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

SURFACE VEHICLE RECOMMENDED PRACTICE

SAE J2286

ISSUED FEB97

Issued 1997-02

An American National Standard

VENDOR COMPONENT PROGRAM DATA FILE INTERFACE FOR OEM ASSEMBLY OPERATIONS

Foreword—SAE J2214 defines the requirements for SAE J2286. SAE J2286 describes the file-based interface between the OEM Shop Floor Program and the Vendor Component Program, superseding the interface described in SAE J1924. This document is intended to document existing systems only. Recommended Practices are under development for the windows based operating system.

SAE J2286 content was developed by tailoring the Interface Design Document Data Item Description, DI-MCCR-80027, and Data Base Design Document Data Item Description, DI-MCCR-800028, that is used with DOD-STD-2167A.

TABLE OF CONTENTS

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

1. Scope

1.1 Identification—This interface document SAE J2286 revises the requirements for file formats as described in SAE J1924. This document describes Interface 1 (I/F 1) in SAE J2214. This document does not imply the use of a specific hardware interface, but may be used with other hardware interfaces such as SAE J1939. The requirements of SAE J2286 supersede the requirements defined by SAE J1924.

1.2 Introduction—SAE J2214 establishes the requirements for Interface 1 (I/F 1), as a replacement of the file-based interface described by SAE J1924, as shown by Figure 1. Interface 1 (I/F) is a bi-directional link between the OEM Shop Floor Program (CSCI 1) and the Vendor Component Program (CSCI 2). Using I/F 1, the OEM Shop Floor Program communicates the desired parameters and programming limits for an assembly job to the Vendor Component Program (VCP). In response, the VCP returns programming results to the OEM Shop Floor Program (CSCI 1).

I/F 1 maintains the file structures described in SAE J1924 as closely as possible. The parameter file setup record was changed to reflect the common means communication provided by CSCI 3, the communication Device Driver. A new record type was added to better define how data files, external to the parameter file, would be used. This clarifies SAE J1924 by clearly defining whether the parameter value string 123.123 refers to a numeric value or an MS-DOS™ filename.

Besides correcting errors, other changes to SAE 1924 include:

a. Deletion of tattletale records
b. Better descriptions of fields and records
c. An additional field in a Definition File record
d. Directory structure and file naming conventions supporting future growth
e. VCP command line content and file directory structure

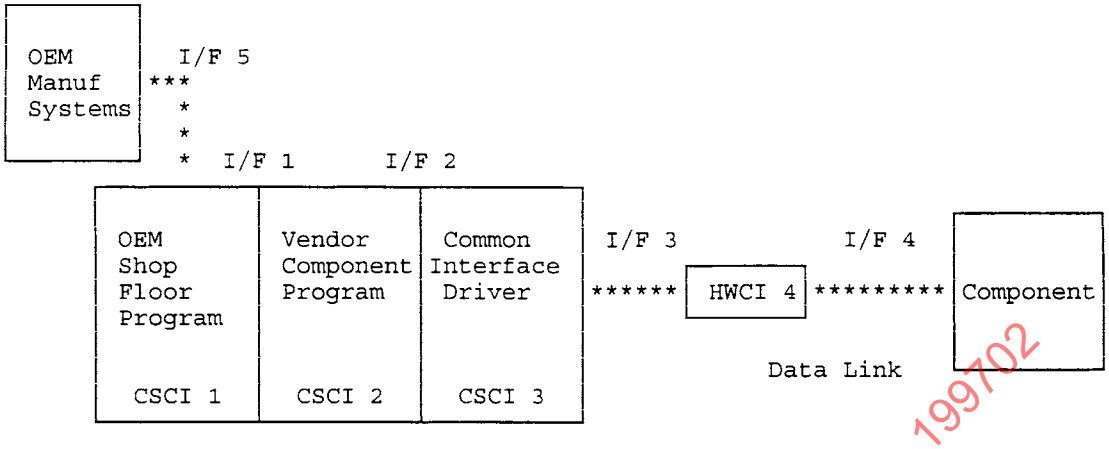

FIGURE 1—VEPS COMPUTER PROGRAMS AND INTERFACES

2. References

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

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

SAE J1683—MS-DOS™ Interface for SAE J1708 Communications
SAE J1708—Serial Data Communications Between Microcomputer Systems in Heavy-Duty Vehicle Applications
SAE J1924—OEM/Vendor's Interface Specification for Vehicle Electronic Programming Station
SAE J2214—Vehicle Electronic Programming Station (VEPS) System Specification for Programming SAE

2.1.2 DOD STD 2167A RELATED DOCUMENTS—Available from DODSSP, Standardization Document Order Desk, 700 Robbins Avenue, Bldg., 4D, Philadelphia, PA 19111-5094.

