
**Tractors and machinery for
agriculture and forestry — Serial
control and communications data
network —**

**Part 7:
Implement messages application layer**

*Tracteurs et matériels agricoles et forestiers — Réseaux de
commande et de communication de données en série —*

Partie 7: Couche d'application de base

STANDARDSISO.COM : Click to view the full PDF of ISO 11783-7:2022



STANDARDSISO.COM : Click to view the full PDF of ISO 11783-7:2022



COPYRIGHT PROTECTED DOCUMENT

© ISO 2022

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

	Page
Foreword.....	v
Introduction.....	vi
1 Scope.....	1
2 Normative references.....	1
3 Terms and definitions.....	1
4 General requirements and recommendations.....	2
4.1 General.....	2
4.2 ISO 11783-7 electronic database.....	2
5 Technical requirements.....	3
5.1 Overview.....	3
5.2 General guidelines.....	3
5.2.1 Signal characterization.....	3
5.2.2 Message format.....	3
5.2.3 ISO Latin 1 character set.....	4
5.2.4 Parameter ranges.....	5
5.2.5 Assignment of ranges to new parameters.....	5
5.2.6 Adding parameters to groups.....	6
5.2.7 Transmission repetition rates — Update rates.....	7
5.3 Parameter definitions.....	8
5.3.1 General.....	8
5.3.2 ASCII parameters.....	8
5.4 Parameter group definitions.....	13
5.5 Application notes.....	13
5.5.1 Parameters with multiple sources.....	13
5.5.2 Conventions for parameter placement notation and unspecified bits in message definitions.....	13
5.5.3 Parameter placement notation.....	13
5.5.4 Guidelines for parameter placement.....	15
5.5.5 Start position notation and parameter placement.....	15
5.5.6 Start position notation for fractional byte length parameters.....	15
5.5.7 Start position notation for integer byte length parameters.....	18
5.5.8 Start position notation for variable length parameters.....	20
5.5.9 Unspecified bits in the PG data field definition.....	21
6 Implement geometry.....	22
6.1 Implement geometry configuration.....	22
7 Navigation and location.....	22
7.1 Navigation and location system messages.....	22
7.2 Navigation and location parameters.....	23
8 Heartbeat.....	23
8.1 General.....	23
8.2 Heartbeat message implementation requirements.....	23
8.3 Heartbeat message handling.....	24
8.3.1 Requesting the heartbeat message.....	24
8.3.2 Heartbeat sequence number.....	24
8.3.3 Validation of the heartbeat sequence number.....	24
8.3.4 Validation of message timing.....	25
8.3.5 Error detected in communication.....	25
8.4 Diagnosing heartbeat.....	25
9 Auxiliary hydraulic valves.....	25
9.1 General.....	25
9.2 Numbering auxiliary valves.....	26

10	Working set messages	26
	10.1 General.....	26
	10.2 Working set application rules.....	26
11	Command messages	28
12	Maintenance agency	28
	Bibliography	29

STANDARDSISO.COM : Click to view the full PDF of ISO 11783-7:2022

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 23, *Tractors and machinery for agriculture and forestry*, Subcommittee SC 19, *Agricultural electronics*.

This fourth edition cancels and replaces the third edition (ISO 11783-7:2015) which has been technically revised. It also incorporates the Amendment ISO 11783-7:2015/Amd 1:2018.

The main changes are as follows:

- the annexes have been moved to ISO 11783 electronic database (<http://www.isobus.net/>);
- the technical requirements of parameter definitions have been clarified;
- all requirements from annexes have been moved to the main clauses except parameter definitions.

A list of all parts in the ISO 11783 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

ISO 11783 specifies a communications system for agricultural equipment based on the ISO 11898-1^[1] protocol. SAE J1939^[2] documents, on which parts of ISO 11783 are based, were developed jointly for use in truck and bus applications and for construction and agriculture applications. Joint documents were completed to allow electronic units that meet the truck and bus SAE J1939 specifications to be used by agricultural and forestry equipment with minimal changes.

General information on ISO 11783 is to be found in ISO 11783-1. The purpose of ISO 11783 is to provide an open, interconnected system for on-board electronic systems. It is intended to enable electronic control units (ECUs) to communicate with each other, providing a standardized system.

The ISO 11783 series specifies a serial data network for control and communications on forestry or agricultural tractors and mounted, semi-mounted, towed or self-propelled implements. Its purpose is to standardize the method and format of transfer of data between sensors, actuators, control elements and information storage and display units, whether mounted on, or part of, the tractor or implement.

STANDARDSISO.COM : Click to view the full PDF of ISO 11783-7:2022

Tractors and machinery for agriculture and forestry — Serial control and communications data network —

Part 7: Implement messages application layer

SAFETY PRECAUTIONS — Caution is to be taken with any automatic control of implements carried out using a message defined in this document. See ISO 11783-9 for safe-mode operations.

1 Scope

This document describes the implement messages application layer of the network, specifying the message set and defining the messages used for communication with and between tractors and connected implements.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 3166-1, *Codes for the representation of names of countries and their subdivisions — Part 1: Country code*

ISO 11783-1, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 1: General standard for mobile data communication*

ISO 11783-3, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 3: Data link layer*

ISO 11783-5, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 5: Network management*

ISO 11783-9, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 9: Tractor ECU*

ISO 11783-10, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 10: Task controller and management information system data interchange*

ISO 11783-12, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 12: Diagnostics services*

ISO 11783-13, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 13: File server*

IEC 61162-3, *Maritime navigation and radio communication equipment and systems — Digital interfaces — Part 3: Serial data instrument network*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 11783-1 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

3.1

service provider

network control function — whether in a single ECU or one control function of many within an ECU — that works and communicates with a working set as a separate entity

Note 1 to entry: It is not a member of the working set being described, but can be a member or master of another working set.

4 General requirements and recommendations

4.1 General

The message set specified by this document is designed to support the basic needs of an implement for information from a tractor, as well as limited controls enabling coordination between implement and tractor. The message set supports messages containing information on

- time,
- ground speed,
- distance,
- navigation,
- PTO (power take-off) parameters,
- three-point hitch,
- general process data, and
- lighting function parameters.

Some of the messages specified in this document are transmitted periodically and others are transmitted upon request.

The message parameters and parameter groups are defined in an electronic database as described in [4.2](#).

4.2 ISO 11783-7 electronic database

The electronic database with the ISO 11783-7 PGN and SPN assignments is accessible at: <http://www.isobus.net/>. The database contains information for the following.

- ISO 11783 Parameter definitions

The electronic database provides the current listing of the ISO 11783-7 parameter definitions which have been assigned and which are officially registered by SAE J1939. In J1939 terminology, a parameter is known as "Suspect Parameter" or "SP". A unique number is assigned to each parameter. This number is the "Suspect Parameter Number" or "SPN".

The electronic database provides the current listing of the ISO 11783-7 parameter groups (PGs) which have been assigned and which are officially registered by SAE J1939. A unique number is assigned to each parameter group. This number is the "Parameter Group Number" or "PGN".

The parameter definitions in the electronic database shall contain references to ISO 3166-1, ISO 11783-3, ISO 11783-5, ISO 11783-9, ISO 11783-10, ISO 11783-12 and ISO 11783-13.

5 Technical requirements

5.1 Overview

The application layer provides a means for application processes to access the OSI environment. This layer contains management functions and generally useful mechanisms to support applications.

5.2 General guidelines

5.2.1 Signal characterization

It is the intent of the SAE J1939/ISO 11783 network to provide current data and signals from a source so that it can be used by other nodes (CFs).

It is recommended that the time between physical data acquisition of a signal and the transmission of the data not exceed twice the repetition rate defined for the data. Additional constraints may be defined for certain parameters.

5.2.2 Message format

The message format of SAE J1939/ISO 11783 uses the parameter group number as the label for a group of parameters. Each of the parameters within the group can be expressed in ASCII, as scaled data defined by the ranges described in 5.2.4, or as function states consisting of two or more bits. Alphanumeric data is transmitted with the most significant byte first.

Most significant byte first for ASCII or alphanumeric data means the individual characters are positioned in the data field in left-to-right reading order of the ASCII string. The leftmost character of the ASCII string shall be transmitted first, and the rightmost character of the ASCII string shall be transmitted last. For example, if the ASCII string is, "The quick brown fox jumped over the lazy dog", then the ASCII character "T" shall be positioned so it is transmitted first, and the ASCII character "g" shall be positioned so it is transmitted last.

In general, characters conform to the ASCII character set as defined in ISO/IEC 8859-1.

Parameters consisting of two or more data bytes are transmitted least significant byte first. Further description of bit placement within a message is described in 5.5.3.

