.

1. ***Scope***—The purpose of the SAE Information Report is to address the method of loading vehicle electronic controllers with chassis and customer specific parameters. This specification shall establish an interface definition. The interface definition must be mutually agreeable to truck OEMs and vendors. The purpose of this specification is not to answer the large protocol issues raised by systems such as GM's MAP.

1.1 **SAE Standard**—In the future, SAE may use this specification as a basis for an OEM/Vendor interface specification standard.

2. ***References***—There are no referenced publications specified herein.

3. ***Programming System***

3.1 **Plant System**—Each OEM will create its own system of programming vehicle control modules. This specification only addresses the programming station setup and the OEM/vendor interface. It does not suggest any specifies concerning the OEM's manufacturing system or production facilities.

3.2 **Station Setup**

3.2.1 PROPOSED STATION—The link between the OEM computer and the vendor interface tool is RS232c. The OEM host communications port shall support the 9 pin standard. The OEM host computer may have multiple ports or one common port. The common port is possible if the OEM provides an external switch. See the example in Figure 1.

TO PLACE A DOCUMENT ORDER; (724) 776-4970 FAX: (724) 776-0790
SAE WEB ADDRESS http://www.sae.org

Printed in U.S.A.

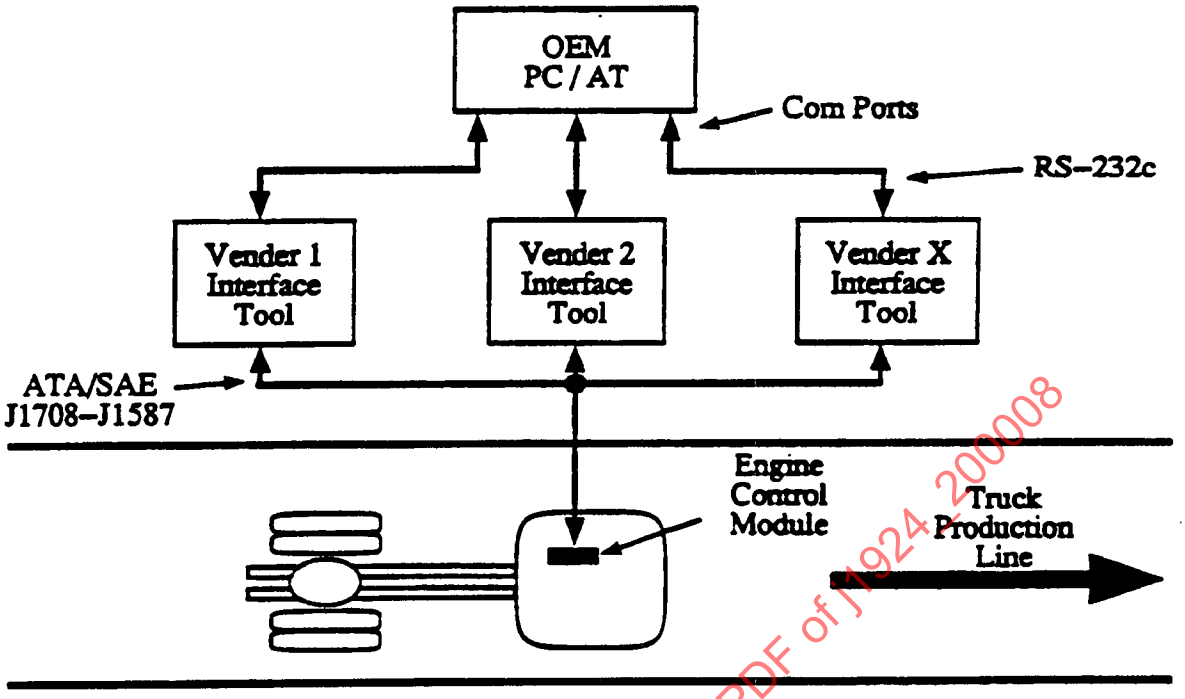


FIGURE 1—PRODUCTION LINE

3.2.2 PROPOSED SOFTWARE—The OEM production line PC shall execute an OEM production line program and the vendor communications programs. The OEM program shall gather customer and chassis specific parameter information from its manufacturing system and invoke the vendor program. The vendor program shall download the parameters via the host computer's communications port to the vehicle control module. The vendor program shall also return parameter verification and warranty data to the OEM program. The vendors shall supply the communication program to the OEMs as an executable program file (.exe). See example in Figure 2.

3.2.3 FUTURE DIRECTION—The OEMs want to phase out vendor tools used on the production line. In the future, an ATA communications card will replace the vendor interface tools. The OEMs intend to evolve to one ATA/SAE communications card which will plug into the host computer. This card will replace the vendor interface tools. The OEMs anticipate that the definition process will begin in the Spring of 1989. This should lead to a production line prototype in the early 1990s.