DOD-STD-2167A—Defense Software Development
DI-MCCR-80026—Interface Requirements Specification
DI-MCCR-80027—Interface Design Document
DI-MCCR-80028—Data Base Design Document

2.1.3 AIAG PUBLICATIONS—Available from AIAG, 26200 Lahser Road, Southfield, MI 48034.

3. Abbreviations and Acronyms

ASCII—American Standard Code for Information Interchange
CSCI—Computer Software Configuration Item
EDI—Electronic Data Interchange
HWCI—Hardware Configuration Item
I/F—Interface
OEM—Original Equipment Manufacturer
SAE—Society of Automotive Engineers, Inc.
VCP—Vendor Component Program
VDT—Video Display Terminal
VEPS—Vehicle Electronics Programming Station

3.1 Trademark Acknowledgments

MS-DOS™ is a trademark of Microsoft, Inc.

4. Requirements

4.1 Interface Summary—Table 1 summarizes I/F 1, a bi-directional, file-based interface between CSCI 1 (the OEM Shop Floor Program) and CSCI 2 (the Vendor Component Program or VCP). The *parameter file* is passed from CSCI 1 to CSCI 2 which effects its instructions. CSCI 2 returns the results of the instructions to CSCI 1 using the *verification file*. The *definition file* documents the parameters supported by CSCI 2; customized definition files may provide CSCI 2 with additional information regarding OEM preferences for his vehicles.

Table 1 defines the interface file types and general definitions.

TABLE 1—INTERFACE FILE TYPES AND INITIATION CONDITIONS

FILE [.EXT] *(Information Flow)*	CONTENT	INITIATION CONDITIONS	RECIPIENT RESPONSE
parameter [.par] *(CSCI 1 to CSCI 2)*	Parameter Programming Values and communications setup	Evocation of the VCP by the OEM Shop Floor Program	The VCP will program the component to the desired parameter settings
Definitions [.def] *(Vendor to OEM CSCI 2 to CSCI 1)*	File naming defaults, parameter data type, scaling and limits	Released by Vendor to OEM with the modified VCP	Use of a modified definition file is solely at the discretion of the vendor
Verification [.ver] *(CSCI 2 to CSCI 1)*	Resulting parameter values and programming status (errors)	Generated by VCP at completion of component programming	Defined by OEM Shop Floor Program
Remarks [.rem] *(Vendor to OEM)*	VCP & Component Notes	Vendor release of VCP to OEM	Read.ME
External Data [.dat] *(CSCI 1 to CSCI 2)*	Programming values not practical for parameter file format	Evocation of the VCP by the OEM Shop Floor Program	The VCP will program the component to the desired parameter settings

4.2 General File Characteristics—Each interface file is an ASCII text file with multiple records. The file structure is similar to Comma Separated Value (CSV) format. The comma (,) is used as the field delimiter and separates each field in a record. An end of line marker <CR><LF> terminates each record, and an end of file marker terminates the file.

4.2.1 TEXT FILE FORMAT—Record fields may contain any printable (0x20 to 0x7E) ASCII characters. A field may contain a comma (0x2C) if the entire comment is enclosed with a pair of double quotes (0x22). Additionally, the use of the double quote enclosing the entire field is not required but is allowed in fields without a comma in the field.

An example of a comment field using commas and double quotes are as follows:

P,B020,B,501,"Tire Size,Revs/Mile"<CR><LF>

NOTE—Entire comment field enclosed in " because of comma in field.

The last field of a record may be delimited by one or more of the ASCII space characters (0x20). If a comment is not included, then a comma may or may not follow the data field. Fields that do not apply for a record may contain ASCII space (0x20) or TAB (0x09) characters. Records shall be terminated by both a carriage return (0x0D) and a line feed (0x0A). A control-Z (0x1A) shall mark the end of each field. All records with only white space or blank lines shall be ignored.

4.2.2 CASE SENSITIVITY—The files may contain either upper or lower case alphabetic characters. However, the OEM and vendor programs are not required to make "case sensitive" tests. The programs shall interpret the upper and lower case alphabetic characters contained in record fields as being the same. For example, the programs shall interpret "WxYz" as equivalent to "wXyZ". Both values are equivalent to "WXYZ".

This requirement may be relaxed only when the goal is to program freeform text as the value of the parameter—in such cases the parameter value field may contain both upper and lower case characters that need not be the equivalent of an all uppercase character string. This is described in 4.2.3.4.