The type of data shall also be identified for each parameter. In J1939, data can be either status or measured; in ISO 11783, data can be either one of the J1939 types or estimated or command.

Each parameter has a data type of either a command or measured.

— Status

Status specifies the present state of a multi-state parameter or function as a result of action taken by the transmitting node (CF). This action is the result of a calculation which uses local and/or network "measured" and/or "status" and/or "estimated" and/or "command" information. Note that specific confirmation of this action is not necessarily assured. For instance, the status may indicate that a solenoid has been activated, yet no measurement may have been taken to ensure the solenoid accomplished its function.

EXAMPLE 1 Engine brakes are enabled, PTO speed control is active, cruise control is active.

— Measured

Measured data conveys the current value of a parameter, as measured or observed by the transmitting CF, determining the condition of the defined parameter.

EXAMPLE 2 Ground-based speed, hitch position, PTO engagement, implement in-work state.

— Estimated

Estimated data means that the value is achieved indirectly, without direct measurement.

EXAMPLE 3 Auxiliary valve estimated flow, General purpose valve estimated flow, Estimated curvature.

— Command

Command data specifies the desired state of a multistate parameter, function or numerical value as requested by a transmitting CF. Specific confirmation of a command is not required. For example, the command can request a solenoid be activated, yet no measurement be taken to ensure the solenoid has accomplished its function.

The tractor is not required to execute any command. The tractor may use its own logic to decide when it's appropriate to execute any command.

EXAMPLE 4 Engage PTO, extend auxiliary valve state, activate headlight high-beam, move rear hitch.

A device shall not receive parameter data from the network segment and retransmit that same parameter data using the same parameter back onto the same network segment.

5.2.3 ISO Latin 1 character set

There are 191 graphic characters of the ISO/IEC 8859-1 Latin 1 Character set shown in [Figure 1](#). Unless otherwise specified, only these 191 character values are permitted for ASCII parameters. The terminology "ASCII characters" and "printable ASCII characters" are used in SAE J1939/ISO 11783 to refer to this set of 191 graphic character values.

The remaining 65 characters values (0 through 31 and 127 through 159) are control functions. According to ISO/IEC 8859-1, these character values are defined in ISO/IEC 6429. The terminology "ASCII control characters" and "non-printable ASCII characters" are used in SAE J1939/ISO 11783 to refer to this set of 65 character values. As specified in ISO/IEC 6429, the character value 0 (zero) is the "NULL" character.

Horizontal boldface characters are the single hexadecimal digit representing the lower nibble of the single byte code for the character. Vertical boldface characters are the single hexadecimal digit representing the upper nibble of the single byte code for the character.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	----- should not be displayed -----															
1	----- should not be displayed -----															
2	space!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	nil
8	----- should not be displayed -----															
9	----- should not be displayed -----															
A	nil	ı	ç	£	¤	¥		§	¨	©	ª	«	¬	-	®	¯
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Figure 1 — ISO 8859-1 Latin 1 Character set

5.2.4 Parameter ranges

Table 1 defines the ranges used to determine the validity of a transmitted signal. Table 2 defines the ranges used to denote the state of a discrete parameter, and Table 3 defines the ranges used to denote the state of a control mode command. The values in the range “error indicator” provide a means for a module (CF) to immediately indicate that valid parametric data is not currently available due to some type of error in the sensor, sub-system, or module (CF).

The values in the range “not available” provide a means for a module to transmit a message which contains a parameter that is not available or not supported in that module. The values in the range “not requested” provide a means for a device to transmit a command message and identify those parameters where no response is expected from the receiving device.

If a component (CF) failure prevents the transmission of valid data for a parameter, the error indicator as described in Tables 1 and 2 shall be used in place of that parameter’s data. However, if the measured or calculated data has yielded a value that is valid, yet exceeds the defined parameter range, the error indicator shall not be used. The data shall be transmitted using the appropriate minimum or maximum parameter value. If the sensor cannot determine if the measured or calculated data are valid, it shall send the error indicator.

The operational range of the signal includes the valid signal range as well as any parameter-specific indicators that have been defined.

There are some exceptions where the entire range is valid for transmission and no means to broadcast “not available” exists. These are generally limited to parameters that are broadcast using the ASCII character set, parameters that are counters that wrap around once the limit is achieved, CRC calculations that use the entire range, parameters that are used to represent a manufacturer ID, a parameter group number (PGN), a suspect parameter number (SPN), or a failure mode indicator (FMI), and some bit-mapped parameters that use the entire range.

5.2.5 Assignment of ranges to new parameters

This subclause is intended to define a set of recommended scaling, limit, offset, and transfer function (SLOTS) which can be used when parameters are added to SAE J1939/ISO 11783. This permits data consistency to be maintained as much as possible between parameters of a given type (temperature, pressure, speed, etc.). Each SLOT is intended to provide a range and resolution suitable for most parameters within a given type. When necessary, a different scaling factor or offset can be used. All

SLOTS should be based on a power of two, scaling from another SLOT. This will minimize the maths required for any internal scaling and reduce the opportunity for misinterpreted values. Preferably, offsets should be selected on the following basis:

- a) Offset = 0, or
- b) Offset = 50 % (equal ± range)

The ISO 11783-7 electronic database defines the recommended SLOTS to be used when ranges are assigned to new parameters.

Unless otherwise specified, all pressure SLOTS are measured as gage pressure.

Table 1 — Logical signal ranges

Parameter size	Valid signal	Parameter-specific indicator	Reserved range for future indicator bits	Error indicator	Not available or Not requested
4 bits	0 ₁₆ to A ₁₆	B ₁₆	C ₁₆ to D ₁₆	E ₁₆	F ₁₆
8 bits	0 ₁₆ to FA ₁₆	FB ₁₆	FC ₁₆ to FD ₁₆	FE ₁₆	FF ₁₆
10 bits	0 ₁₆ to 3FA ₁₆	3FB ₁₆	3FC ₁₆ to 3FD ₁₆	3FE ₁₆	3FF ₁₆
12 bits	0 ₁₆ to FAF ₁₆	FB0 ₁₆ to FBF ₁₆	FC0 ₁₆ to FDF ₁₆	FE0 ₁₆ to FEF ₁₆	FF0 ₁₆ to FF ₁₆
16 bits	0 ₁₆ to FAFF ₁₆	FB00 ₁₆ to FBFF ₁₆	FC00 ₁₆ to FDFF ₁₆	FE00 ₁₆ to FEFF ₁₆	FF00 ₁₆ to FFFF ₁₆
20 bits	0 ₁₆ to FAFFF ₁₆	FB000 ₁₆ to FBFFF ₁₆	FC000 ₁₆ to FDFFF ₁₆	FE000 ₁₆ to FEFFF ₁₆	FF000 ₁₆ to FFFFF ₁₆
24 bits	0 ₁₆ to FAFFF ₁₆	FB0000 ₁₆ to FBFFF ₁₆	FC0000 ₁₆ to FDFFF ₁₆	FE0000 ₁₆ to FEFFF ₁₆	FF0000 ₁₆ to FFFFF ₁₆
28 bits	0 ₁₆ to FAFFFF ₁₆	FB00000 ₁₆ to FBFFFF ₁₆	FC00000 ₁₆ to FDFFFF ₁₆	FE00000 ₁₆ to FEFFFF ₁₆	FF00000 ₁₆ to FFFFF ₁₆
32 bits	0 ₁₆ to FAFFFF ₁₆	FB000000 ₁₆ to FBFFFF ₁₆	FC000000 ₁₆ to FDFFFF ₁₆	FE000000 ₁₆ to FEFFFF ₁₆	FF000000 ₁₆ to FFFFF ₁₆

Table 2 — Transmitted values for discrete parameters (measured or estimated)

Range name	Transmitted value
Disabled (Off, passive, etc.)	00 ₂
Enabled (On, active, etc.)	01 ₂
Error indicator	10 ₂
Not available or not installed	11 ₂

Table 3 — Transmitted values for control commands

Range name	Transmitted value
Command to disable function (turn Off, etc.)	00 ₂
Command to enable function (turn On, etc.)	01 ₂
Reserved	10 ₂
Don't care/take no action (leave function as is)	11 ₂

5.2.6 Adding parameters to groups

Several of the parameter groups contain bytes that are not defined and may be replaced with new parameters, as appropriate. If existing parameter group definitions do not permit the inclusion of a new parameter, a new parameter group may be defined. Per ISO 11783-1:2017 Annex A: "Request forms to obtain new ISO 11783-1 parameter group, address and identity assignments or request updates to the descriptions of existing ISO 11783-1 parameter group, address and identity assignments are available on the electronic database, accessible at: <http://www.isobus.net/>.

