INTERNATIONAL STANDARD

ISO 11783-3

First edition
1998-07-01

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

Part 3:
Data link layer

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

Partie 3: Couche liaison de données

Reference number
ISO 11783-3:1998(E)

Contents

Page

International Organization for Standardization
Case postale 56 • CH-1211 Genève 20 • Switzerland
Internet central@iso.ch
X.400 c=ch; a=400net; p=iso; o=isocs; s=central

Printed in Switzerland

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.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

International Standard ISO 11783-3 was prepared by Technical Committee ISO/TC 23, *Tractors and machinery for agriculture and forestry*, Subcommittee SC 19, *Agricultural electronics*.

ISO 11783 consists of the following parts, under the general title *Tractors and machinery for agriculture and forestry — Serial control and communications data network*:

— *Part 1: General standard for mobile data communication*

— *Part 2: Physical layer*

— *Part 3: Data link layer*

— *Part 4: Network layer*

— *Part 5: Network management*

— *Part 6: Virtual terminal*

— *Part 7: Basic applications layer*

— *Part 8: Power train*

— *Part 9: Tractor ECU*

— *Part 10: Process data applications layer*

— *Part 11: Task controller and management information system data interchange*

Annexes A and B form an integral part of this part of ISO 11783. Annexes C and D are for information only.

Introduction

Parts 1 to 11 of ISO 11783 specify a communications system for agricultural equipment based on the CAN 2.0 B protocol. SAE J 1939 documents, on which parts of ISO 11783 are also 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 J 1939 specifications to be used by agricultural and forestry equipment with minimal changes.

The purpose of ISO 11783 is to provide an open interconnected system for on-board electronic systems. It is intended to enable electronic units to communicate with each other providing a standardized system.

The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this part of ISO 11783 may involve the use of a patent concerning the Controller Area Network (CAN) protocol referred to throughout the document.

ISO takes no position concerning the evidence, validity and scope of this patent.

The holder of this patent has assured ISO that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants (manufacturers of CAN devices) throughout the world. For this purpose, the statement of the holder of this patent is registered with ISO. Information may be obtained from:

Robert Bosch GmbH
Wernerstraße 51
Postfach 30 02 20
D-70442 Stuttgart-Feuerbach
Germany

Attention is drawn to the possibility that some of the elements of this part of ISO 11783 may be the subject of patents other than that identified above. ISO shall not be held responsible for identifying any and all such patent rights.

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

Part 3:
Data link layer

1 Scope

This part of ISO 11783 specifies a serial data network for control and communications on forestry or agricultural tractors, mounted, semi-mounted, towed or self-propelled implements. Its purpose is to standardize the method and format of data transfer between sensor, actuators, control elements, information storage and display units whether mounted or part of the tractor, or any implements. This part of ISO 11783 describes the CAN 29 Bit Identifier [5] Data Link Layer [6].

2 General description

The data link layer enables the reliable transfer of data across the physical link. This consists of sending the CAN data frame with the necessary synchronization, error control and flow control. The flow control is accomplished through a consistent message frame format.

3 Technical requirements

3.1 Message frame format

The message frame format shall conform to the CAN requirements. The CAN specification referenced throughout this document is CAN Specification version 2.0 B [5]. It should be noted that when there are differences between the above-mentioned CAN specification and ISO 11783, ISO 11783 is the governing document.

The CAN document specifies, in an information routing related discussion, that controller addresses are not used. While this is true for some applications of CAN, it is not true for ISO 11783. The definition of the network according to ISO 11783 requires that controller addressing be used to prevent multiple controllers from using the same CAN Identifier field. Many additional requirements are given in ISO 11783 that are not specified by CAN.

CAN 2.0 B specifies two message frame formats, Standard Frame and Extended Frame. "CAN 2.0 B compatibility" implies that messages of both format can potentially be present on a single network, by using certain bit codings which allow for the recognition of the different formats. ISO 11783 also accommodates both message frame formats; however, ISO 11783 only defines a full strategy for standardized communications using the Extended Frame format. All standard frame format messages are for proprietary use following the rules defined in this part of ISO 11783.

Therefore, controllers according to ISO 11783 must use the Extended Frame format. Standard Frame format messages can reside on the network, but only as described in this part of ISO 11783.

NOTE — Standard frame controllers do not respond to network management messages and will not be able to support the strategy for standardized communications.

The CAN data frame is parsed into different bit fields, as shown in figure 1. The number and parsing of the bits in the arbitration and control field differ between the CAN Standard and CAN Extended Frame messages. CAN

Standard Frame messages, shown in "A", contain 11 Identifier bits in the arbitration field and CAN Extended Frame messages, shown in "B", contain 29 Identifier bits in the arbitration field. ISO 11783 further defines the Identifier bits in the arbitration field of the CAN message frame formats. These definitions are given in table 1.