4.2.3 PARAMETER VALUE DATA TYPE CONVENTIONS—The parameter value data types shall conform to the following paragraphs.

4.2.3.1 *Numeric Data Types*—Parameters with numeric data types shall be composed as:

<digit> [>digit>*] [.] <digit>*]*The character <digit>* implies zero or more occurrences of <digit>*.

where digit is an ASCII numeral: {0,1,2,3,4,5,6,7,8, or 9}

4.2.3.2 *Enumerated Data Types*—A range of integers, [0:N-1], is the preferred method for encoding an enumerated data type of N elements. To retain past usage, sets of ASCII strings such as {RED, BLUE, BLACK, WHITE} may also be used. The ASCII strings providing values for enumerated data types may contain either upper or lower case alphabetic characters—lower case alphabetic characters shall be interpreted the same as upper case alphabetic characters. The advantage of a range over strings is that integer ranges have upper and lower values that can be used for simple rationality checks. (See 4.4.)

4.2.3.3 *Boolean Data Types*—Boolean data types are a special case of enumerated date types. Parameters conveying Boolean data utilize the ranges [0:1] or [N:Y] as data values. The use of the range [0:1] is preferred.

4.2.3.4 *Textual (ASCII) Types*—Textual data may also be used to supply parameter values. Text containing lower case alphabetic characters, the space character (0x20), or the field delimiter (,) shall be enclosed by the double-quote ("), ASCII 0x22 per 4.2.1.

4.2.3.5 *External File Data Types*—A valid MS-DOS™ file including an optional path may be provided in a parameter value file to identify the file containing the data to be used for the parameter value. External data files may be used only in conjunction with record type "F". See 4.4.4, 4.5.3, and 4.6.4 for more information regarding the use of external data files.

4.2.4 FILE NAMES—As a default, all vendor supplied files shall conform to the following naming conventions: except for external data files, the file name shall encode the vendor's company name, the product family supported by the VCP, the VCP version number, and the type of file. The first three letters of the file name shall contain the vendor's company mnemonic. The next two characters shall identify the product supported by the VCP. The last three characters shall encode a version number of the VCP. External data files may have any name the Vendor determines to be appropriate, as long as the file names are unique from the vendor-product-version scheme.

The file extension shall identify the type of file. The following file extensions shall be used for the files indicated:

a. Executable file = .EXE
b. Verification file = .VER
c. Definition file = .DEF
d. Parameter file = .PAR
e. Remarks file = .REM
f. External Data File = .DAT

In all cases, file extensions can be assumed if not explicitly defined.

4.2.4.1 *File Naming Examples*—As described in 4.3, file names no longer need to be unique for each VCP that an OEM may receive from their vendors. The following examples illustrate the previous convention.

a. EATTC120.DEF Definition file version 1.20 for Eaton Corp. ABS/TC Product VCP
b. CATE100.VER Verification file version 1.00 for Caterpillar Electronic Engines
c. DDCD3210.EXE Executable file version 2.10 for Detroit Diesel Product “D3”
d. CECCL302.REM Remarks file version 3.02 for Cummins engine VCP
e. CALWR6SP.DAT External data file for a wide ratio six forward speed transmission

4.3 VCP Command Line Content and File Directory Structure—The OEM shall provide a unique sub-directory for each VCP installed on the Shop Floor PC. This directory shall contain all vendor provided external data files. Data file names are supplied as the value of a parameter in a parameter file record type F. The invocation of the VCP (CSCI 2) contains the following elements:

[d.] [exe-path[vcpname [.exe]

[-p [par-path] [parameter file name]
[-v [ver-path] [verification file name]
[-d [def-path] [definition file name]
[vendor defined options]

-p, -v, -d define switches modifying the default file and directory information. One or more ASCII spaces (0x20) must be included after the switch. Table 2 explains the content and default values of the invocation elements. As required by MS-DOS™, the invocation string shall not exceed 128 bytes.

TABLE 2—VCP (CSCI 2) INVOCATION CONTENT

d:	Optional device specification of the VCP executable file name path. The default value is the current device.
exe-path	Directory path information for the VCP executable file name. The path given may be a full or partial path specification including DOS file path specifiers such as “.” and “..”.
vcpname	Executable file name of the VCP program to be run.
-p[par-path] parameter	Optional path and file name for the parameter file content. The default parameter file shall be the same name as the executable BCP with the extension of PAR. The file name default path shall be the same as *exe-path*.
-v[ver-path] verification	Optional path and file name for the verification file. The default file name and path shall be the same file name and path given for the parameter file—the default file extension shall be .VER.
-def[def-path] definition	Optional path and file name for the definition file. The default file name and path shall be the same file name and path given for the executable file—the default file extension shall be .DEF.
[vendor define options]	Vendors may define other optional switches as needed for their own use. OEMs are not required to support vendor supplied options in VCP invocations.