The maintenance agency (MA) appointed by the ISO Technical Management Board (ISO/TMB) processes the requests and synchronizes the code registrations with the SAE J1939 committee.

In general, parameters should be grouped into parameter groups as follows:

- a) by function (oil, coolant, fuel, etc.) and not by type (temperature, pressure, speed, etc.)
- b) with similar update rates (to minimize unnecessary overhead)
- c) by common subsystem (the device likely to measure and send data)

5.2.7 Transmission repetition rates — Update rates

5.2.7.1 Definition of transmission repetition rate

All transmission repetition rates defined in SAE J1939-71/the ISO 11783 electronic database are nominal rates. The actual transmission repetition rate on the network should be at this rate plus/minus the “typical” jitter which occurs in microcontroller-based systems. The average rate should be the nominal value.

5.2.7.2 Transmission repetition rate for on-change messages

Some periodic messages contain information that is of particular interest when a state change occurs. For example, it is desirable to immediately broadcast a change in the engine configuration rather than waiting a significant period of time for the next periodic update window.

Messages contain information that may change states at a very high rate. A rapidly changing state is not useful to consumers of this information, and unnecessarily increases bus loading.

Transmission repetition rate definition for these messages takes the form of:

Every MAXUPDATEPERIOD and on CHANGECRITERIA, but no faster than every MINUPDATEPERIOD

where:

CHANGECRITERIA = the criterion that prompts an immediate broadcast of a new message

MAXUPDATEPERIOD = The maximum period of the message. When CHANGECRITERIA is not satisfied, this is the preferred period of the message.

MINUPDATEPERIOD = The minimum period of the message. If CHANGECRITERIA indicates the message should be broadcast more often, the period must be equal to MINUPDATEPERIOD. This does not apply to the first message after a periodic broadcast.

Two acceptable implementations are illustrated in [Figures 2](#) and [3](#). In each illustration, the horizontal line represents time, the vertical bars topped with a numbered circle represent messages, and the diagonal line represents a timer that counts down to zero, which triggers the transmission of the next periodic message. In both illustrations, all messages are triggered by MAXUPDATEPERIOD, except for message 2, which is triggered by CHANGECRITERIA.

[Figure 2](#) shows the method where CHANGECRITERIA results in extra messages that do not change the timing of the subsequent periodic messages. In this illustration, message 2 is triggered by CHANGECRITERIA, but since the countdown timer is not reset, message 3 is then broadcast after MAXUPDATEPERIOD elapses since message 1.

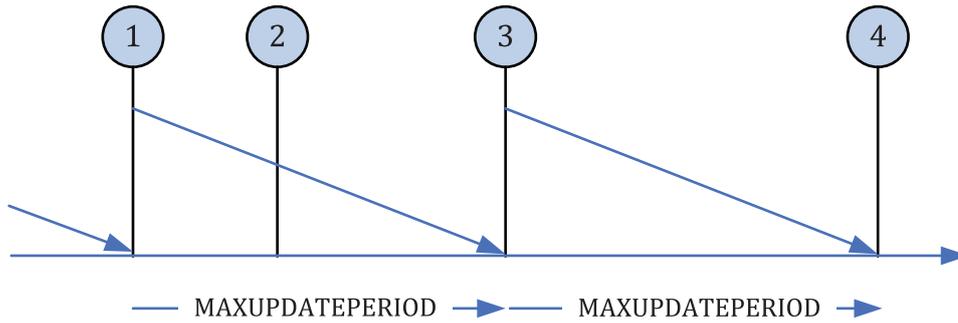


Figure 2 — On-change implementation option 1

Figure 3 shows the method where the message period is controlled by the last broadcast message. In this method, message 2 resets the timer, forcing message 3 to occur at a later time than if CHANGECRITERIA had not been satisfied. This implementation results in a lower average bus loading, as illustrated by the lack of message 4 in the same overall time as shown in the previous illustration.

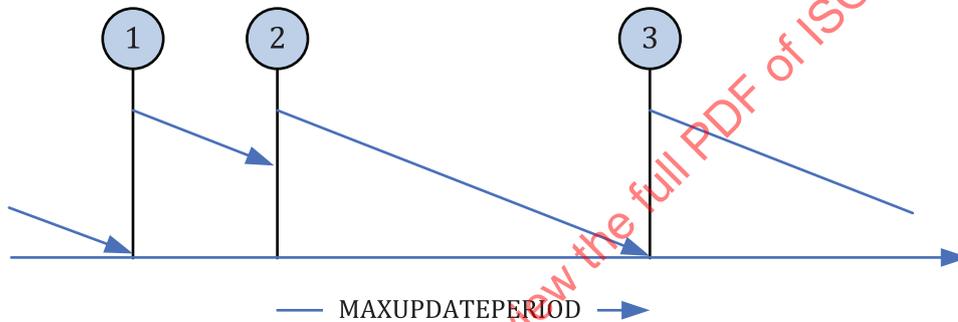


Figure 3 — On-change implementation option 2

This message definition was created after many “on change” messages were published. As a result, the implementation of those messages may vary from the description in this subclause. This subclause is intended to provide uniformity of future implementations of “on change” messages.

5.3 Parameter definitions

5.3.1 General

This subclause provides a description of each parameter used in the SAE J1939/ISO 11783 network. The description includes data length, data type, resolution, range, and a tag (label) for reference.

After power on, a node (CF) should internally set the “availability bits” of received parameters as not available and operate with default values until valid data is received. When transmitting, undefined bytes should be sent as 255 (FF₁₆) and undefined bits should be sent as 1.

5.3.2 ASCII parameters

5.3.2.1 General

This subclause describes the standard practices for ASCII data parameters. SAE J1939/ISO 11783 have three standard ASCII SLOT types for different data length designation techniques, which are summarized in Table 4. Some of the ASCII SLOT types use a delimiter technique for data length designation which may reduce the ASCII characters available for parameter data. The ASCII SLOT types are discussed individually in more detail in 5.3.2.2, 5.3.2.5, and 5.3.2.6.

The parameter definitions in the ISO 11783-7 electronic database (see 4.2) may have multiple ASCII SLOTS for the same ASCII SLOT type to accommodate different maximum bytes allowed. The numerical designator in the ASCII SLOT name reflects the maximum bytes allowed for an ASCII SLOT. For example, the ASCII SLOT “SAEatad0200” has a maximum length of 200 bytes, while the ASCII SLOT “SAEatad0025” has a maximum length of 25 bytes.

Table 4 — Summary of ASCII slot types

ASCII SLOT type	Description	Data length indication	Delimiter character included in data length?	Any characters not allowed within the data?	Required to fill data to a specific length?
Fixed length ASCII	The data length is a fixed or constant number of bytes.	Fixed number of bytes of data.	No. The length indicates required data space for parameter data.	No. All of the ASCII characters may be used in the data.	Yes. Shall provide data in all of the required number of bytes.
Character delimited, Variable length ASCII	The data length may vary within defined limits, and a specific ASCII character is placed following the data to indicate the end of the parameter data.	Length indicated by the delimiter character (always required).	No. The length indicates allowed data space for parameter data.	Yes. The delimiter character is not allowed within the parameter data, since it will be interpreted as the delimiter.	No, unless the parameter definition has a minimum data length of one or more characters.
Byte count delimited, Variable length ASCII	The data length may vary within defined limits, and a separate parameter within the PG data field specifies the byte length of the ASCII data.	Separate parameter that specifies the ASCII data byte length.	No. The length indicates allowed data space for parameter data.	No. All of the ASCII characters may be used in the data.	No, unless the parameter definition has a minimum data length of one or more characters.

5.3.2.2 ASCII characters

The character values for ASCII characters are specified in 5.2.3 of this document. By default, only the printable ASCII characters are allowed in the data for ASCII parameters. The ASCII control characters, or non-printable ASCII characters, are not allowed in the data for an ASCII parameter, unless the ASCII parameter definition explicitly states otherwise. SPN 162 and SPN 163 are examples of ASCII parameters with explicit statements allowing the use of ASCII control characters in the parameter data.

5.3.2.3 ASCII byte order

The standard practice for the ordering of data bytes for ASCII data parameters is defined in 5.2.2.

5.3.2.4 ASCII SLOT type - fixed length ASCII

The fixed length ASCII SLOT type defines an ASCII data parameter with a fixed, or non-varying, number of ASCII characters in the data field. All ASCII characters are available for use in the SP data with this type of ASCII SLOT.

Some examples of SPs using a fixed length ASCII SLOT type are SPNs 162, 3620, and 4254.

5.3.2.4.1 Parameter definition for fixed length ASCII data

A parameter using a fixed length ASCII SLOT type has the following data definition characteristics:

- the “Resolution” property indicates “ASCII”;
- the “Data Length” property is a fixed byte length, such as “5 bytes”.