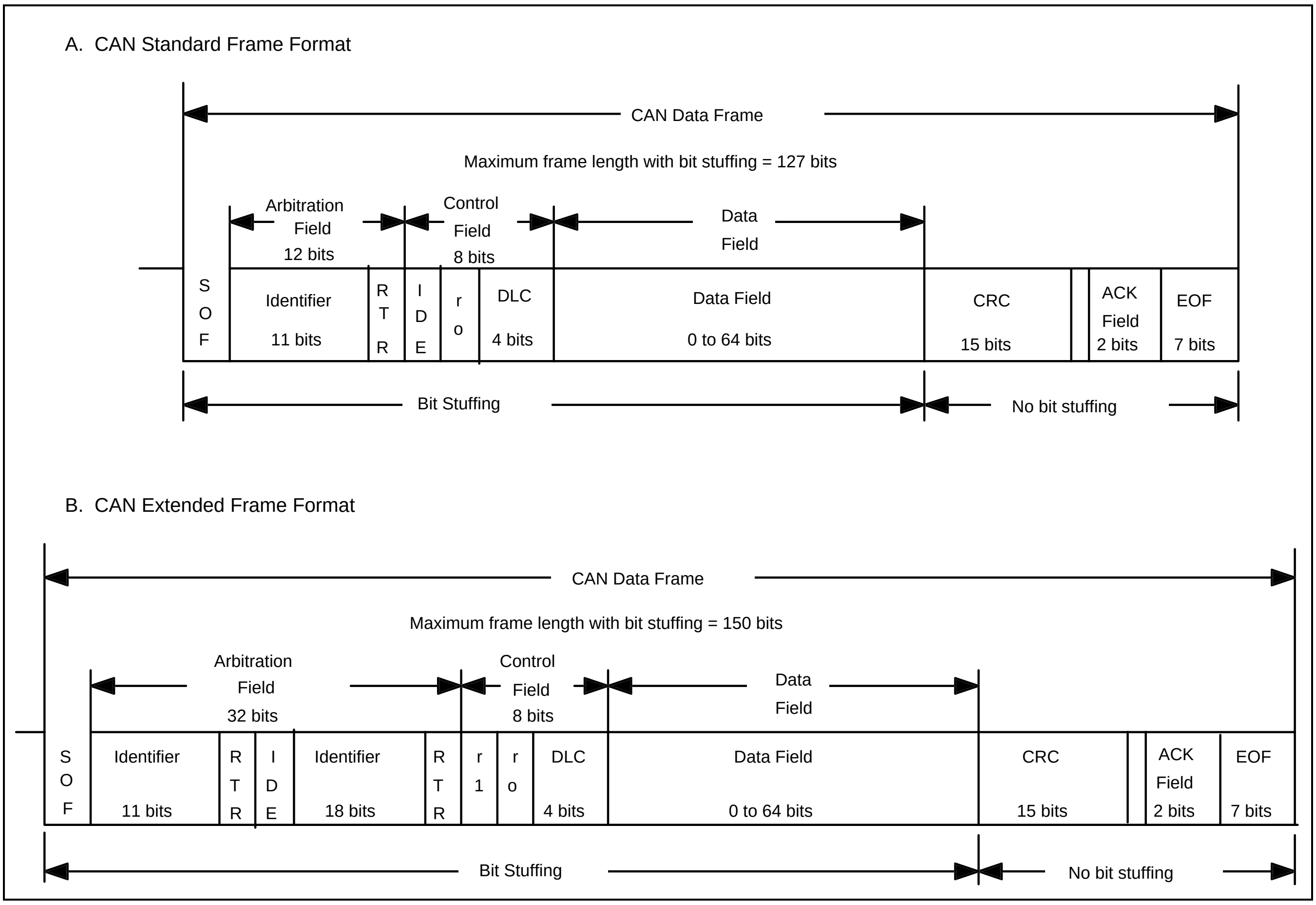


Figure 1 — CAN Data Frames

3.1.1 Message frame format according to ISO 11783 (CAN 2.0 B Extended Frame Format)

The CAN extended frame message, as shown in figure 1, encompasses a single Protocol Data Unit (PDU). PDUs consists of seven pre-defined fields. These fields are assimilated from information provided by the application layer. They are Priority, Reserved, Data Page, PDU Format, PDU Specific (which can be a destination address, Group Extension, or proprietary), Source Address and Data Fields. They are then packaged into one or more CAN data frames and sent over the physical media to other network controllers. The layers of the OSI model that ISO 11783 supports are shown in figure 2. It should be recognised that some Parameter Group definitions require more than one CAN data frame to send their information. Table 1 shows the arbitration and control fields of the 29 bit identifier for CAN, 29 bit identifier for ISO 11783, 11 bit identifier for CAN, and the 11 bit identifier for ISO 11783. A complete definition for each of the bit field assignments according to ISO 11783 is given in 3.3.

3.1.2 Parameter Group Number (PGN)

The Parameter Group Number is determined from the following constituent components: Reserved bit, Data Page bit, PDU Format Field (8 bits), and Group Extension Field (8 bits). These 18 bits are used to establish the 24 bit PGN. Whenever it is necessary to identify a Parameter Group in the data-field of a CAN data frame, it is expressed in 24 bits with the most significant bits set to zero. See table 2 for an illustration of PGNs, their corresponding bits and their conversion to a decimal number.

The procedure for the bit fields to be converted to Parameter Group Numbers is as follows. If the PDU Format (PF) value is less than 240 ($F0_{16}$), then the lower byte of the PGN is set to zero.

NOTE — Not all 131 072 combinations (2^{17}) are available to be assigned as PGNs. Only a total of 8 670 combinations are available for assignment (calculated as: 2 pages × [240 + (16 × 256)] = 8 670, using the conventions specified in this part of ISO 11783).