4.3.1 INVOCATION PRECEDENCE—A precedence on names and path locations for all file types is established. The precedence in order of priority is:

1. Any file name and path supplied on the invocation line of the VCP will have precedence over names and paths specified in the definition file (see 4.4 for details).
2. With the absence of one or more file name switches on the invocation line, any corresponding file names and paths specified in the definition file take precedence. (see 4.4 for details).
3. With the absence of file name switches on the invocation line and specifications in the definition file, the default names and paths as defined in Table 2 shall be used.

4.4 Definition File—The definition file describes the parameters supported by the VCP for the OEM. The definition file also describes the components for whch the VCP is intended and the default file naming conventions for the VCP.

4.4.1 DEFINITION FILE RECORD TYPES—Definition files are comprised of four record types: Setup Records, Parameter Records, External Data File Records, and Remarks Records. Together these records describe default file naming conventions for the interface file names and the parameters which may be programmed for a component controller. Setup records may be preceded by only remarks records. Parameter and external data file records may be interspersed in the definition file, after all the setup records.

The first field of each record shall be the record type field. 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 (no leading white space is allowed):

“S” - setup definition type record
“P” - parameter definition type record
“F” - external data file description record
“R” - remarks type record

Figure 2 shows an example definition file. Subsequent sections define the record types in detail.


```
REM,This definition file describes the parameters for <CR><LF>
REM,the product name(s), the product name version(s). <CR><LF>
REM,<CR><LF>
SETUP,EXEFILE,CECCL234.EXE,   Each setup filename uses the <CR><LF>
SETUP,PARFILE,CECCL234.PAR,   Vendor-Product-Version convention <CR><LF>
SETUP,VERFILE,CECCL234.VER,   VVVPPNNN.EXT <CR><LF>
SETUP,REMFILE,CECCL234.REM,<CR><LF>
REM,<CR><LF>
REM,Parameter and File Records follow SETUP records <CR><LF>
REM,<CR><LF>
PARM,,VSL,BE00,BOTH,MPH,65,10,90,5,Road Speed Limit <CR><LF>
PARM,,HCL,A0A0,READ,MPH,5,2,10,1,High Cruise Set Speed <CR><LF>
PARM,,,VPULSES,WRITE,RPM,25000,10000,50000,1,Speedo Calibration <CR><LF>
FILE,,,SHIFTCAL,BOTH,,CALWR6SPD.DAT,,,,Default Shift Calibration <CR><LF>
...
<EOF>
```

FIGURE 2—DEFINITION FILE EXAMPLE

4.4.2 DEFINITION FILE SETUP RECORD—Four setup records documenting the default file naming conventions between the OEM Shop Floor Program and the Vendor Component Program shall appear in the definition file. Setup records are comprised of four fields which are described as follows. The first three fields shall be followed by the standard field delimiting character, ",".

4.4.2.1 Record Type–Field 1—The record type field shall contain an ASCII "S". No leading white space is allowed.

4.4.2.2 Variable Name–Field 2—The four required variable names are:

a. EXEFILE [Executable file name of the Vendor Component Program]
b. PARFILE [File name of the Parameter File]
c. VERFILE [File name of the Verification File]
d. REMFILE [File name of the Remarks File]

The EXEFILE parameter defines the executable program file name for the Vendor Component Program.

The PARFILE parameter defines the default parameter file name to be used by the Vendor Component Program to obtain the parameter values to be programmed.

The VERFILE parameter defines the default verification file name to be used by the Vendor Component Program to provide the results of its programming procedure.

The REMFILE parameter defines the default remarks file name provided by the vendor documenting the definition file and Vendor Component Program.

4.4.2.3 File Name–Field 3—The variable data contains the file name expected by the VCP for the setup variable given in field 2. Full pathnames are allowed in the file definitions.

4.4.2.4 Comment–Field 4—ASCII text may follow the delimiter for field 3 to comment the record—when comments are provided the comma should be used to delimit field 3.

4.4.3 DEFINITION FILE PARAMETER RECORD—The parameter records contain the information necessary to translate and describe vehicle component controller settings. Each parameter record defines one calibration setting for the component the VCP supports.