4. Interface Files

4.1 File Characteristics—Each interface file is an ASCII text file with multiple records. An end of line marker terminates each record and delimiters separate each field. A maximum character count limits the length of each field. If a larger field is needed, a field may contain a file name pointing to a separate file. An end of file marker terminates the file.

4.1.1 TEXT FILE FORMAT—The record fields may contain any printable (20H to 7EH) ASCII characters except the field delimiter. The comma shall delimit record fields and a carriage return line feed shall terminate each record. A control-Z shall mark the end of file.

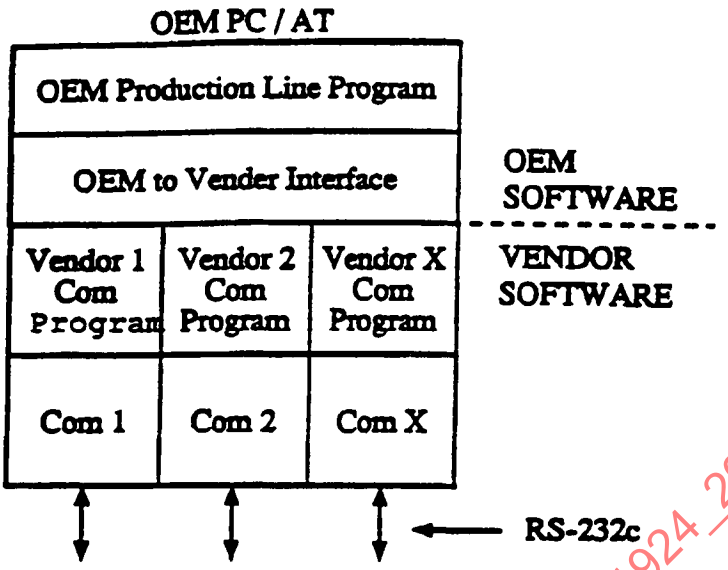


FIGURE 2—PRODUCTION LINE COMPUTER

4.1.2 CASE SENSITIVITY—The files may contain either upper or lower case alpha characters. However, the OEM and vendor programs are not required to make "case sensitive" tests. The programs shall interpret the upper and lower case alpha characters (A through Z) as being the same. For example, the programs shall interpret "COMx" as equivalent to "COMX".

4.1.3 FIELD LENGTH—The field lengths shall be limited to 64 characters. Any field requiring more than 64 characters shall be passed as a file. Field may contain a file name which points to a separate file.

4.1.4 FILE NAMES—All vendor-supplied files shall conform to the following naming conventions. The file name shall include the vendor's company name, the version number, and the type of file. The first few letters of the file name shall be the vendor's company mnemonic and the following letters, the version number. The file qualifier shall identify the type of file.

4.1.4.1 Vendor Mnemonic—The first 5 characters of the file name shall identify vendor company name.

4.1.4.2 Program Version—The next 3 characters following the vendor mnemonic shall identify the version of the vendor file.

4.1.4.3 File Type—The 3-character file qualifier shall identify the file type. The following list itemizes the possible file types. See the example in Figure 3.

a. Executable file = .EXE
b. Verification file = .VER
c. Definition file = .DEF
d. Parameter file = .PAR
e. Remarks file = .REM

4.1.4.4 Examples

a. CECOM120.DEF—Definition file version 1.20 for Cummins Electronics Company
b. CATPL100.VER—Verification file version 1.00 for Caterpillar
c. DDALS210.EXE—Executable file version 2.10 for Detroit Diesel Allison

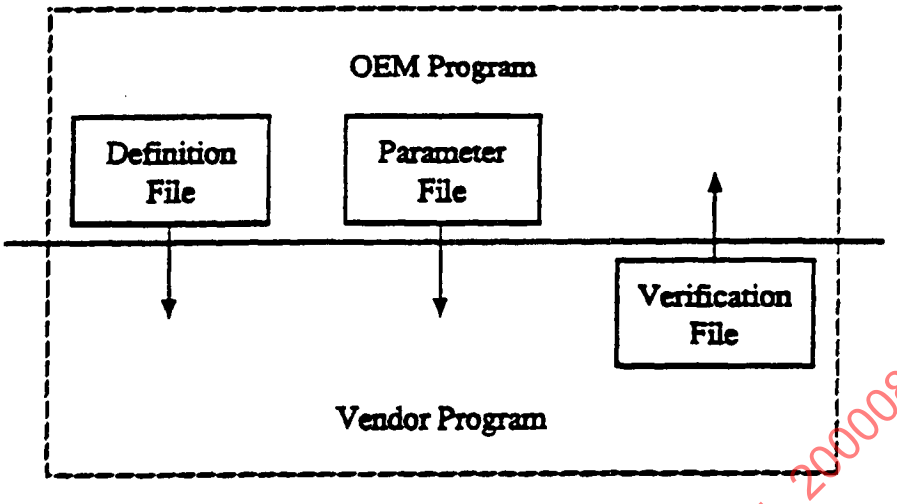
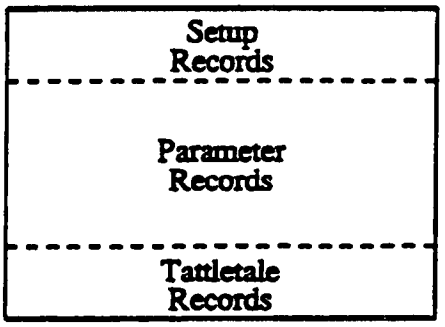