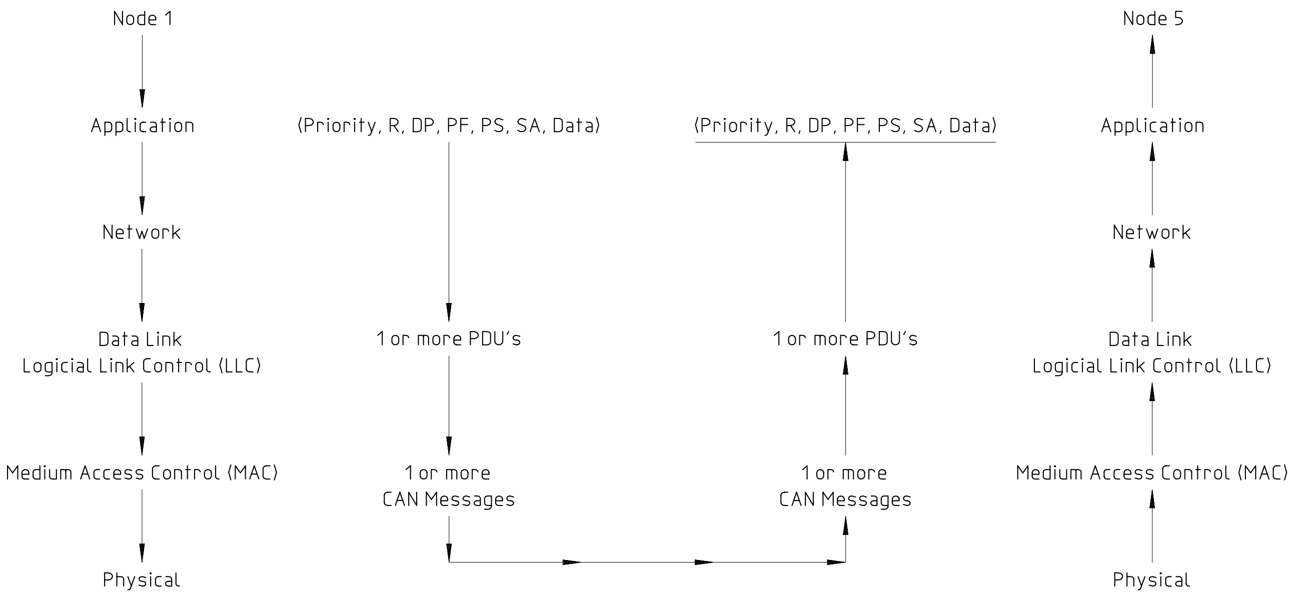
Table 1 — Mapping of ISO 11783 into CAN's Arbitration and Control fields

Bit No.	29 bit identifiers		11 bit identifiers	
	CAN	**ISO 11783**	**CAN**	**ISO 11783**1)
1	SOF	SOF*)	SOF	SOF*)
2	ID 28	P 3	ID 11	P 3
3	ID 27	P 2	ID 10	P 2
4	ID 26	P 1	ID 9	P 1
5	ID 25	R 1	ID 8	SA 8
6	ID 24	DP	ID 7	SA 7
7	ID 23	PF 8	ID 6	SA 6
8	ID 22	PF 7	ID 5	SA 5
9	ID 21	PF 6	ID 4	SA 4
10	ID 20	PF 5	ID 3	SA 3
11	ID 19	PF 4	ID 2	SA 2
12	ID 18	PF 3	ID 1	SA 1
13	SRR (r)	SRR*)	RTR (x)	RTR*)
14	IDE (r)	IDE*)	IDE (d)	IDE*)
15	ID 17	PF 2	R 0	R 0*)
16	ID 16	PF 1	DLC 4	DLC 4
17	ID 15	PS 8	DLC 3	DLC 3
18	ID 14	PS 7	DLC 2	DLC 2
19	ID 13	PS 6	DLC 1	DLC 1
20	ID 12	PS 5		
21	ID 11	PS 4		
22	ID 10	PS 3		
23	ID 9	PS 2		
24	ID 8	PS 1		
25	ID 7	SA 8		
26	ID 6	SA 7		
27	ID 5	SA 6		
28	ID 4	SA 5		
29	ID 3	SA 4		
30	ID 2	SA 3		
31	ID 1	SA 2		
32	ID 0	SA 1		
33	RTR (x)	RTR*)		
34	r 1	r 1*)		
35	r 0	r 0*)		
36	DLC 4	DLC 4		
37	DLC 3	DLC 3		
38	DLC 2	DLC 2		
39	DLC 1	DLC 1		

SOF	Start of Frame Bit	R#	Reserved Bit #n according to ISO 11783
ID##	Identifier Bit #n	SA#	Source Address Bit #n according to ISO 11783
SRR	Substitute Remote Request	DP	Data Page according to ISO 11783
RTR	Remote Transmission Request Bit	PF#	PDU Format Bit #n according to ISO 11783
IDE	Identifier Extension Bit	PS#	PDU Specific Bit #n according to ISO 11783
r#	CAN Reserved Bit #n	(d)	dominant bit
DLC#	Data Length Code Bit #n	(r)	recessive bit
P#	Priority Bit #n according to ISO 11783	(x)	bit state dependent on message

*) CAN Defined Bit, Unchanged in ISO 11783.

1) Required format of proprietary 11 bit identifiers.



Key

PF	PDU Format
R	Reserved
DP	Data Page
SA	Source Address
PS	PDU Specific

Figure 2 — Application of OSI Model according to ISO 11783

3.1.3 ISO 11783 support of CAN 2.0 B Standard Frame Format messages

It is recognised that controllers on the ISO 11783-specified network may support the CAN Standard Frame (11 bit identifier) message format. Though these are not compatible with the message structure according to ISO 11783, a minimum level of definition is given to accommodate the co-existence of the two formats. This minimum definition allows controllers that use this format to avoid interfering with other controllers. CAN Standard Frame Format messages are defined as being proprietary. As shown in table 1, the 11 bit Identifier field is parsed as follows: the three most significant bits are used as priority bits, the eight least significant bits identify the source address of the PDU. Priority bits are described in 3.3.1. The source address is defined in the Source Address Table [2] or is determined by the procedures outlined in the part of ISO 11783 which deals with network management [3].

NOTE — Incorrect bus arbitration could occur when two messages, one Standard Frame and one Extended Frame, access the bus at the same time. The source address (SA) has a higher relative priority in the Standard Frame messages than in the Extended Frame messages. The message with the 11 bit identifier (Standard Frame) could have a SA indicating a higher priority than the reserved bit, data page bit and PDU Format of the 29 bit identifier (Extended Frame) message. The three priority bits should be used to achieve the correct bus arbitration.

ISO 11783 only specifies a full strategy for standardized communications using the extended frame format. Hardware conforming to CAN 2.0 A specification must not be used on the network, since these versions of hardware do not allow the Extended Frame messages to be communicated.