All ASCII characters are available for use in the ASCII data with this type of ASCII SLOT.

The "Data Length" property defines the required byte length of the data for this parameter. If it is possible to have actual parameter data that is shorter than the required data length, then the parameter definition should specify the acceptable ASCII character(s) for an application to use to fill or pad the remaining data bytes. The definition should indicate if there is a standard for preference for inserting the pad or fill characters before or after the actual parameter data.

5.3.2.4.2 PG data field details for fixed length ASCII parameters

Within the PG data field, the specified number of bytes of data is required in the data field position for the fixed length ASCII data parameter, whether the source application (CF) supports the parameter or not.

The data for a subsequent parameter shall immediately follow the required number of data bytes.

If the source application (CF) is reporting data for the parameter, then the source application (CF) shall fill each of the data bytes for the parameter. If the actual parameter data is shorter than the required data length, then the source application (CF) is required to fill any remaining data bytes. The remaining data bytes shall be filled according to the parameter definition. If the parameter definition does not specify the data fill method, then the application shall fill the remaining bytes as appropriate for the data content.

If the source application (CF) does not support the parameter, then the source application (CF) is still required to fill each of the parameter data bytes with the "not available" value.

5.3.2.5 ASCII SLOT type - character delimited, variable length ASCII

5.3.2.5.1 General

The character delimited, variable length ASCII SLOT type defines an ASCII data parameter with a varying number of ASCII characters in the data field, and uses a specific ASCII character (delimiter) to indicate the end of the ASCII text for the parameter. All ASCII characters except for the delimiter character are permitted in the parameter data with this type of ASCII SLOT. The delimiter character is not permitted in the parameter data because it will be interpreted as the end of data indicator.

The delimiter character is not considered part of the data for the parameter. Consequently, the delimiter character is not included in the data length maximum byte length value in the SLOT definition. The delimiter is a mechanism within the PG data content to denote the end of the parameter data for the variable length ASCII parameter. However, this fundamental perspective should not be seen as restricting how the parameter data is handled internally by an application.

Some examples of parameters using a character delimited, variable length ASCII SLOT types are SPNs 965 and 234.

5.3.2.5.2 Parameter data definition for character delimited, variable length ASCII

A parameter using a character delimited, variable length ASCII SLOT type has the following data definition characteristics:

- the "Resolution" property indicates "ASCII";
- the "Data Length" property indicates a variable length, such as "Variable - up to 200 characters";
- the "Data Length" property indicates the delimiter character, such as "followed by an '*' delimiter".

All ASCII characters, except for the delimiter character, are available for use in the parameter data with this type of ASCII SLOT. The delimiter character is not permitted in the parameter data for this type of ASCII SLOT because it will be interpreted as the end of data indicator. The asterisk (*) character is the standard delimiter character for SAE J1939 parameters of this SLOT type. There is a SLOT type that

uses the NULL character (value zero) as the delimiter character. The SLOT type with a NULL delimiter character is appropriate when there is a need to have the asterisk character available as a valid data character rather than a delimiter.

The “Data Length” property defines the maximum length available for the ASCII data for the parameter. There is no minimum data length required for the data, unless the data length property explicitly states otherwise. The delimiter character is not included in the maximum data length value in the “Data Length” property. The delimiter character is specified within the SLOT definition and parameter definition because it places a restriction on the allowed ASCII characters for the parameter data. It is included in the “Data Length” property, since this property appears in the PG definition content.

5.3.2.5.3 PG data field details for character delimited, variable length ASCII

Within the PG data field, the maximum data length defines the maximum number of bytes available for the ASCII data for this parameter in the data field position. The designated delimiter character shall immediately follow the last valid byte of ASCII data for the parameter in the PG data field. The maximum data length does not define the required number of bytes for the data. A source application (CF) should not fill or pad the ASCII data for this type of parameter just to occupy the maximum length allowed. The delimiter character denotes the end of the data for the ASCII data parameter and indicates the starting position for a subsequent parameter. The delimiter is a mechanism within the PG data content to denote the end of the parameter data for the variable length ASCII parameter.

The data for a subsequent parameter shall immediately follow the delimiter character.

The delimiter character is always required after a delimited variable length ASCII data field within the PG data field, including situations when:

- the delimited variable length ASCII parameter is not supported by the source application (CF);
- the delimited variable length ASCII parameter is the only parameter in the data field;
- the delimited variable length ASCII parameter data is 0 bytes or characters in length;
- the delimited variable length ASCII parameter is the last parameter in the PG data field;
- the delimited variable length ASCII parameter data uses the maximum data length available for the parameter.

It is not necessary to include ASCII text for a delimited variable length ASCII parameter; however, the delimiter is always required. In other words, it is acceptable to transmit zero-length ASCII text for a variable length ASCII parameter as long as the delimiter character is included in the PG data.

The asterisk (*) character is the standard delimiter character for SAE J1939 parameters of this SLOT type. There is a SLOT type that uses the NULL character (Hex value zero) as the delimiter character. The SLOT type with a NULL delimiter character is appropriate when there is a need to have the asterisk character available as a valid data character rather than a delimiter.

The following examples are given to illustrate the PG data field content for several situations. For these examples, the letters “a” through “e” represent the data for five consecutive variable length ASCII parameters (asterisk * delimited) within the PG data field.

EXAMPLE 1 Data provided for each parameter. aaaa*bbb*c*dddd*eee*

EXAMPLE 2 Data only for parameters “a” and “b.” aaaaaaaa*bbbbbbbbbb****

EXAMPLE 3 Data only for parameter “b” and “d.” *bbbbbbb**dddd**

EXAMPLE 4 Data only for parameter “e.” ****e*****

5.3.2.6 ASCII SLOT type - byte count delimited, variable length ASCII

5.3.2.6.1 General

The byte count delimited, variable length ASCII SLOT type defines an ASCII data parameter with a varying number of ASCII characters in the data field, and relies upon a separate parameter to report the ASCII data parameter byte length. The ASCII parameter and the separate ASCII data byte length parameter must be transmitted in the same PG, since it is possible for the length of the ASCII data to vary from one instance of the parameter to another. All ASCII characters are permitted in the parameter data with this type of ASCII SLOT.

Special design considerations shall be recognized by any application that is the source of a PG with a parameter of this ASCII SLOT type. One design consideration involves maintaining synchronization between the value for the associated number of bytes parameter and the length of the ASCII parameter. Another design consideration involves the value reported for the data length parameter value if the ASCII parameter is not available or supported by the source.

5.3.2.6.2 Parameter data definition for byte count delimited, variable length ASCII

A parameter using a variable length ASCII with byte count parameter SLOT data type has the following data definition characteristics:

- The “Resolution” property indicates “ASCII”
- The “Data Length” property indicates a variable length, such as “Variable - up to 100 characters”
- The “Data Length” property does not specify a delimiter character
- The description notes identify the number of bytes parameter that reports the ASCII data byte length

All ASCII characters are available for use in the ASCII data with this type of ASCII SLOT.

The “Data Length” property defines the maximum length available for the ASCII data for the parameter. There is no minimum data length required for the ASCII data, unless the data length property explicitly states otherwise. A source application (CF) should not fill or pad the ASCII data for this type of parameter just to occupy the maximum length allowed.

5.3.2.6.3 PG data field details for byte count delimited, variable length ASCII

Within the PG data field, the maximum data length defines the maximum number of bytes available for the ASCII data for this parameter in the data field position. The maximum data length does not define the required number of bytes for the data. A source application (CF) should not fill or pad the ASCII data for this type of parameter just to occupy the maximum length allowed.

The number of bytes parameter shall be positioned somewhere before the ASCII data parameter within the PG data field to enable recipient applications (receiving CFs) to determine the end of the data for the ASCII parameter. The source application (transmitting CF) shall make sure the value in the number of bytes parameter is correct for the length of the instance of ASCII data parameter. The number of bytes parameter denotes the end of the ASCII data parameter and indicates the starting position for a subsequent parameter. The number of bytes parameter is the mechanism within the PG data content to denote the end of the data for the variable length ASCII parameter.

The data for a subsequent parameter shall immediately follow the specified number of bytes after the starting byte position for the ASCII data parameter.

5.4 Parameter group definitions

Parameter groups (PGs) for use on the SAE J1939/ISO 11783 network may be found in the ISO 11783-7 electronic database. All undefined and reserved bits shall be transmitted with a value of "1", unless noted otherwise in the particular parameter group definition.

Unique to this document are several messages with single-bit parameters such as availability of individual features. Portions of these messages may be reserved for additional single-bit indicators. The value of these reserved parameters may differ from the rules defined above. In some cases, the default value is zero ("0") for forward compatibility. The value of zero indicates "not supported" in these messages.