FIGURE 3—OEM/VENDOR INTERFACE FILES

4.2 Definition Files—The definition file provides descriptive information for both the OEM and vendor programs. It contains a set of records necessary to describe vehicle controller parameters and tattletale warranty information. The definition file shall also contain setup configuration data for the OEM and vendor programs. Each definition file record will hold different sets of data and, therefore, have a unique record structure. The OEM and vendor shall jointly create and maintain the definition file. They shall perform these operations off-line. See Figures 4 through 7 as examples.

	OEM Name	Vendor Name	Type	Units	Default Value	Lower Limit	Upper Limit	Increment
	SpeedCal	BC00	BOTH	RPM	3000	0	3000	.5
	SpeedLim	ADFF	BOTH	MPH	127	0	127	.5
2	RPMSpeed	92CF	BOTH	RPM	3000	0	3000	10
**	LowCruise	ADC0	BOTH	MPH	127	0	127	.5

FIGURE 4—SAMPLE DEFINITION TABLE



Records grouped by Record Type
with remark records placed where needed

FIGURE 5—DEFINITION FILE STRUCTURE

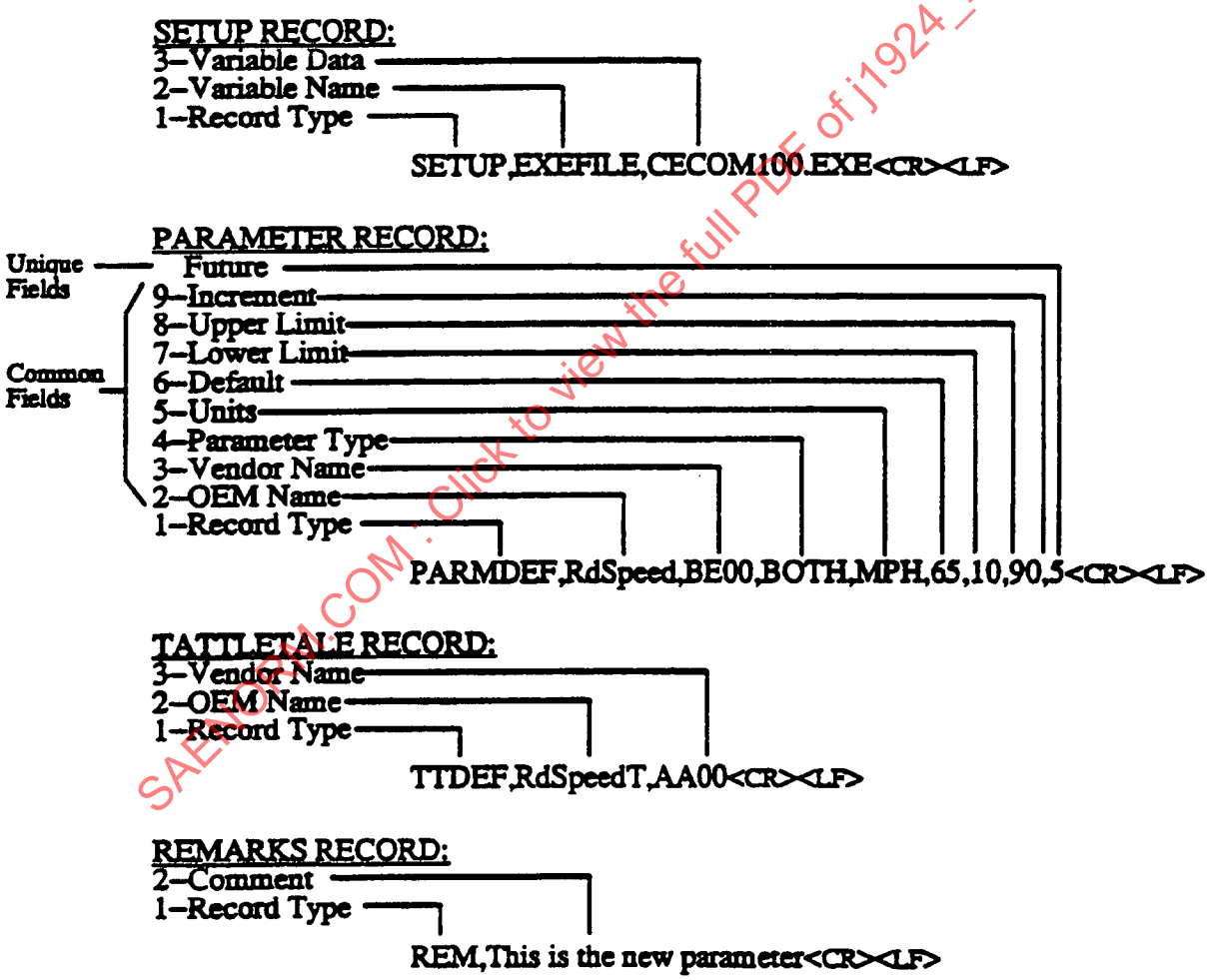


FIGURE 6—DEFINITION FILE RECORD FORMATS

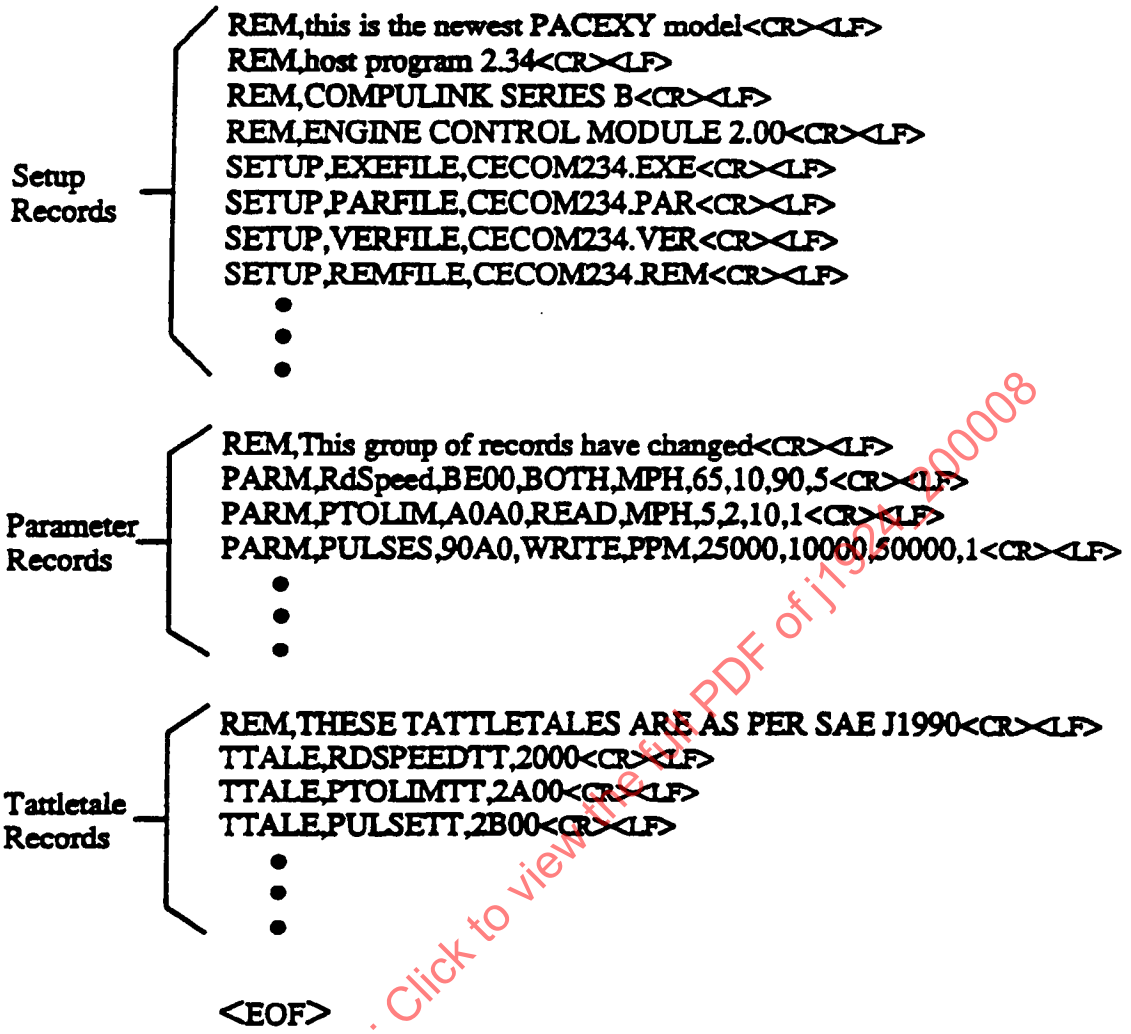


FIGURE 7—SAMPLE DEFINITION FILE

4.2.1 DEFINITION FILE RECORD TYPES—Each of the records shall contain a record type field. The record type field shall be the first field of each record. It shall indicate what the remaining fields of the record will contain. For example, if the record type field contains the ASCII character "P," then the following fields of that record will contain parameter definition information. The first character of this field shall be one of the following:

a. "S"—setup definition type record
b. "P"—parameter definition type record
c. "T"—tattletale definition type record
d. "R"—remarks type record

Other characters may follow the first character but only the first is significant to the OEM and vendor programs.