3.2 Protocol Data Unit (PDU)

The Applications and/or Network Layer provides a string of information that is assimilated into a protocol data unit. The protocol data unit provides a framework for organizing the information that is key to each CAN Data Frame that is sent. The protocol data unit (PDU) according to ISO 11783 shall consist of seven fields. These fields are Priority, Reserved, Data Page, PDU Format, PDU Specific (which can be a destination address, Group Extension or proprietary), Source Address and Data. These fields are then packaged into one or more CAN data frames and sent over the physical media to other network controllers. There is only one PDU per CAN data frame.

NOTE — It should be recognised that some Parameter Group Number definitions will require more than one CAN data frame to send the corresponding data.

Some of the CAN Data Frame fields have been left out of the PDU definition because they are controlled entirely by the CAN specification and are invisible to all of the OSI layers above the Data Link Layer. They include the SOF, SRR, IDE, RTR, parts of the control field, CRC field, ACK field and the EOF field. These fields are defined by the CAN protocol definition and were not modified for ISO 11783. The seven PDU fields are illustrated in figure 3. Each of the fields within the PDU are defined in 3.2.1 to 3.2.7.

Table 2 — Parameter Group Number (PGN) examples

PGN constituent components					**PGN**		**Number of assignable PGs**	**Cumulative number of PGs**	**ISO or manufacturer assigned**
PGN (msb) Byte 1			PGN Byte 2	PGN Byte 3	DEC	HEX			
Bits8-3	R Bit 2	DP Bit 1	PF	PS					
0	0	0	0	0	0	000000	—	—	ISO
0	0	0	238	0	60928	00EE00	239	239	—
0	0	0	239	0	61184	00EF00	1	240	Manufacturer
0	0	0	240	0	61440	00F000	—	—	ISO
0	0	0	254	255	65279	00FEFF	3 840	4 080	—
0	0	0	255	0	65280	00FF00	—	—	Manufacturer
0	0	0	255	255	65535	00FFFF	256	4 336	—
0	0	1	0	0	65536	010000	—	—	ISO
0	0	1	239	0	126720	01EF00	240	4 576	—
0	0	1	240	0	126976	01F000	—	—	ISO
0	0	1	255	255	131071	01FFFF	4 096	8 672	—

	Priority,	R,	DP,	PF,	PS,	SA,	Data Field
No. of Bits	...3...,	.1.,	.1.,	...8...,	...8...,	...8...,	...64...

Key:
- R Reserved
- DP Data Page
- PF PDU Format
- PS PDU Specific
- SA Source Address

Figure 3 — Protocol Data Unit (PDU)

3.2.1 Priority (P)

Priority bits are used to optimize message latency for transmission onto the bus only. They should be globally masked off by the receiver (ignored). The priority of any message can be set from highest, 0 (000_2), to lowest, 7 (111_2). The default for all control oriented messages is 3 (011_2). The default for all other informational, proprietary, request and NACK messages is 6 (110_2). This permits the priority to be raised or lowered in the future as new PGNs values are assigned and bus traffic changes. A recommended priority is assigned to each PGN when it is added to the application layer documents.

3.2.2 Reserved bit (R)

This bit is currently reserved for use in a future International Standard. This reserved bit should not be confused with the CAN reserved bits. All messages should set the ISO-specified reserved bit to ZERO on transmit. Future definitions might possibly be expanding the PDU Format field, defining new PDU formats, expanding the Priority field, or increasing the address space.

3.2.3 Data Page (DP)

The data page bit sets an auxiliary page of Parameter Group descriptions. Assignment of all Parameter Group Numbers available in page zero are complete (filled) before the page one assignments are made (see 3.9 which discusses PGN Assignments).

3.2.4 PDU Format (PF)

The PDU Format is an 8-bit field that determines the PDU format and is one of the fields used to determine the Parameter Group Number assigned to the CAN data field. Parameter Group Numbers are used to identify or label commands, data, some requests, acknowledgements and negative-acknowledgements. Parameter Group Numbers identify or label information that may require one or more CAN data frames to communicate the information. If there is more information than will fit in 8 data bytes, a multi-packet message needs to be sent. A Parameter Group Number can represent one or more parameters, where a parameter is a piece of data such as engine rotations per minute. Even though a Parameter Group Number label can be used for one parameter, it is recommended that multiple parameters be grouped so that all 8 bytes of the data field are used.

The definition of two proprietary Parameter Group Numbers has been established allowing both PDU1 and PDU2 Formats to be used. The interpretation of the proprietary information will vary by manufacturer. For example, even though two different engines may use the same source address, manufacturer "A's" proprietary communications will most likely be different from manufacturer "B's".

3.2.5 PDU Specific (PS)

The PDU Specific field is an 8-bit field and its definition depends on the PDU format. Depending on the PDU format, it can be a Destination Address or a Group Extension. See table 3.

Table 3 — Definition of a PDU Specific field

Format	PDU Format field	PDU Specific Field
PDU1	0 - 239	Destination Address
PDU2	240 - 255	Group Extension

3.2.5.1 Destination Address (DA)

This field defines the specific address to which the message is being sent. Any other controller should ignore this message. The global Destination Address (255) requires all controllers to listen and respond accordingly as message recipients.

3.2.5.2 Group Extension (GE)

The Group Extension field, in conjunction with the four least significant bits of the PDU Format field, provide for 4 096 Parameter Groups per data page. These 4 096 Parameter Groups are only available using the Group Extension Format PDU. In addition, 240 Parameter Groups are provided in each data page for use only in the Destination Specific Format PDU. In total, 8 672 Parameter Groups are available to be defined using the two data pages currently available.

NOTE — When the four most significant bits of the PDU Format field are set it indicates that the PS field is a Group Extension.