All undefined bits should be received as "don't care" (either masked out or ignored). This permits them to be defined and used in the future without causing any incompatibilities.

The size of the CAN data field is 8 bytes. Parameter groups that are 0 to 8 data bytes in length use the services of the data link layer. Refer to ISO 11783-3. Parameter groups that exceed 8 data bytes or parameter group definitions that are variable in length and may exceed 8 data bytes shall utilize the services of the transport protocol. Refer to ISO 11783-3. Network nodes (CFs) that receive multipacket messages must anticipate that the received message size may vary due to a possible assignment of new parameter to a given message.

5.5 Application notes

5.5.1 Parameters with multiple sources

Each parameter received by a node (CF) for control purposes shall be configurable by the system integrator to identify the primary source of the data, as well as the secondary source, if applicable. It is to be expected that the system integrator configures each receiving device on a network identically. A secondary source of data is defined to be a device on the network that measures the data independently of the primary source of that data.

5.5.2 Conventions for parameter placement notation and unspecified bits in message definitions

Subclauses [5.5.2](#) to [5.5.9](#) explain the various notations used by SAE J1939/ISO 11783 documents to specify the position of parameter data within the PG data field and illustrates the bit placement associated with the notations. It also explains how to deal with the unspecified bits in the data field definition. The information in this subclause is intended to aid the reader in determining the proper placement of parameter data based upon the "Start Position" and "Length" attributes specified in the PG definition. The information in this subclause is also intended to serve as a guide for how to properly define the Start Position attribute to define the placement of parameter data in a PG.

5.5.3 Parameter placement notation

5.5.3.1 Parameter data length classification notation

Three different classifications of parameters have been defined for the purposes of discussing parameter placement. The classifications are based upon the parameter data length. The three classifications are fractional byte length, integer byte length, and variable byte length. The "Start Position" notation is explained according to each of the parameter data length classifications.

- a) Fractional Byte Length: Classifies a parameter with a fixed data length where the data length is not an integer number of bytes. A parameter with a data length of 2 bits, a parameter with a data length of 5 bits, and a parameter with a data length of 10 bits are examples of fractional byte length parameters.
- b) Integer Byte Length: Classifies a parameter with a fixed data length where the data length is an integer number of bytes. A parameter with a data length of 1 byte, a parameter with a data

length of 2 bytes, and a parameter with a data length of 8 bits are examples of integer byte length parameters.

- c) Variable Byte Length: Classifies a parameter with a variable data length that is an integer number of bytes. A parameter with a data length of "Variable - up to 200 characters" is an example of a variable byte length parameter. Alphanumeric or textual data parameters are the primary examples of variable byte length parameters.

5.5.3.2 Notation for start position

The following notations are used throughout the parameter placement to describe the "Start Position" notation style.

- a) Fixed: Describes a "Start Position" notation that defines an absolute or fixed position for the placement of the parameter data in the data field. Some examples of fixed start position notations are "3," "5.4," "1-2," and "1.7-2."
- b) Equation: Describes a "Start Position" notation that defines the placement of the parameter data using an equation rather than an absolute position. Equation start position notations are appropriate when the parameter data length is variable, when the PG data field has multiple variable length parameters, or when there are fixed length parameters after variable length parameters in the data field. Some examples of equation start position notations are "14-n," "2 to n," "5 to A," and "A+1 to B."
- c) Field: Describes a "Start Position" notation that defines the placement of the parameter data in terms of its relative sequence in the data field rather than with an absolute position or equation. Field start position notations are appropriate when the PG data field has multiple consecutive variable length parameters in the data field or the parameter is repeated in the data field. The placement order of fields follows the alphabetical sequence of the start positions. Some examples of field start position notations are "a" "b," and "c."

5.5.3.3 Start position diagrams

Illustrations are included for many of the parameter placement notation styles to help clarify the parameter placement practices and the transmission order of the data over the SAE J1939/ISO 11783 network. These illustrations include one or more of the following diagrams.

- a) Data definition: The data diagram serves to illustrate the parameter data bits for the example data, shown where the data bits go highest order bit to lowest order significant bit in a left to right manner. Individual bits are identified with a "b" followed by a number. The "b" is the abbreviation for "bit" and the number denotes the significance order of the bit, where bits with lower significance have a lower number value. This diagram serves as a convenient way of discussing bit placement for the SAE J1939/ISO 11783 data order practices. In the ASCII examples, the "b" identifier may be preceded by a "c" plus a number to designate the character instance.
- b) Placement: The placement diagram illustrates the placement of the parameter bits using a common view of data in memory, where the bytes go most significant to least significant in a left to right manner and the bits within a byte go highest order to lowest order in a left to right manner. This diagram serves as a convenient way of discussing bit placement for the SAE J1939/ISO 11783 data order practices.
- c) Transmission Order: The transmission order diagram illustrates the parameter data bits in the order they are transmitted over the SAE J1939/ISO 11783 network. As specified in [5.1](#), the data is transmitted in increasing byte order (i.e., byte 1, byte 2, byte 3, etc.) with the bits within a byte transmitted highest order bit first (i.e., bit 8, bit 7, bit 6, etc.).

5.5.4 Guidelines for parameter placement

The following guidelines provide the basis for the parameter data placement conventions. These guidelines and the conventional parameter placement methods should be applied when defining the placement of parameters in PGs.

- a) Parameters with less than 8 bits should reside within a byte boundary.
- b) Parameters with more than 8 bits should either start or stop on a whole byte boundary.
- c) Only parameters with more than 8 bits should span a byte boundary.
- d) ASCII parameters, variable length parameters, and parameters with repeating data fields should start and stop on whole byte boundaries.
- e) Byte ordering rules are specified in [5.2.2](#).

5.5.5 Start position notation and parameter placement

The “Start Position” specified for a parameter in the PG definition and the “Length” attribute of the parameter describes the placement of the parameter data into the PG data field. Generally, the “Start Position” notation reflects the bit position for the lowest order bit of the parameter data within the byte. When the parameter data is confined to a single byte, then the “Start Position” consists of one numerical value declaring the position for the lowest order bit of the parameter data. When the parameter data spans one or more byte boundaries, then the “Start Position” consists of two numerical values; each declaring the position for the lowest order bit of the parameter data in the lowest and highest order bytes. For numerical start position notation, the integer value identifies the byte and the decimal value identifies the bit position (1 to 8, with 1 as the lowest order bit) within the byte. When the start position value does not include a decimal value, then the parameter data consumes the entire byte.

The “Start Position” notation has several formats to accommodate the different parameter data length types and the different parameter placement needs. For the purposes of parameter placement discussion, parameter data length is classified as fractional byte length (2 bits, 4 bits, 10 bits, etc.), integer byte length (1 byte, 2 bytes, etc.), and variable byte length. Each of these parameter length classifications have different requirements when it comes to specifying the position data field position of the data. Subclauses [5.5.6](#) to [5.5.8](#) explains the “Start Position” notation according to each of the parameter data length classifications.

5.5.6 Start position notation for fractional byte length parameters

Fractional byte length parameters are parameters with a data length that is not an integer number of bytes, for example, 2 bits, 5 bits, 10 bits, etc. The information in [Table 5](#) presents the “Start Position” notations used with fractional byte length parameters and explains the respective parameter placement. [Figure 4](#) to [Figure 8](#) show examples of these “Start Position” notations and illustrate the parameter placement.

The following guidelines explain how to determine data placement from the “Start Position” and “Length” attributes for a parameter with fractional byte length data.

- a) In the “Start Position” notation, the number before the decimal point identifies the byte and the number after the decimal point identifies the bit position within that byte.
- b) If the data length is less than 1 byte and all data bits are within the same byte, then the “Start Position” consists of one numerical value.
- c) If the data length is larger than 1 byte or the data spans a byte boundary, then the “Start Position” consists of two numerical values separated by a comma or dash. The number before the comma or dash is the first position designation and the number after the comma or dash is the second position designation.