4.2.2 DEFINITION FILE SETUP RECORDS—Setup records shall provide configuration data for the OEM and vendor programs. These programs shall read the setup records to find such things as interface file names. These records shall contain a field for a variable name and variable data. The variable name indicates the type of data contained in the record. The variable data field shall contain the record data. The following define types of setup records.

a. Executable File Name—(vendor supplied communications program)
Purpose: enable the vendor to define a version specific executable program file name for the OEM program.
Variable Name: "EXEFILE"
Variable Data: executable file name
Example: EXEFILE,CECOM234.EXE

b. Parameter File Name
Purpose: enable the vendor to define a version specific parameter file name for the OEM program.
Variable Name: "PARFILE"
Variable Data: parameter file name
Example: PARFILE,CATPL002.PAR

c. Verification File Name
Purpose: enable the vendor to define a version specific verification file name for the OEM program.
Variable Name: "VERFILE"
Variable Data: verification file name
Example: VERFILE,DDALS001.VER

d. Remarks File Name
Purpose: enable the vendor to define a version specific comments file name for the OEM.
Variable Name: "REMFILE"
Variable Data: remarks or "readme" file name
Example: REMFILE,BENDX100.REM

4.2.3 DEFINITION FILE PARAMETER RECORDS—The parameter records contain the information necessary to translate and describe vehicle controller settings. Each record contains data required to define one setting. Each record contains groups of common and unique fields. The common fields define the basic attributes of each parameter which are common to all vendors and OEMs. The unique fields shall define parameter attributes which are unique to the vendor.

4.2.3.1 Parameter Record—Common Fields—The common fields define parameter attributes which are common to all vendors and OEMs. The following is a list of common fields:

a. OEM Name—The OEM name identifies the parameter which may be changed or verified.
b. Vendor Name—The vendor name identifies a label or address used by the vendor program.
c. Parameter Type—The parameter type indicates the accessibility of the parameter to the OEM program. The first character of the third field shall be one of the following:

"R"—read only
"W"—write only
"B"—both read and write
"S"—special types of parameters

Other characters may follow the first character but only the first character is significant to the OEM and vendor programs.

d. Units—The units field contains the parameter value units. As an example, the units field may have the following values:

"PPM"—pulses per mile
"MPH"—miles per hour
"RPM"—revolutions per minute
" "—(blank character)—no units

e. Default—The default field contains the value of the parameter as supplied by the vendor unchanged.
f. Lower Limit—The lower limit field contains the lowest value to which the parameter may be set.
g. Upper Limit—The upper limit field contains the highest value to which the parameter may be set.
h. Increment—The increment field contains the finest resolution to which the parameter may be set.

4.2.3.2 *Parameter Record—Unique Fields*—The unique fields define items specific to the vendor. These fields shall be defined by the vendor for the OEM as needed.

4.2.4 DEFINITION FIELD TATTLETALE RECORDS—The tattletale records provide a definition of data items which the OEM may use for warranty purposes. Each record shall contain OEM name and a vendor name. These records shall contain additional fields in the future when a new standard for tattletale information is established.

a. OEM Tattletale Name—The OEM tattletale name identifies the parameter tattletale information which the OEM program may read for warranty purposes.
b. Vendor Name—The vendor name identifies a label or address used by the vendor program.

4.2.5 DEFINITION FILE REMARK RECORDS—Remark records shall provide text information such as a version description of a new vehicle control module or vendor interface tool. This information shall identify the components of the programming station used to set the vehicle control parameters. As a minimum, the vendor shall include the following identification items:

a. Host-resident vendor program identification
b. Vendor interface tool identification
c. Vehicle control module identification

The vendor may also add remark records to clarify other sections of the file as needed.

4.2.6 DEFINITION FILE CREATION—The vendors shall supply definition files to each OEM. The vendors shall fill in all fields except for the OEM Name. The OEMs shall fill in the OEM name for each parameter and tattletale record.

4.2.7 DEFINITION FILE MAINTENANCE—The vendor shall update or replace the definition file for each new version of the vendor's program. The distribution of this new information shall be specified by the OEM.

4.3 Parameter File—The parameter file shall contain a set of records which the OEM program passes to the vendor program. These records shall communicate to the vendor program: setup data, controller parameter settings, and requests for tattletale information. The program setup records shall contain special OEM to vendor communications data such as the communication port assignment. The parameter setting records shall provide the information necessary to change the vehicle controller settings. The tattletale records shall request a specific tattletale data item. Each of these records shall contain the data required to modify and verify one setting. Figures 8 through 10 as examples.