The total number of Parameter Group labels available can be calculated as follows:

$$[240 + (16 \times 256)] \times 2 = 8\,672$$

where

240 = number of PDU Format field values available per data page (i.e. PDU1 Format);

16 = PDU Format values per Group Extension value (i.e. PDU2 Format only);

256 = number of possible Group Extension values (i.e. PDU2 Format only);

2 = number of Data Page states (both PDU Formats).

3.2.6 Source Address (SA)

The Source Address field is 8 bits long. There shall only be one controller on the network with a given source address. Therefore, the source address field assures that the CAN identifier is unique, as required by CAN. Address management and allocation and procedures to prevent duplication of source addresses are defined in a future International Standard [3].

3.2.7 Data Field

3.2.7.1 Data from 0 to 8 bytes

When eight or less bytes of data are required for expressing a given Parameter Group, then all eight data bytes of the CAN data frame can be used. It is generally recommended that eight bytes be allocated or reserved for all Parameter Group Number assignments which are likely to expand in the future. This provides a means to add parameters easily and not be incompatible with previous revisions which only define part of the data field. Once the number of bytes of data associated with a Parameter Group Number is specified, it cannot be changed (nor can it become multi-packet, unless originally defined as multi-packet). It is important to note that a given group function (see 3.4.5) must use the same data field length because the CAN identifier is always identical, while the CAN data field is used to convey the specific group subfunctions. These group functions require many different interpretations based on the CAN data field.

3.2.7.2 Data from 9 to 1785 bytes

When 9 to 1785 data bytes are needed to express a given Parameter Group, the communication of this data is done in multiple CAN Data Frames. Thus the term "multi-packet" is used to describe this type of Parameter Group Number. The ”Transport Protocol Function” is used when more than one CAN data frame is required to send a particular Parameter Group. The "Transport Protocol Function’s Connection Management" capability is used to set up and close out the communication of the multi-packet Parameter Groups. The Transport Protocol Data Transfer capability is used to communicate the data itself in a series of CAN Data Frames (packets) containing the packetized data. Additionally, the Transport Protocol Function provides flow control and handshaking capabilities for destination-specific transfers (see 3.10).

All CAN data frames associated with a particular multi-packet response are required to have a DLC of 8. All unused data bytes are set to "not available". The number of bytes per packet is fixed; however, ISO 11783 defines multi-packet messages that have a variable or a fixed number of packets. The Parameter Group Number for active diagnostic codes is an example of a multi-packet message that has a "variable" number of packets. Parameter Groups that are defined as multi-packet only use the transport protocol when the number of data bytes to send exceeds eight.

3.3 Protocol Data Unit (PDU) formats

The available PDU formats are illustrated in figure 4. Two PDU formats are defined; PDU1 Format (PS = Destination Address) and PDU2 Format (PS = Group Extension). The PDU1 Format allows for direction of the CAN Data Frame to a specific destination address (controller). The PDU2 Format can only communicate CAN Data Frames that are not destination-specific. The creation of two separate PDU formats was established in order to provide more possible Parameter Group Number combinations while still providing for destination-specific communications. Proprietary Parameter Group definitions have been assigned so that both PDU formats are available for proprietary communications. A standardized method was established for proprietary communications to prevent possible conflicts in identifier usage.

The definition of two proprietary Parameter Group Numbers was established which allows both PDU1 and PDU2 Formats to be used. The interpretation of the proprietary information will vary by manufacturer. For example, engine manufacturer "A's" proprietary communications will likely be different from engine manufacturer "B's" even though they both use the same source address.

3.3.1 PDU1 Format

The PDU1 format allows for applicable Parameter Groups to be sent to either a specific or global destination(s). The PDU Specific (PS) field contains a destination address (DA). PDU1 Format messages can be requested or sent as unsolicited messages.

PDU1 Format messages are determined by the PDU Format (PF) field. When the PDU Format field value is 0 to 239 the message is PDU1 Format. The Format of the PDU1 message is illustrated in figure 4.

Parameter Groups requiring a destination (PDU1) and minimal latency start at PF = 0 and increment toward x (or x1) (see table 5).

Parameter Groups requiring a destination where latency is not critical will start at PF = 239 and decrement toward x (or x1) (see table 5).

A PF equal to 239 (reserved bit = 0 and data page bit = 0) is assigned for proprietary use. In this case the PDU Specific (PS) field is a destination address (see 3.4.5). The PGN for Proprietary A is 61184.

PDU 1

	Priority,	R,	DP,	PF,	PS(DA),	SA,	Data Field
No. of Bits	...3...,	.1.,	.1.,	...8...,	...8...,	...8...,	...64...

PDU 2

	Priority,	R,	DP,	PF,	PS(GE),	SA,	Data Field
No. of Bits	...3...,	.1.,	.1.,	...8...,	...8...,	...8...,	...64...

Key:
R Reserved
DP Data Page
PF PDU Format
PS PDU Specific
DA Destination Address
GE Group Extension
SA Source Address

Figure 4 — Available PDU Formats

3.3.2 PDU2 Format

This format can only be used to communicate Parameter Groups as global messages. PDU2 Format messages can be requested or sent as unsolicited messages. Selection of PDU2 Format, at the time a PGN is assigned, prevents that PGN from ever being able to be directed to a specific destination. The PDU Specific (PS) field contains a Group Extension (GE).

PDU2 Format messages are defined to be those where the PDU Format (PF) value is equal to 240 to 255 (see table 3). The format of the PDU2 message is illustrated in figure 4.

The Parameter Group Number of messages that are sent at fast update rates (generally less than 100 ms) start at PF = 240 and increment towards y (or y1) (see table 5).

The Parameter Group Number of messages that are only requested, sent on change, or are sent at slow update rates (generally greater than 100 ms) start at PF = 254 and decrement toward y (or y1) (see table 5).