4.4.3.1 Record Type–File 1—The record type for definition file parameter records is an ASCII "P". No leading white space is allowed.

4.4.3.2 OEM Name–Field 2—The OEM name reserves space for the OEM to provide a unique label for the parameter. The vendor may provide zero or more ASCII spaces (0x20) as the value of this parameter.

4.4.3.3 EDI Label–Field 3—The EDI label identifies the EDI label defined by the vendor for the parameter. The use of labels promulgated as a standard by the Automotive Industries Action Group (AIAG), Truck Action Group (TAG) is preferred to proprietary EDI labels defined by the Vendor.

4.4.3.4 Vendor Name–Field 4—The vendor name identifies a label or address used by the VCP to distinguish the parameter from other parameters. The values provided in Field 4 of the definition file are the only values that the VCP must recognize in Field 2 of the parameter file.

4.4.3.5 *Parameter Access–Field 5*—Parameter access codes the accessibility of the parameter by the OEM program. The first character of the fifth field shall be one of the following:

"R"- the parameter value can only be read and shall be displayed in the verification file
"W"- the parameter value can only be written and shall not be displayed in the verification file (the programming status will reflect the result of this programming request)
"B"- the parameter may both be written and read and the resulting data displayed in the verification file

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

4.4.3.6 *Units of Measure–Field 6*—The units of measure for the parameter are documented in Field 6. When no units of measure apply to the parameter, zero or more ASCII space characters will be provided as the content of the units of measure field. Some example units of measure are:

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

4.4.3.7 *Default Value–Field 7*—The default value field contains the value of the parameter on the component as supplied by the vendor before OEM programming. A typical value may be supplied for the defaults when the VCP supports multiple products with different defaults. In cases where a typical value is not appropriate, zero or more ASCII space (0x20) characters should be used. The remarks file should document what the default values are for each case when a single default value does not exist.

4.4.3.8 *Lower Limit–Field 8*—The lower limit field contains the lower value to which the parameter may be set.

4.4.3.9 *Upper Limit–Field 9*—The upper limit field contains the highest value to which the parameter may be set.

4.4.3.10 *Scaling Increment–Field 10*—The scaling increment defines the resolution to which the parameter may be set. For example, a parameter with a lower limit of 1 and a scaling increment of 0.25, may be programmed with one value from the following series: 1, 1.25, 1.5, 1.75, 2.0, .. upper limit.

4.4.3.11 *Comment–Field 11*—ASCII text may follow the delimiter for field 10 to comment the data content of the record. When a comment is provided, the comma field delimiter shall be used for field 10.

4.4.4 DEFINITION FILE EXTERNAL DATA FILE RECORD—The definition file external data file record is a special case of the definition file parameter record. This record identifies the file name with optional path of the external data file that is to be programmed. The parameter record fields that do not apply are left blank. The definition file external data file record identifies the file that contains those parameters whose programming values are too long to be practically used in the parameter file format and should be provided the VCP as an external data file. In general, external data files should be used whenever the parameter file parameter record would exceed 132 characters. Parameters supported by files should have specific instruction on selecting the vendor provided file for its value. A parameter that is supported by a file may provide the calibration for a group of calibration settings.

4.4.4.1 *Record Type–Field 1*—The record type for definition file file records is an ASCII "F". No leading white space is allowed.

4.4.4.2 *OEM Name–Field 2*—The OEM name identifies the parameter which may be changed or verified.

4.4.4.3 *EDI Label–Field 3*—The EDI Label identifies the EDI label defined by the vendor for the file. The use of labels promulgated as a standard by the Automotive Industries Action Group (AIAG), Truck Action Group (TAG) is preferred to proprietary EDI labels defined by the Vendor.

4.4.4.4 *Vendor Name–Field 4*—The vendor name identifies a label or address used by the VCP to distinguish the file from other files. The values provided in Field 4 of the definition file are the only values that the VCP must recognize in Field 2 of the parameter file.

4.4.4.5 *File Access–Field 5*—File access codes the accessibility of the parameter to the OEM Shop Floor Program. The first character of the fifth field shall be one of the following:

"R" - the parameter values in the file can only be read
"W" - the parameter values in the file can only be written
"B" - the parameter values in the file may both be written and read

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

4.4.4.6 *Units of Measure–Field 6*—The units of measure for the file record will be zero or more ASCII spaces (0x20).

4.4.4.7 *Default Value–Field 7*—The default value field contains the name of the file provided with the VCP that contains the default values that should be programmed. In cases where a default file is not appropriate, zero or more ASCII space (0x20) characters should be used. The remarks file should document what the default values are for each case when a single default value does not exist.