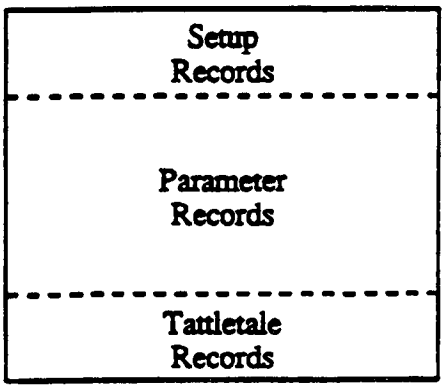


FIGURE 8—PARAMETER FILE STRUCTURE

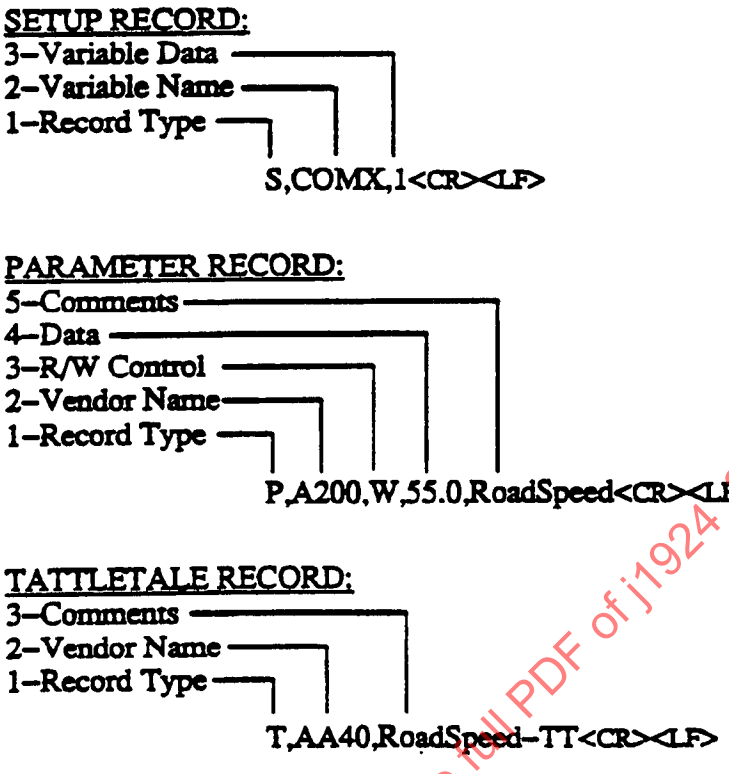


FIGURE 9—PARAMETER FILE RECORD FORMATS

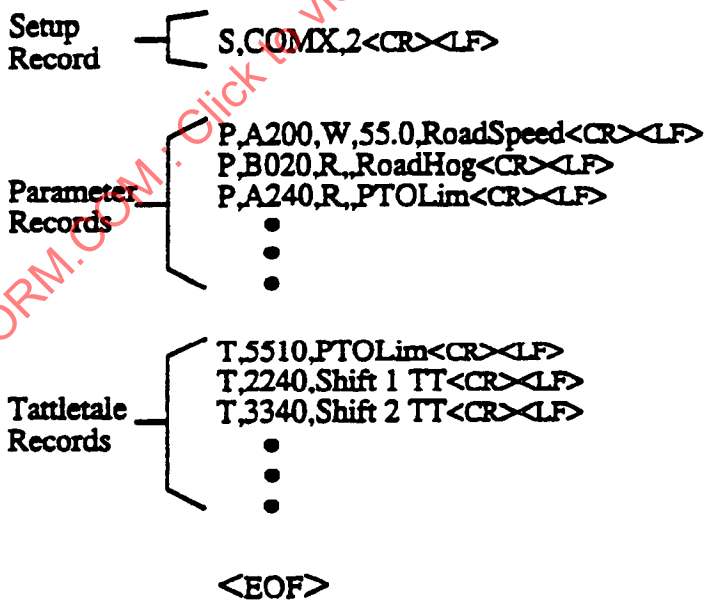


FIGURE 10—SAMPLE PARAMETER FILE

4.3.1 PARAMETER FILE RECORD TYPES—Each of the records shall contain a record type field. The record type field shall be the first field of each record. It shall indicate what the remaining fields of the record will contain. For example, if the record type field contains the ASCII character "P" then the following fields of that record shall contain parameter data. The first character of this field shall be one of the following:

a. "S"—setup variables type record
b. "P"—parameter data type record
c. "T"—tattletale request type record

Other characters may follow the first character but only the first is significant to the OEM and vendor programs.