A PF equal to 255 (reserved bit = 0 and data page bit = 0) is assigned for proprietary use. The PDU Specific field is left to be defined and used by each manufacturer (see 3.4.5). The PGN for Proprietary B is 65280 to 65535.

3.4 Message types

There are five message types currently supported. These types are: Commands, Requests, Broadcasts/Responses, Acknowledgement and Group Functions. The specific message type is recognised by its assigned Parameter Group Number [2]. The RTR bit (defined in the CAN protocol for remote frames) is not to be used in the recessive state (logical 1). Therefore, Remote Transmission Requests (RTR =1) is not available for use according to ISO 11783.

3.4.1 Command

The Command message type categorises those Parameter Groups that command a specific or global destination from a source. The destination is then supposed to take specific actions based on the reception of this command message type. Both PDU1 (PS = Destination Address) and PDU2 Format (PS = Group Extension) messages can be used for commands. Example command modes may include: "Transmission Control", "Address Request", "Torque/Speed Control", etc.

3.4.2 Request

The Request message type, identified by the PGN, provides the capability to request information globally or from a specific destination. Requests specific to one destination are known as destination-specific requests. The information below assigns a Parameter Group Number to the "Request PGN" Parameter Group. The information is in the same format as Parameter Group definitions given in ISO 11783.

Parameter Group Name:	REQUEST PGN
Definition:	Used to request a Parameter Group from a network controller.
Transmission Repetition Rate:	Per User requirements, generally recommended that requests occur no more than 2 or 3 times per second.
Data Length:	3 bytes
Data Page:	0
PDU Format:	234
PDU Specific field:	Destination Address
Default Priority:	6
Parameter Group Number:	59904 ($00EA00_{16}$)
Byte 1, 2, 3:	Parameter Group Number being requested (see 3.1.2)

Table 4 gives two examples of how the Request PGN is used.

Table 4 — Request PGN examples

Message type	Use of the specified fields in PDU1 Format according to ISO 11783					
	PF	PS (DA)	SA	Data 1	Data 2	Data 3
Global Request	234	255 (Responders)	SA 1 (Requester)	PGN lsb[1)]	PGN	PGN msb[1)]
Specific Request	234	SA 2 (Responder)	SA 1 (Requester)	PGN lsb[1)]	PGN	PGN msb[1)]
1) Parameter Group Number in the data field is used to identify the information being requested.						

A response is always required from a specified destination (not global), even if it is a NACK indicating that the particular PGN value is not supported. A global request shall not be responded to with a NACK when a particular PGN is not supported by a controller.

NOTE — Some PGNs are multi-packet, so several CAN Data Frames can occur for a single request.

3.4.3 Broadcast/Response

The Broadcast/Response message type can be an unsolicited broadcast of information from a controller or it can be the response to a Command or a Request.

3.4.4 Acknowledgement

There are two forms of acknowledgement available. The first form is provided for by the CAN protocol. It consists of an "in-frame" acknowledgement which confirms that a message is received by at least one controller. In addition, messages are further acknowledged by the absence of CAN error frames. Their absence acknowledges that all other powered and connected controllers received the message correctly.

The second form of acknowledgement, which is in the data field, is a response of a "normal broadcast" or "ACK" or "NACK" to a specific command or request as provided for by an application layer. The type of acknowledgement required for some Parameter Groups is defined in the applications layer.

Parameter Group Name:	ACKNOWLEDGEMENT
Definition:	The Acknowledgement PG is used to provide a handshake mechanism between transmitting and receiving controllers.
Transmission Repetition Rate:	Upon reception of a Parameter Group Number that requires this form of acknowledgement.
Data Length:	8 bytes
Data Page:	0
PDU Format:	232
PDU Specific field:	Destination Address = Global (255)[1)]
Default Priority:	6
Parameter Group Number:	59392 ($00E800_{16}$)

Data ranges for parameters used by this Message Type:

Control byte:

0 and 1	See definition below
2 to 255	Reserved for assignment in a future International Standard

1) The global destination address makes it possible to filter on one CAN identifier for all acknowledgement messages

Positive Acknowledgement:

Control byte:		0
Byte:	1	Control byte = 0, Positive Acknowledgement (ACK)
	2-5	Reserved for assignment in a future International Standard, send each of these bytes as "FF_{16}"
	6-8	Parameter Group Number of requested information

Negative Acknowledgement:

Control byte:		1
Byte:	1	Control byte = 1, Negative Acknowledgement (NACK)
	2-5	Reserved for assignment in a future International Standard, send each of these bytes as "FF_{16}"
	6-8	Parameter Group Number of requested information

3.4.5 Group Function

The Group Function message type is used for groups of special functions (e.g. proprietary functions, network management functions, multi-packet transport functions etc.). Each group of functions is recognised by its assigned PGN. The function itself is defined within the data structure. More detailed explanation of the group functions proprietary and transport protocol will be given in the following subclauses. The proprietary group function provides a means to transmit proprietary messages in a way that eliminates CAN Identifier usage conflicts between different manufacturers. It also provides a means for receiving and distinguishing proprietary messages for use when desired. Group Functions may need to provide their own request, ACK and/or NACK mechanisms.