4.4.4.8 *Lower Limit–Field 8*—The lower limit field for the file record will be zero or more ASCII spaces (0x20).

4.4.4.9 *Upper Limit–Field 9*—The upper limit field for the file record will be zero or more ASCII spaces (0x20).

4.4.4.10 *Scaling Increment–Field 10*—The scaling increment for the file record will be zero or more ASCII spaces (0x20).

4.4.4.11 *Comment–Field 11*—ASCII text may follow the delimiter for field 10 to comment the data content of the record. When a comment is provided, the comma field delimiter shall be used for field 10.

4.4.5 DEFINITION FILE REMARKS RECORD—Within the definition file, remarks records shall provide textual description of the products the VCP supports. Remarks records should clearly describe both the name and the version description of the products supported by the VCP. The vendor may also add remark records to clarify other sections of the file needed. Remark records shall begin with an ASCII "R", fulfilling the record type requirement. A comma will be used to separate the record type "R" and the comment itself.

4.4.6 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 OEM's shall fill in the OEM name for each parameter.

4.4.7 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.

4.5 Parameter File—The parameter file for SAE J2214 I/F 1 contains records that the OEM Shop Floor Program provides to the Vendor Component Program. The parameter file communicates setup data and desired component controller parameter settings to the VCP. The parameter file setup record identifies the software interrupt for the communications device driver defined by the applicable CSCI 3 SAE Truck and Bus Recommended Practice (e.g., SAE J1683 MS-DOS™ Interface for SAE J1708). Other records provide the information necessary to change the component controller settings. Each parameter file parameter record and file record contains the data required to modify and verify one setting.

The first field of each parameter file record is the record type field. The record type field 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 (no leading white space is allowed):

"S" - setup variables type record
"P" - parameter data type record
"F" - file description record
"R" - remarks record

Any parameter programming errors, invalid records, unauthorized access, or missing external data files shall be noted in the verification file as defined by Appendix A–Error Code Table.

Figure 3 displays an example parameter file. The parameter file setup must precede any parameter or file records. Subsequent sections define the record types in detail.

```
REM,Only Remarks records may precede the Setup Record<CR><LF>
S,COMSWINT,0x64,The J1683 Software Interrupt is set at 64 HEX<CR><LF>
REM,The setup record COMSWINT must precede the parameter and<CR><LF>
REM,file records<CR><LF>
P,A200,B,55.0,Road Speed Limit<CR><LF>
P,B020,B,501,Tire Size in Revs/Mile<CR><LF>
P,A240,R,,PTOLIM<CR><LF>
P,PASSWORD1,W,0000,Customer Password<CR><LF>
F,CALID,B,123.dat,123_ver.dat,Close Ratio/No Retarder/Std IO<CR><LF>
...
<EOF>
```

FIGURE 3—SAMPLE PARAMETER FILE

4.5.1 PARAMETER FILE SETUP RECORD—As shown in Figure 3, there is one required parameter file setup record containing three fields: a record type, variable name, and variable data. The setup variable COMSWINT replaces the SAE J1924 communication setup parameter, COMX. The VCP shall read the setup record and obtain the interrupt vector for the communications device driver facility provided by CSCI 3. Vendors may define additional parameter file setup records. Additional parameter file setup records may also be defined for future CSCI 3 solutions that use a non-TSR solution.

4.5.1.1 Record Type–Field 1—The record type shall contain an ASCII "S".

4.5.1.2 Variable Name–Field 2—The variable name COMSWINT will be used.

4.5.1.3 *Variable Data–Field 3*—The variable data for COMSWINT shall contain the software interrupt for the SAE J1708 driver (CSCI 3). The interrupt vector will be coded as four ASCII characters, 0xYZ, where YZ is the hexadecimal number of the software interrupt for the OEM-supplied SAE J1683 communications package. Non-TSR driven CSCI 3 solutions will set this field to XXXX.

4.5.1.4 *Comment–Field 4*—ASCII text may follow the delimiter for field 3 to comment the data content of the record. When a comment is provided, the comma field delimiter must be used for field 3.

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

4.5.2.1 *Record Type–Field 1*—The record type code for all parameter records is an ASCII "P". No leading white space is allowed.

4.5.2.2 *Vendor Name–Field 2*—The vendor name identifies a label or address used by the VCP. This name shall identify the parameter that the VCP shall read, write, or verify. The only Vendor Names the VCP must recognize are contained in the Vendor Name field of the definition file for the VCP version invoked by the OEM Shop Floor Program (CSCI 1).