- d) If a position designation in the “Start Position” does not have a decimal value, then the start bit is at bit 1 in that byte. For example, a position designation of “2” is equivalent to the position designation “2.1.” This abbreviated style is only used when the data occupies the whole byte. In [Table 5](#), a designation of “R” is equivalent to the designation “R.1,” and a designation of “S” is equivalent to the designation “S.1.” This is illustrated in [Figures 6 to 8](#). In [Figure 6](#), the second position designation is “2,” so the lowest order data bit placed into byte 2 will be positioned at bit 1.
- e) For fractional byte length data, the least significant data bit is always positioned at the first position designation, and each next higher order data bit is placed into the next higher order data field bit position. In [Table 5](#), “R.x” represents the first position designation, so the least significant bit of the data is placed at bit “x” of byte “R,” the next higher order bit of the data is placed at bit “x+1” of byte “R,” etc. This is illustrated in [Figures 4 to 8](#).
- f) When higher order data bit placement reaches a byte boundary and the next higher data field byte is an intermediate byte between the bytes specified in the first and second position designations, then the next higher order data bit is placed at bit 1 of the next higher order data field byte and additional higher order data bits are placed in next higher order fashion from that point. This is illustrated in [Figure 8](#). In [Figure 8](#), the “Start Position” notation identifies byte 6 in the first position designation and byte 8 in the second position designation, so byte 7 is an intermediate byte. When bit placement reaches byte 7, the next higher order data bit (bit “b9”) is placed at bit 1 of byte 7, and the next higher order data bits are placed into byte 7 in next higher order fashion from that point.
- g) When higher order data bit placement reaches a byte boundary and the next higher data field byte is the byte identified in the second position designation in the “Start Position,” then the number after the decimal in the second position designation indicates the bit position in that byte where the next higher order data bit is placed in the byte and any remaining higher order data bits are to be placed in next higher order fashion from that point. In [Table 5](#), “S.w” represents the second position designation, so when data bit placement reaches byte “S” of the data field, the next higher order bit of the data is placed at bit “w” of byte “S,” the next higher order bit of data after that is placed at bit “w+1” of byte “S,” etc. This is illustrated in [Figures 5, 7, and 8](#). In [Figure 8](#), the second position designation is “8.6.” When bit placement gets to byte 8, then next higher order data bit, bit “b17,” is placed at bit 6 of byte 8 and the last two bits, “b18” and “b19,” are placed at bit 7 and bit 8 of byte 8, respectively.

Table 5 — Start position notation for fractional byte length parameters

Start position	Length	Interpretation	Example illustration
R.x	Y bits (Y less than 8)	Fixed position of the data within a byte boundary for a fractional byte length parameter with less than 8 bits. The parameter occupies “Y” number of bits of byte “R” with the least significant bit of the parameter data at bit “x” in byte “R” and the most significant bit of the parameter data is at bit (“x” + (“Y”-1)) in byte “R.”	Figure 4
R.x-S.w	Y bits (Y less than 8)	Fixed position of the data across a byte boundary for a fractional byte length parameter with less than 8 bits. The parameter occupies the most significant bits of byte “R” from bit “x” to bit 8 and the remaining number of data bits start from bit “w” in byte “S.” The least significant bit of the parameter data is placed at bit “x” in byte “R.”	Figure 5
R.x-S	Y bits (Y greater than 8)	Fixed position of a fractional byte length parameter with more than 8 bits where the data crosses a byte boundary and stops on a whole byte. The parameter occupies the most significant bits of byte “R” from bit “x” to bit 8 plus all whole bytes up to “S.”	Figure 6
R-S.w	Y bits (Y greater than 8)	Fixed position of a fractional byte length parameter with more than 8 bits where the data crosses a byte boundary and starts on a whole byte. The parameter occupies all whole bytes from “R” up to “S” and the remaining modulo-8 number of bits starting from bit “w” in byte “S.”	Figure 7 , Figure 8 **

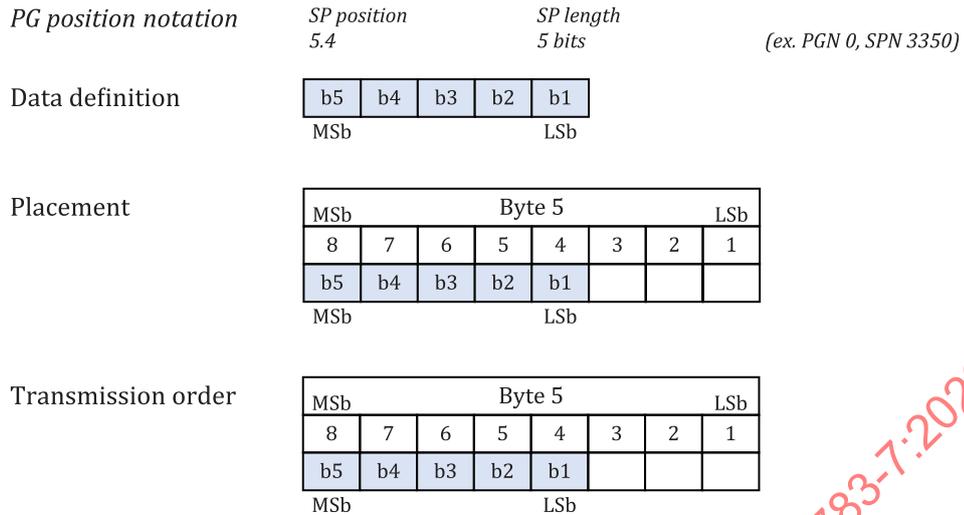


Figure 4 — Fractional byte (less than 1 byte) within byte boundary

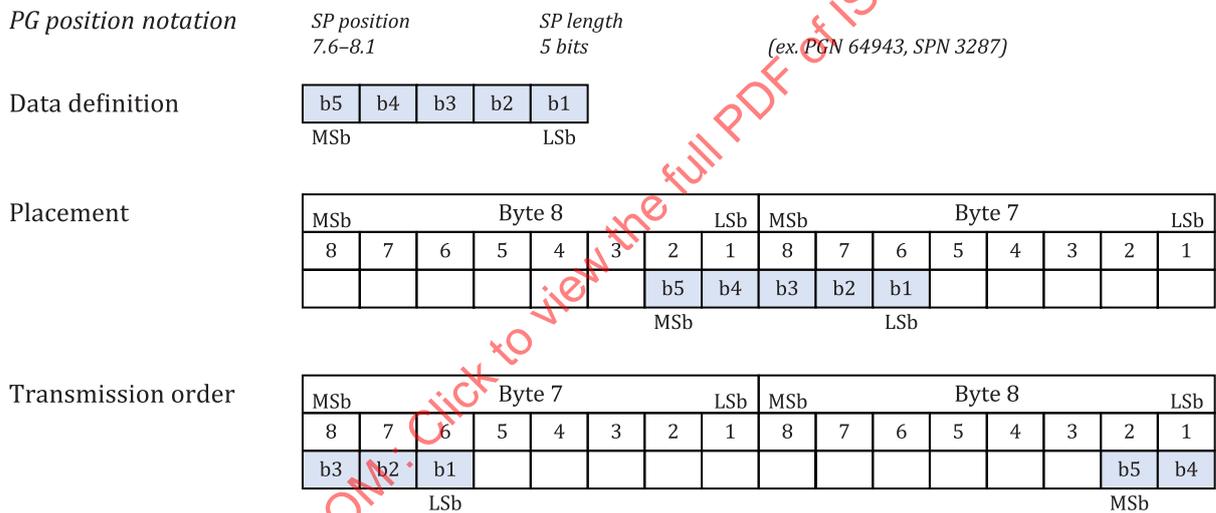


Figure 5 — Fractional byte (less than 1 byte) across byte boundary

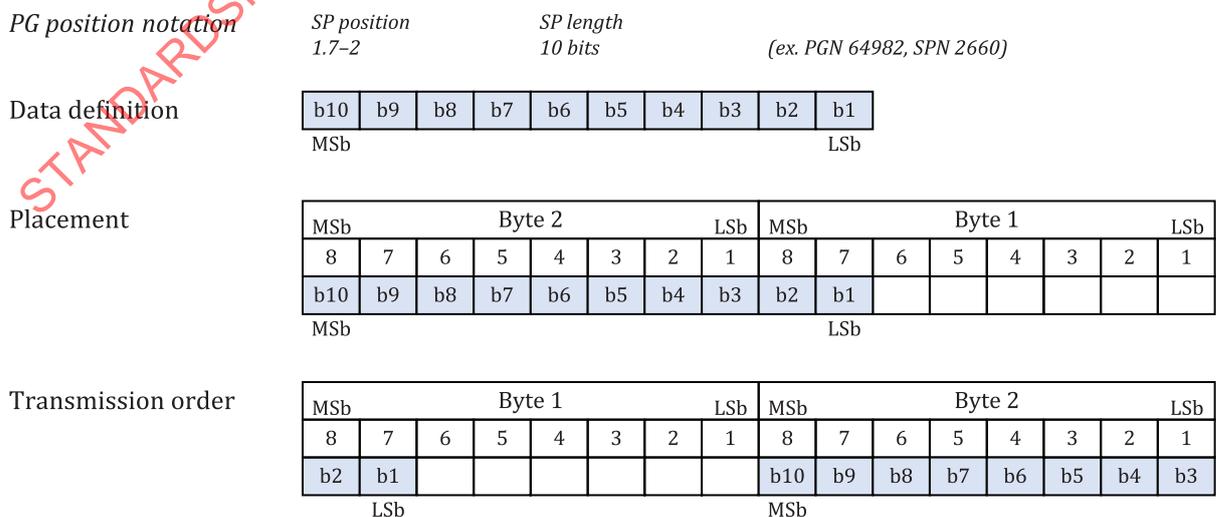


Figure 6 — Fractional byte (larger than 1 byte) ending on byte boundary

PG position notation SP position SP length
 6-7.1 6-7.1 12 bits (ex. PGN 64955, SPN 3169)