4.3.2 PARAMETER FILE SETUP RECORDS—Setup records shall provide information for the vendor program. The vendor program shall read the setup records to find such things as the communications port assignment. These records shall contain a field for a variable name and variable data. The variable name indicates the type of data contained in the record. The variable data field shall contain the record data. The following define the types of setup records:

a. Port Assignment
Purpose: to notify the vendor program which communications port of the host computer to use.
Variable Name: COMX
Data: number of the host computer communications port (1–8)
Example: COMX,2

4.3.3 PARAMETER FILE PARAMETER RECORDS—The parameter data records contain the information necessary to set and verify vehicle controller settings. Each record contains the data required to set and verify one parameter. The parameter data record shall consist of the following:

a. Vendor Name—The vendor name identifies a label or address used by the vendor program. This name shall identify for the vendor program the parameter which the OEM program needs to read, write, or verify.
b. Read/Write Control—The Read/Write control field contains an OEM command. The command shall instruct the vendor program to read or write/verify a controller setting. The first character of the read/write control field shall contain one of the following:

"R"—the vendor program shall read the parameter
"W"—the vendor program shall write and verify the parameter

Other characters may follow the first character but only the first is significant to the OEM and vendor programs.
c. Parameter Data—The parameter data field contains the new parameter values. These values represent the customer and chassis specific settings which shall be written to the vehicle controller memory.
d. Comments—The comments field may supply text information. This field will enhance the field readability. For example, it may contain the OEM parameter name.

4.3.4 PARAMETER FILE TATTLETALE RECORDS—The OEM program shall pass a request for tattletale information to the vendor program. The tattletale records in the parameter shall contain the following fields:

a. Vendor Name—The vendor name identifies a label or address used by the vendor program. This name shall identify tattletale information which the vendor program shall read for the OEM program.
b. Comments—The comments field may supply text information. This field is intended to enhance the file readability. For example, it may contain the OEM parameter name.

4.4 **Verification File**—The verification file shall contain a set of records which the vendor program passes to the OEM program. These records shall communicate to the OEM program: setup, parameter verification data, and tattletale information. The program setup records shall contain special vendor to OEM communications data such as error codes for component failures. The parameter verification and tattletale records shall provide feedback information for OEM reliability and warranty reports. Each of these records shall contain the data required to verify one setting. See Figures 11 through 13 as examples.