4.5.2.3 *Access Control–Field 3*—The Access control field directs the VCP to read, write/verify, or read and write/verify a controller setting. The first character of the access control field shall contain one of the following:

"R" - the parameter value can only be read and shall be displayed in the verification file
"W" - the parameter value can only be written and shall not be displayed in the verification file (the programing status will reflect the result of this programming request)
"B" - the parameter may both be written and read and the resulting data displayed in the verification file

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

4.5.2.4 *Parameter Data–Field 4*—The parameter data field contains the new parameter value. This value represents a customer's or chassis specific "B" or "W" settings which should be written to the component controller memory.

4.5.2.5 *Comments–Field 5*—The comments field may be supplied to document the parameter setting. When provided, this field delimiter for field 4 must be a comma. VCP will ignore all comments—there are no processing requirements for comments in the parameter file on the VCP.

4.5.3 PARAMETER FILE EXTERNAL DATA FILE RECORD—The external data file record is a special case of the parameter file parameter record. The parameter record fields that do not apply are left blank. The external data file record identifies those parameters whose programming values are too long to be practically used in the parameter file parameter record format and should be provided to the VCP as an external data file. In general, external data files should be used whenever the parameter file parameter record would exceed 132 characters. Parameters supported by files should have specific instruction on selecting the vendor provided file for its value. A parameter that is supported by a file may provide the calibration for a group of calibration settings.

4.5.3.1 *Record Type–Field 1*—The record type code for all file records is an ASCII "F". No leading white space is allowed.

4.5.3.2 *Vendor Name–Field 2*—The vendor name identifies a label or address used by the VCP. This name shall identify for the VCP the parameter that the OEM Shop Floor Program need to read, write, or verify. The only Vendor Names the VCP must recognize are contained in the Vendor Name field of the definition file and are defined with a definition record type of "F".

4.5.3.3 *Access Control Field–Field 3*—The access control field contains an OEM command. Access control commands instruct the VCP to read, or write/verify a controller setting. The first character of the access control field shall contain one of the following:

"R" - the parameter value can only be read and shall be displayed in the verification file
"W" - the parameter value can only be written and shall not be displayed in the verification file (the programming status will reflect the result of this programming request)
"B" - the parameter may both be written and read and the resulting data displayed in the verification file

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

4.5.3.4 *Input File Name–Field 4*—The input file name field identifies the name of the file containing the desired input data. The file name may be preceded by a full or partial path name. When the access control field is "W" or "B", the default path shall be the same as for the VCP executable file. When an "R", the default path shall be the same as for the verification file.

4.5.3.5 *Output File Name–Field 5*—The output file name field identifies the name of the file that will contain the desired output data when the access control field is a "B" or "B". The file name may be preceded by a full or partial path name. The default path shall be the same as for the verification file.

4.5.3.6 *Comments–Field 6*—The comments field may be supplied to document the parameter setting. When provided, this field delimiter for field 4 must be a comma.

4.5.4 PARAMETER FILE REMARKS RECORDS—Within the parameter file, remarks may describe the intended use of an individual parameter file and further explain parameter settings. Remark records shall begin with an ASCII "R", fulfilling the record type requirement. A comma will be used to separate the record type "R" and the comment itself.

4.6 Verification File—The verification file shall contain a set of records that the VCP provides to the OEM Shop Floor Program. These records shall communicate setup data and parameter verification data. The setup data records identify component errors and confirm or deny the existence of parameter errors. Verification records for individual parameters provide feedback information that may be used to analyze programming problems. Figure 4 displays an example verification file. Subsequent sections define the record types in detail.

```
S,CERROR,1003,CERROR is always the first record<CR><LF>
S,PERROR,1,PERROR is always the second record<CR><LF>
S,INVOCATION,ETNTC100.EXE -d ETNTC120.DEF<CR><LF>
P,A200,0000,55.0,<CR><LF>
P,B020,0003,,Remarks are at the sole discretion of the vendor<CR><LF>
P,A240,0000,3,There are no remarks records in the verification file.<CR><LF>
P,A002,0002,0,Example of an error message<CR><LF>
F,SHIFTCAL,1003,SHIFT_V,<CR><LF>
R,Remarks record for verification file<CR><LF>
...
<EOF>
```

FIGURE 4—SAMPLE VERIFICATION FILE

4.6.1 VERIFICATION FILE RECORD TYPES—The first field of each record in the verification file shall be the record type field. The record type field indicates what the remaining fields of the record can 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 (no leading white space is allowed):