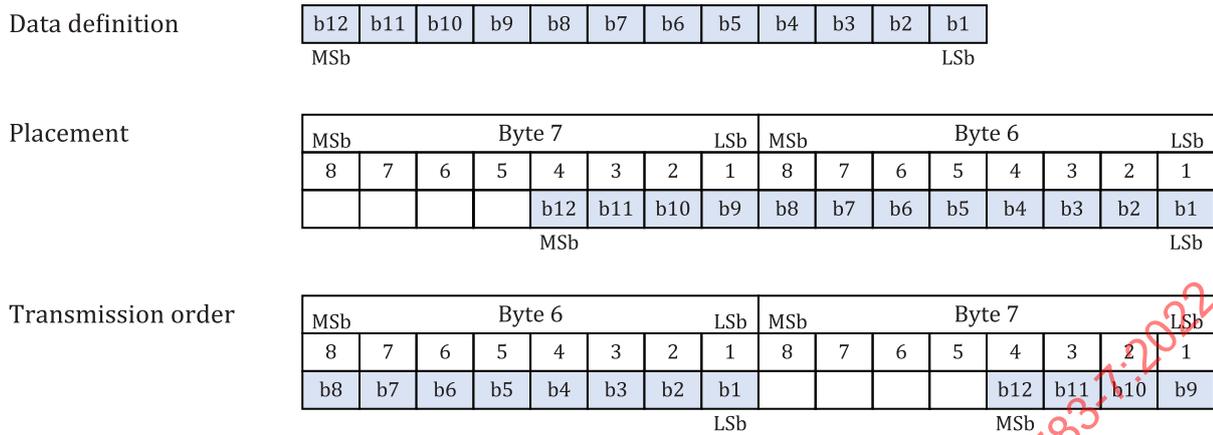


Figure 7 — Fractional byte (larger than 1 byte) starting on byte boundary

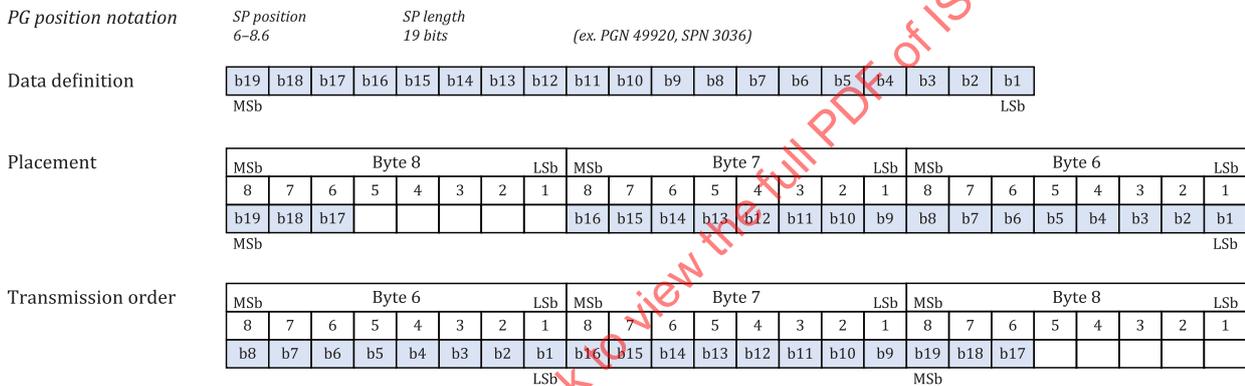


Figure 8 — Fractional byte (larger than 1 byte) starting on byte boundary

5.5.7 Start position notation for integer byte length parameters

The information in [Tables 6 and 7](#) present “Start Position” notations used with parameters with integer byte length. Examples of these “Start Position” notations are illustrated in [Figures 9 to 12](#). Integer byte length parameters are parameters with a fixed data length in whole bytes, e.g. 1 byte, 2 bytes, 4 bytes, 16 bits, etc. The placement of the data bytes for integer byte length parameters larger than 1 byte depends upon whether the parameter is non-alphanumeric (e.g. scaled data or state list) or alphanumeric. As noted in [5.2.2](#), the placement or ordering of the data bytes for multiple byte parameters differs between alphanumeric and all other data types. The parameter definition must be referenced to determine if the parameter is non-alphanumeric or alphanumeric data.

Table 6 — Start position notation for integer byte length parameters (non-alphanumeric)

Start position	Length	Interpretation	Example illustration
R	1 byte or 8 bits	Fixed position of a one byte data parameter within a whole byte. The parameter occupies the entire byte “R.”	Figure 9

Table 6 (continued)

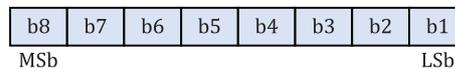
Start position	Length	Interpretation	Example illustration
R-S R, S R	Y bytes or 16 bits	Fixed position of a multiple byte data. Since this parameter is non-alphanumeric data (based upon parameter definition), the data is positioned so the least significant byte is transmitted first, per 5.2.2. The parameter occupies “Y” number of bytes from byte “R” through byte “S.”	Figure 10, Figure 11

Table 7 — Start position notation for integer byte length parameters (alphanumeric)

Start position	Length	Interpretation	Example illustration						
R	1 byte or 8 bits	Fixed position of a one-byte data parameter within a whole byte. The parameter occupies the entire byte “R.”	Figure 9						
R-S	Y bytes	Fixed position of a multiple byte data. Since this parameter is alphanumeric data (based upon parameter definition), the data is positioned so the most significant byte is transmitted first, per 5.2.2. The parameter occupies “Y” number of bytes from byte “R” through byte “S.”	Figure 12						
“n”	Y bytes	Field position of a multiple byte data. Since this parameter is alphanumeric data (based upon parameter definition), the data is positioned so the most significant byte is transmitted first, per 5.2.2. The parameter occupies “Y” number of bytes from the point that the field starts (i.e. in the first byte following the previous field). Example (PGN 64912, SPN 3560 and 3561 - defined in SAE J1939) <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Start position</th> <th>Length</th> </tr> </thead> <tbody> <tr> <td>a</td> <td>2 bytes (SPN 3560)</td> </tr> <tr> <td>b</td> <td>2 bytes (SPN 3561)</td> </tr> </tbody> </table> The structure of these two parameters repeats in the data field. The 2 bytes of data for SPN 3561 (field “b”) is placed in the 2 bytes following the last byte of SPN 3560 (field “a”).	Start position	Length	a	2 bytes (SPN 3560)	b	2 bytes (SPN 3561)	
Start position	Length								
a	2 bytes (SPN 3560)								
b	2 bytes (SPN 3561)								

<i>PG position notation</i>	<i>SP position</i>	<i>SP length</i>	
	3	8 bits	(ex. PGN 61451, SPN 2928)
	3	1 byte	(ex. PGN 256, SPN 525)

Data definition



Transmission order

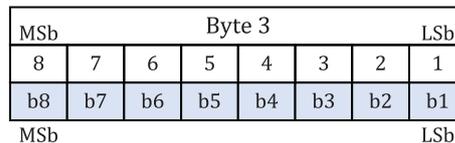


Figure 9 — Single byte data placement (non-alphanumeric and alphanumeric)

<i>PG position notation</i>	<i>SP position</i>	<i>SP length</i>	
	3	2 bytes	(ex. PGN 61454, SPN 3217)
	3-4	2 bytes	(ex. PGN 61450, SPN 132)
	3,4	2 bytes	(ex. PGN 64957, SPN 3085)

Data definition

b16	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1
MSb								LSb							

Transmission order

MSb								LSb							
Byte 3				Byte 4				MSb				LSb			
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
b8	b7	b6	b5	b4	b3	b2	b1	b16	b15	b14	b13	b12	b11	b10	b9
LSb								MSb							

Figure 10 — Multiple byte placement (non-alphanumeric data)

<i>PG position notation</i>	<i>SP position</i>	<i>SP length</i>	
	1-4	4 bytes	(ex. PGN 65199, SPN 1039)
	01-04	4 bytes	(ex. PGN 65211, SPN 994)

Data definition

b32	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1
MSb																LSb															