A request using PGN 59904 (see 3.4.2) can be used to find out if specific Parameter Group of the message type, Group Function, is supported. If it is supported then the responding controller will send the Acknowledgement PGN with the control byte equal to zero, for Positive Acknowledgement. If it is not supported, the responding controller will send the Acknowledgement PGN with the control byte set to one, for Negative Acknowledgement. The remaining portions of the ISO 11783-specified PDU format and Parameter Group must be filled in appropriately (see 3.4.4)

Parameter Group Name:	PROPRIETARY A
Definition:	This proprietary PG uses the Destination Specific PDU Format allowing manufacturers to direct their proprietary communications to a specific destination controller. How the data field of this message is used is up to each manufacturer. Use of proprietary messages is at the manufacturers discretion with the constraint that significant percentages (2 % or more) of system network utilization must be avoided.
Transmission Repetition Rate:	Per user requirements
Data Length:	0 to 8 bytes
Data Page:	0
PDU Format:	239
PDU Specific field:	Destination Address
Default Priority:	6
Parameter Group Number:	61184 ($00EF00_{16}$)
Byte: 1-8	Manufacturer-specific use

Data ranges for parameters used by this Group Function: None defined by ISO 11783

Parameter Group Name:	PROPRIETARY B
Definition:	This proprietary PG uses the PDU2 Format message allowing manufacturers to define the PS (GE) field content as they desire. However, significant percentages (2 % or more) of system network utilization must be avoided. How the PS (GE) and data fields of this message are used is up to each manufacturer. The data length of these messages has been left up to each manufacturer. Therefore, two manufacturers may use the same GE value and it

	may very well have a different data length code. Receivers of this information would need to differentiate between the two manufacturers.
Transmission Repetition Rate:	Per user requirements
Data Length:	0 to 8 bytes
Data Page:	0
PDU Format:	255
PDU Specific field:	Group Extension (manufacturer assigned)
Default Priority:	6
Parameter Group Number:	65280 to 65535 ($00FF00_{16}$ to $00FFFF_{16}$)
Byte: 1-8	Manufacturer-defined usage
Data ranges for parameters used by this Group Function:	Manufacturer-defined usage will result in the Data Length Code being unique per component supplier and source address. Caution should be used when using the Proprietary B Parameter Group (PGN = 65280).

3.5 Message priority

The CAN data frame priority shall be according to CAN 2.0B. The value within the CAN identifier field determines the message priority. A low value (0) has a high priority, while the largest CAN identifier has the lowest priority. The assignment are identified in the application layer following the guidelines in 3.9 and as defined in [2].

3.6 Bus access

When the bus is free, any controller may begin transmitting a frame. If two or more controllers start to transmit frames at the same time, the bus access conflict is resolved by contention-based arbitration using the CAN Data Frame Identifier. The mechanism of arbitration ensures that neither information nor time is lost. The transmitter with the frame of the highest priority gains bus access.

3.7 Contention-based arbitration

During arbitration, every transmitter compares the level of the bit transmitted with the level that is monitored on the bus. If these levels are equal, the controller may continue to send. When a "recessive" level is sent and a "dominant" level is monitored, that controller has lost arbitration and must withdraw without sending one more bit. When a "dominant" level is sent and a "recessive" level is monitored that controller detects a bit error.

3.8 Error detection

The following measures are provided for detecting errors:

— monitoring (transmitters compare the bit levels to be transmitted with the bit level detected on the bus);

— bit Cyclic Redundancy Check (CRC);

— variable bit stuffing with a stuff width of 5;

— frame format check.

More detailed explanation regarding these error detection techniques is provided in the CAN 2.0 B specification [5].

3.9 Assignment process for Source Addresses and Parameter Group Numbers

The protocol data units that are available provide for two different formats, PDU1 and PDU2. Parameter Groups are assigned specifically to use either PDU1 format or PDU2 format. Once assigned to a format the other is not available for that Parameter Group. PDU1 format must be used anytime it is necessary to direct a Parameter Group to a specific destination. The assignment of a Parameter Group should be done using the following characteristics:

priority, update rate, importance of the data in the packet to other network controllers, and length of the data associated with the Parameter Group. In order to help with this assignment process, a request form has been created which should be used when requesting new Source Addresses or Parameter Group Number assignments [2].

Table 5 provides a template for assigning Parameter Group Numbers. Note that the priority column is used to assign a default priority value for each PGN. The priority field should be programmable for each PGN value so that network tuning can be performed by an OEM if necessary. Although any PGN can be requested, requests are strongly discouraged for messages that are already periodically broadcast.

Messages should be assigned a Parameter Group Number which requires a destination only if it is a parameter intended to directly control (command) one of several specific controllers. Otherwise a Parameter Group Number should be selected without a destination so that any controller can have access to the parameters within the message.

Source Addresses are assigned in a linear fashion without concerns for message priority, update rate, or importance.

Parameter Group Numbers are assigned linearly to the various sections of table 5 based on the criteria provided on the Parameter Group Number and Source Address Request form. Note that multi-packet messages are not permitted when the repetition rate is greater than or equal to 10 Hz.

3.9.1 Address assignment criteria

The number of unassigned addresses according to ISO 11783 is limited and new address assignments must be made efficiently. The maximum number of addresses assigned within the system cannot exceed 256. Additions to the address definitions therefore must be limited to units that provide specific functions within the tractor or implement. Examples of specific functions include the currently defined addresses for engine, transmission, brakes and the fuel system. Functions proposed for new address assignments within the standard should have a scope similar to currently defined addresses and should be useful to most ISO 11783 users.

3.9.2 Parameter Group assignment criteria

The number of available unassigned Parameter Groups according to ISO 11783 is limited when compared to the large number that might be proposed for forestry or agricultural applications. The need for large numbers of Parameter Groups is alleviated by features provided by ISO 11783. Three primary communication methods are provided by ISO 11783 and appropriate use of each type allows effective use of the available Parameter Groups. The three communications methods are:

— PDU1 Format (PS = Destination Address allowing destination specific communications);

— PDU2 Format (PS = Group Extension);

— Proprietary Communications using two pre-defined proprietary Parameter Group Numbers.

Each of the communications methods has an appropriate use. Destination specific Parameter Groups are needed where the same message must be directed to one or another destination. A torque control message is defined in ISO 11783 which may be sent to an engine. In the case of more than one engine, this message must be sent only to the desired engine and a destination specific Parameter Group is needed and has been assigned. PDU2 Format communications apply in several situations, including:

— messages sent from a single or multiple sources to a single destination;

— messages sent from a single or multiple sources to multiple destinations.