"S" - a setup program variable type record
"P" - a parameter verification type record
"F" - a file description record
"R" - a remarks record

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

4.6.2 VERIFICATION FILE SETUP RECORD—The OEM Shop Floor Program shall read the setup records to determine the overall success of the programming operation. Verification file setup records contain three fields: record type, variable name, and variable data. The variable name indicates the type of data contained in the record. The variable data field shall contain data required by the variable name. Verification file setup records are explained in the following paragraphs:

4.6.2.1 Record Type–Field 1—The record type shall contain an ASCII "S". No leading white space is allowed.

4.6.2.2 Variable Name/Variable Data–Fields 2 and 3—Table 3 illustrates both verification file setup records:

TABLE 3—VERIFICATION FILE SETUP RECORD VARIABLES AND CONTENT

CERROR	Always the first record in the verification file. A value of 1000 indicates there were no process errors. Values for specific errors are noted in Appendix A.
PERROR	Always the second record in the verification file. A value of F indicates that all parameters were successfully programmed. A value of T indicates that at least one parameter had a programming problem. An error code is provided for each parameter in error.
INVOCATION	Always the third record in the verification file. This is the returned invocation line that was used to start the VCP.

4.6.2.3 Comments—Field 4—The comments field may be supplied to document the error value. When provided, the field delimiter for field 3 must be a comma. The OEM Shop Floor Program will ignore all comments—there are no processing requirements for comments in the verification file on the OEM Shop Floor Program.

4.6.3 VERIFICATION FILE PARAMETER RECORD—Verification file parameter records provide the resulting component controller settings. Each record contains the data resulting from the corresponding request in the parameter file used by the VCP. The verification file parameter records shall contain the following five fields:

4.6.3.1 Record Type–Field 1—The record type field shall be an ASCII "P". No leading white space is allowed.

4.6.3.2 Vendor Name–Field 2—The vendor name identifies a label or address used by the VCP. This name shall identify the parameter that the VCP read or wrote and verified.

4.6.3.3 Programming Results–Field 3—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 given in Appendix A, Figure A1.

4.6.3.4 *Verification Data–Field 4*—The verification data field contains the actual value stored in the components controller module memory for the parameter given in field 2 regardless of the value of the access control field (Read, Write, or Both).

4.6.3.4.1 Resulting Parameter Value Anomalies—The VCP shall return verification data that represents the actual data stored in the component memory. Sometimes, the VCP may return a value which differs from the parameter data sent by the OEM Shop Floor Program–this may be acceptable. When the difference is due to vendor algorithms or scaling round-off errors, the VCP shall set the results field to 0000. For example, suppose the OEM Shop Floor Program sends 62.0 and the vendor returns 61.9. If the results field contains 0000, then 61.9 is the actual parameter setting and not 62.0.

4.6.3.5 *Comments–Field 5*—When field 3 contains 0006, the comments field shall contain an ASCII description of the ERROR discovered by the VCP. Any other content of the comments field is solely up to the vendor.

4.6.4 VERIFICATION FILE EXTERNAL DATA FILE RECORD—The verification file external data file record is a special case of the verification file parameter record. The verification file external data file record identifies those parameters whose programming values were too long to be practically used in the verification file parameter record format and were provided by the VCP as an external data file.

4.6.4.1 *Record Type–Field 1*—The record type field shall be an ASCII "F". No leading white space is allowed.

4.6.4.2 *Vendor Name–Field 2*—The vendor name identifies a label or address used by the VCP. This name shall identify the file that the VCP read or written/verified.

4.6.4.3 *Programming Results–Field 3*—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.

4.6.4.4 *Verification Data–Field 4*—The verification data field contains the name of the external file containing the data stored in the component's controller module memory for the vendor name given in field 2. The file shall be the name and use the path as provided in the output file name field in the associated parameter file. The file shall be written when the value of the Access Control Field in the parameter file is "Read" or "Both".

4.6.4.5 *Comments–Field 5*—When field 3 contains 0006, the comments field shall contain an ASCII description of the ERROR discovered by the VCP. Any other content of the comments field is solely up to the vendor.

4.6.5 VERIFICATION FILE REMARKS RECORD—There are no processing requirements for the Remarks Record in the verification file on the OEM Shop Floor Program. The contents of these optional records shall be at the discretion of the vendor, but may include items such as supplementary troubleshooting data or runtime datestamp.

4.7 Remarks File—The remarks file is a separate file used to provide textural information for clarification, and instructions regarding the use of the VCP, the setting of individual parameters and descriptions of file parameter use. No format will be specified for this file.