Transmission order

MSb								LSb								MSb								LSb							
Byte 1				Byte 2				Byte 3				Byte 4				MSb				LSb											
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
b8	b7	b6	b5	b4	b3	b2	b1	b16	b15	b14	b13	b12	b11	b10	b9	b24	b23	b22	b21	b20	b19	b18	b17	b32	b31	b30	b29	b28	b27	b26	b25
LSb								MSb								LSb															

Figure 11 — Multiple byte placement (non-alphanumeric data)

<i>PG position notation</i>	<i>SP position</i>	<i>SP length</i>	
	5-6	2 bytes	(ex. PGN 61445, SPN 162 - ASCII)

Data definition

ASCII Character 1								ASCII Character 2							
c1b8	c1b7	c1b6	c1b5	c1b4	c1b3	c1b2	c1b1	c2b8	c2b7	c2b6	c2b5	c2b4	c2b3	c2b2	c2b1
MSb				LSb				MSb				LSb			

Transmission order

MSb								LSb								MSb								LSb							
Byte 5				Byte 6				MSb				LSb																			
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1																
c1b8	c1b7	c1b6	c1b5	c1b4	c1b3	c1b2	c1b1	c2b8	c2b7	c2b6	c2b5	c2b4	c2b3	c2b2	c2b1																
MSb								LSb								MSb								LSb							

Figure 12 — Multiple byte placement (alphanumeric data)

5.5.8 Start position notation for variable length parameters

The information in [Table 8](#) presents “Start Position” notations used with variable length parameters. Alphanumeric or textual data parameters are the primary examples of variable byte length parameters. The “Start Position” is typically denoted using letters and equations to explain the position of the content within the message data field.

Table 8 — Start position notation for variable length parameters

Start Position	Length	Interpretation										
R-"N"	Variable - up to Y characters ("*" delimited)	<p>The parameter starts at byte "R" and continues through some variable number of bytes where the end is denoted by an asterisk character in the data stream. The length of the parameter does not include the "*" delimiter.</p> <p>Example (PGN 65242, SPN 234)</p> <table> <tr> <td>Start Position</td> <td>Length</td> </tr> <tr> <td>2-N</td> <td>Variable - up to 200 characters ("*" delimited)</td> </tr> </table> <p>Parameter starts at byte 2 and continues up to the asterisk character (at byte 203 at the highest).</p>	Start Position	Length	2-N	Variable - up to 200 characters ("*" delimited)						
Start Position	Length											
2-N	Variable - up to 200 characters ("*" delimited)											
R to "N" "N"+1 to "P"	<p>Variable - up to Y characters</p> <p>Variable - up to Y characters</p>	<p>Equations define the starting position of two consecutive variable length parameters. The first parameter starts at byte "R" and continues through some variable number of bytes. And the second parameter starts at the first byte last character of the first parameter and continues through some variable number of bytes.</p> <p>Example (PGN 64958, SPNs 3074 and 3075)</p> <table> <tr> <td>Start Position</td> <td>Length</td> </tr> <tr> <td>2</td> <td>1 byte (Number of bytes in SPN 3074)</td> </tr> <tr> <td>3</td> <td>1 byte (Number of bytes in SPN 3075)</td> </tr> <tr> <td>5 to A</td> <td>Variable - up to 100 characters (SPN 3074)</td> </tr> <tr> <td>A+1 to B</td> <td>Variable - up to 100 characters (SPN 3075)</td> </tr> </table> <p>First variable length parameter starts at byte 5 and continues through the number of bytes specified in byte 2. The second variable length parameter starts at first byte after SPN 3074 data and continues through the number of bytes specified in byte 3.</p>	Start Position	Length	2	1 byte (Number of bytes in SPN 3074)	3	1 byte (Number of bytes in SPN 3075)	5 to A	Variable - up to 100 characters (SPN 3074)	A+1 to B	Variable - up to 100 characters (SPN 3075)
Start Position	Length											
2	1 byte (Number of bytes in SPN 3074)											
3	1 byte (Number of bytes in SPN 3075)											
5 to A	Variable - up to 100 characters (SPN 3074)											
A+1 to B	Variable - up to 100 characters (SPN 3075)											
"n"	Variable - up to Y characters ("*" delimited)	<p>Field position of a variable length data parameter. The parameter is the nth ordered field. The parameter occupies the first data byte following the previous parameter and continues some variable number of bytes where the end is denoted by an asterisk character in the data stream. The length of the parameter does not include the "*" delimiter.</p> <p>Example (PGN 64965, SPN 2903)</p> <table> <tr> <td>Start Position</td> <td>Length</td> </tr> <tr> <td>c</td> <td>Variable - up to 200 characters ("*" delimited)</td> </tr> </table> <p>Parameter is the third field and continues up to the asterisk delimiter character (201 bytes beyond start of third field at the highest). The starting byte number depends upon the length of the data before this field.</p>	Start Position	Length	c	Variable - up to 200 characters ("*" delimited)						
Start Position	Length											
c	Variable - up to 200 characters ("*" delimited)											

5.5.9 Unspecified bits in the PG data field definition

5.5.9.1 General

Unspecified bits are the bits within the PG data field byte length that are not assigned to a parameter or are not used by the data for the collection of parameters in the PG. In the SAE J1939/ISO 11783 PG definitions, the unspecified bits are typically not shown or explicitly identified in the PG definition.

The "Data Length" property of the PG definition specifies the minimum and maximum byte length of the data field for the PG. The transmitted data field shall be at least the minimum length specified by the "Data Length" property for the PG, and all unspecified bit within the transmitted data field shall be

filled with a value of one. This document makes it possible to assign unspecified bits to parameters at some future time.

5.5.9.2 Unspecified bits — Illustrated example

An example of unspecified bits is provided in Figure 13 using an example PGN 12345. The top section of Figure 13 shows the PG data length and PG content definition for the PG. There are 36 unspecified bits in the PG definition in this example. The unspecified bits are bit 5 to bit 8 of byte 2 (4 bits total) and all bits in byte 4 through byte 8 (32 bits total).

The PG definition indicates SPN_1 is a 1-byte parameter with a data start position at byte 1. Since SPN_1 occupies all the bits in byte 1, there are no unspecified bits in byte 1. Next, the PG definition indicates SPN_2 is a 4-bit parameter with a starting position of “2.1,” which means the data for SPN_2 occupies bit 1 to bit 4 of byte 2. The next parameter in the PG definition has a starting position in byte 3, which means bit 5 through bit 8 of byte 2 are Unspecified Bits. The third parameter in the PG definition indicates SPN_3 is a 2-byte parameter with a data start position of “3- 4.” Since SPN_3 occupies all the bits in bytes 3 and 4, there are no unspecified bits in byte 3 or byte 4. Finally, the PG “Data Length” property indicates the PG has a message data field length of 8 bytes, but the PG definition only lists parameter content through byte 4. All of the bits in byte 5 through byte 8 are Unspecified Bits. When transmitted, the message data field for this PG shall be 8 bytes in length, as specified by the PG Data Length property. The 36 Unspecified Bits must be set to one, and the other 28 bits for the data for SPNs SPN_1, SPN_2, and SPN_3 must be filled appropriately.

PGN 12345 Example J1939 PGN Message
 PG data length: 8 bytes

SP position	SP length	Parameter name	SPN
1	1 byte	Example parameter 1	SPN_1
2.1	4 bits	Example parameter 2	SPN_2
3-4	2 bytes	Example parameter 3	SPN_3

PG data

Byte 1								Byte 2								Byte 3								Byte 4											
MSb	8	7	6	5	4	3	2	1	MSb	8	7	6	5	4	3	2	1	MSb	8	7	6	5	4	3	2	1	MSb	8	7	6	5	4	3	2	1
b8	b7	b6	b5	b4	b3	b2	b1	1	1	1	1	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	b16	b15	b14	b13	b12	b11	b10	b9				
SPN_1 data								Unspecified bits				SPN_2 data				SPN_3 data																			
Byte 5								Byte 6								Byte 7								Byte 8											
MSb	8	7	6	5	4	3	2	1	MSb	8	7	6	5	4	3	2	1	MSb	8	7	6	5	4	3	2	1	MSb	8	7	6	5	4	3	2	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
Unspecified bits																																			

Figure 13 — Unspecified bits example

6 Implement geometry

6.1 Implement geometry configuration

The geometry configuration of a tractor/implement connection and the offset to and from the tractor and implement reference points are used in the navigational parameters and in the implement configuration of process data messages. The definition and use of geometry parameters are specified in ISO 11783-10.

7 Navigation and location

7.1 Navigation and location system messages

CFs on an ISO 11783 network shall use the navigation location messages specified in IEC 61162-3 (NMEA 2000). The preferred (minimum) messages for ISO 11783 are “GNSS position data”, “Position