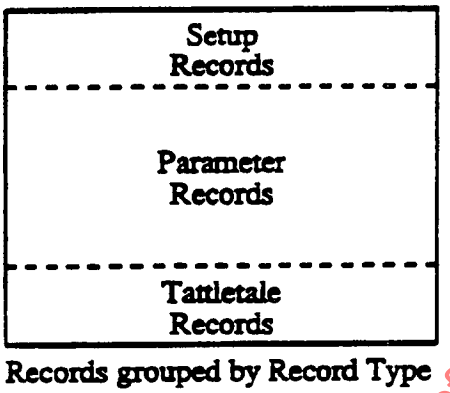


FIGURE 11—VERIFICATION FILE STRUCTURE

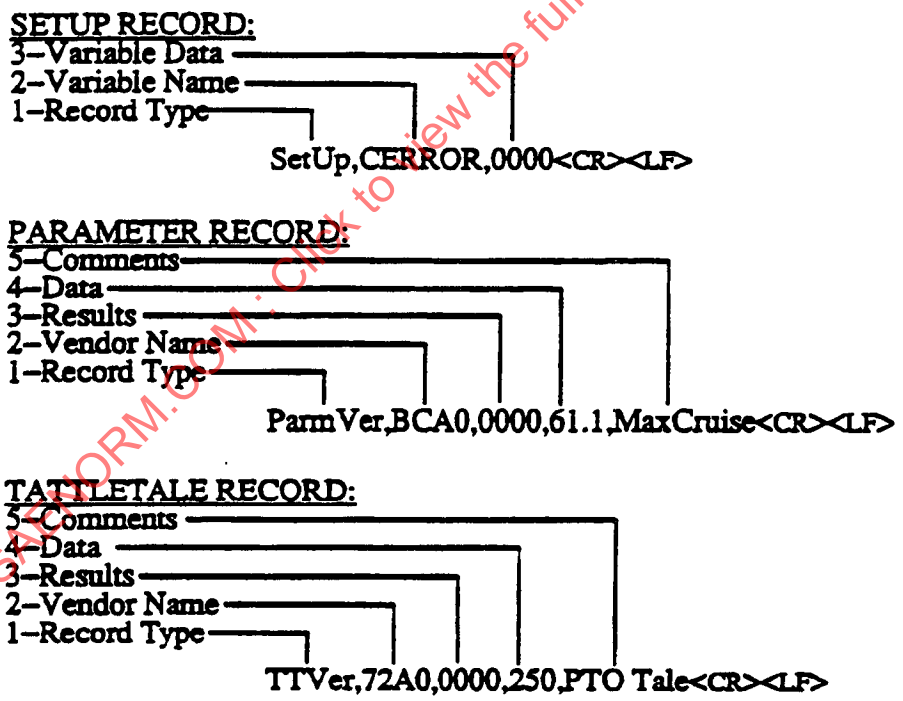


FIGURE 12—VERIFICATION FILE RECORD FORMATS

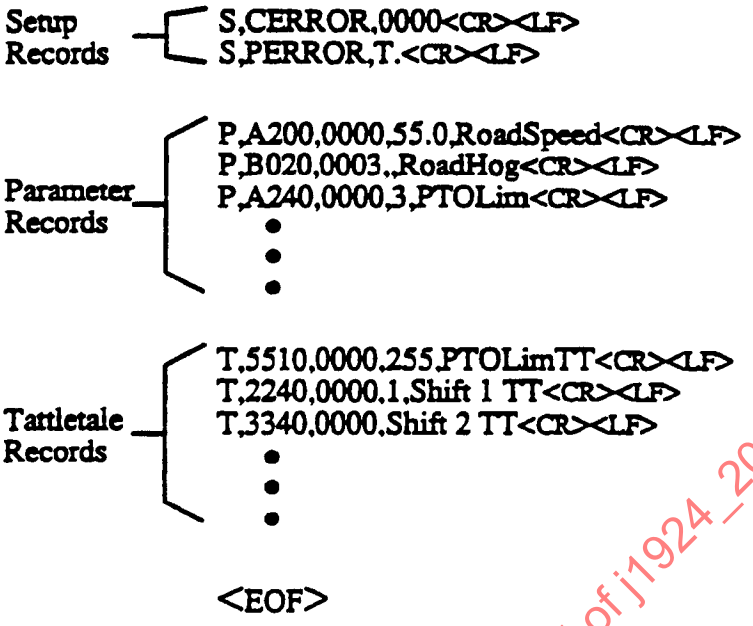


FIGURE 13—SAMPLE VERIFICATION FILE

4.4.1 VERIFICATION FILE RECORD TYPES—Each of the records shall contain a record type field. The record type field shall be the first field of each record. It shall indicate what the remaining fields of the record will contain. For example, if the record type field contains the ASCII character "P," then the following fields of that record shall contain parameter verification data. The first character of this field shall be one of the following:

a. "S"—a setup program variable type record
b. "P"—a parameter verification type record
c. "T"—a tattletale type record

Other characters may follow the first character but only the first is significant to the OEM and vendor programs.

4.4.2 VERIFICATION FILE SETUP RECORDS—Setup records shall provide information for the OEM program. The OEM program shall read the setup records to find such things as program error codes. These records shall contain a field for a variable name and variable data. The variable name indicates the type of data contained in the record. The variable data field shall contain the record data. The following define the types of setup records:

a. Component Error
Purpose: to notify the OEM program that a component of the programming station has failed.
Variable Name: "CERROR"
Variable Data: component error codes as listed in the error code table found in APPENDIX A
Example: CERROR,0000

b. Parameter Error
Purpose: to notify the OEM program that one of the parameter records of the verification file contains an error code.
Variable Name: "PERROR"
Data: shall contain one of the following characters:

"T"—true, there is an error present
"F"—false, there is not an error present

Example: PERROR,T

4.4.3 VERIFICATION FILE PARAMETER RECORDS—The parameter verification data shall provide the information necessary to validate vehicle controller settings. Each record contains the data required to validate one parameter. The verification record shall contain the following fields:

a. Vendor Name—The vendor name identifies a label or address used by the vendor program. This name shall identify to the OEM program that parameter which the vendor program read or wrote/verified.
b. Programming Results—The programming results field shall indicate the success or failure of the programming operation. This field shall contain one of the parameter error codes listed in the error code table. See the error code table in Appendix A.
c. Verification Data—The verification data field contains the new parameter values as stored in the vehicle control module memory. These values represent the actual customer and chassis specific settings which were written to the vehicle controller memory.
d. Comments—The comments field may supply text information. This field is intended to enhance the file readability. For example, it may contain the OEM parameter name.

4.4.4 VERIFICATION FILE TATTLETALE RECORDS—The tattletale records shall provide warranty information for the OEM program. When the OEM requests tattletale information, the vendor program shall return tattletale information. The vendor program shall return the tattletale information in the same manner as other parameters in the verification file. The tattletale records in the verification file shall contain the following fields:

a. Vendor Name—The vendor name identifies a label or address used by the vendor program. This name shall identify to the OEM program that tattletale information which the vendor program read.
b. Read Results—The read results field shall indicate the success or failure of the tattletale read operation. This field shall contain one of the parameter error codes listed in the error code table. The error code table may be found in Appendix A.
c. Tattletale Data—The tattletale data field contains the tattletale values as stored in the vehicle control module. These values represent parameter history data which the OEM may use to process warranty claims.
d. Comments—The comments field may supply text information. This field is intended to enhance the file readability. For example, it may contain the OEM parameter name.

4.4.5 PARAMETER ANOMALIES—The vendor program shall return verification data which will represent the actual data that it stored in the controller's middle memory. Sometimes, the vendor program may return verification data which is different from the parameter data sent by the OEM program. If the vendor returns data different from the parameter data, this may be acceptable. It is acceptable if the difference is due to vendor algorithms or round-off errors. When this is the case, the vendor program shall set the results field to OK. An example: the OEM sends 62.0 and the vendor returns 61.9. If the results field indicates OK, then 61.9 was the actual controller setting and not 62.0.