PDU2 communications cannot be used where a message must be sent to one or another destination and not to both.

Table 5 —Parameter Group Number template according to ISO 11783

P	DP	PF	PS	Parameter Group Definition	Multipacket	PGN
	0	0	DA	PDU1 Format — less than 100 ms	NA	000
	0	1	DA		│	256
	...			↓	│	
	Boundary x			————————		
	...			↑	│	
	0	238	DA	PDU1 Format — greater than 100 ms	Allowed	60928
	0	239	DA	Proprietary	—	61184
	0	240	0	PDU2 Format — less than 100 ms	NA	61440
	0	240	0		│	61441
	...			↓	│	
	Boundary y			————————		
	...			↑		
	0	254	254		│	65278
	0	254	255	PDU2 Format — greater than 100 ms	Yes	65279
	0	255	un	PDU2 Format — Proprietary	—	65280 - 65535
	1	0	DA	PDU1 Format — less than 100 ms	NA	65536
	1	1	DA		│	65792
	...			↓	│	
	Boundary x1			————————		
	...			↑		
	1	238	DA	PDU1 Format — greater than 100 ms		126464
	1	239	DA	PDU1 Format — greater than 100 ms	Allowed	126720
	1	240	0	PDU2 Format — less than 100 ms	NA	126976
	1	240	1		│	126977
	...			↓	│	
	Boundary y1			————————		
	...			↑		
	1	255	253		│	
	1	255	254	PDU2 Format — greater than 100 ms	Allowed	131070
	1	255	255	PDU2 Format — greater than 100 ms	Allowed	131071

Key

DP	Data Page (1 bit)	GE	Group Extension (8 bits)
PF	PDU Format (8 bits)	P	Priority
PS	PDU Specific Field (8 bits)	NA	Not allowed
DA	Destination Address (8 bits)	un	Undefined
PGN	Parameter Group Number (3 bytes)		

The third communications method defined in ISO 11783, proprietary communications, is provided by the use of the proprietary Parameter Group Number. A Parameter Group Number has been assigned for non-destination specific proprietary communications and a Parameter Group Number has been assigned for destination specific proprietary communications. This allows for two functions:

a) a specific source can send its proprietary message in a PDU2 Format (non-destination specific) with the PS field identified as desired by the user, or

b) use PDU1 Format (destination specific) allowing for situations where a diagnostic controller must direct its communication to a specific destination out of a possible group of controllers. (For instance, this case arises when an engine uses more than one controller but wants to be able to perform diagnostics while all of its controllers are connected to the same network. In this case the proprietary protocol needs to be able to be destination-specific.)

Proprietary communications are useful in two situations:

— when it is unnecessary to have standardized communications;

— when it is important to communicate proprietary information.

Much of the communications between controllers constructed by a single manufacturer do not require standardization. The information that is communicated is not generally useful to other controllers on the network. In this situation the proprietary Parameter Group can be used.

When Parameter Group assignment is contemplated, proprietary and then PDU2 Format communications methods should be considered. If proprietary information is being communicated, the proprietary method should be used. If the information to be communicated is not of general interest, proprietary methods should be used. If the information is of general interest and does not require direction of the message to a particular controller, a PDU2 Format assignment should be sought. Finally, if the information is of general interest but requires direction to one or another controllers, then destination specific addressing is needed and a destination PDU1 Format Parameter Group assignment should be sought.

3.9.3 Data field definition

Minimising message overhead with CAN based systems requires full use of the data field (all 8 bytes). Except in the case of very time critical messages, related parameters should be grouped to fill the 8 byte data field. Following this principal conserves Parameter Group Numbers for future assignment. Strong justification is needed to allow definition of Parameter Groups that result in sparsely used data fields.

3.10 Transport protocol functions

Transport protocol functions are described as a part of the data link layer with the recognition that Transport protocol functionality is subdivided into two major functions: Message Packetization and Reassembly and Connection Management. They are described in 3.10.1 and 3.10.2.

3.10.1 Packetization And Reassembly

Messages greater than eight bytes in length are too large to fit into a single CAN Data Frame. They must therefore be broken into several smaller packets, and those packets transmitted in separate message frames. At the destination end, the individual message frames must be received, parsed, and the original message reassembled from the received packets.

Only one multi-packet, global destination message can be sent from a given controller at a given time. However, receiving controllers must recognise that multiple multi-packet messages can be received interspersed with one another from different sources.

3.10.1.1 Message packets

The CAN Data Frame includes an eight byte data field. Because the individual packets which comprise a large message must be identified individually so that they may be reassembled correctly, the first byte of the data field is defined as the sequence number of the packet.

Individual message packets are assigned a sequence number of one to 255. This yields a maximum message size of (255 packets $\times$ 7 bytes/packet) = 1 785 bytes.

3.10.1.2 Sequence numbers

Sequence numbers are assigned to packets for transmission on the network during message packetization and then used on reception of packets to reassemble them back into a message.

Sequence numbers shall be assigned to individual packets beginning with one and continuing sequentially until the entire message has been packetized and transmitted. The packets shall be sent sequentially in ascending order starting with packet 1.

3.10.1.3 Packetization

A multi-packet message is defined as a message packet whose data will not fit into the data field of a single CAN Data Frame (i.e. messages with a data field greater than eight bytes).

For the purposes of this protocol, the large message is considered to be a Parameter Group that has associated with it a string of 9 or more bytes. The first Data Transfer Packet contains the sequence number 1 and the first seven bytes of the string. The second seven bytes are placed into another data frame along with the sequence number 2, the third with sequence number 3, and so on until all the bytes in the original message have been placed into ISO 11783-specified CAN data frames and transmitted.

The time between packets for Multi-packet Broadcast messages shall be 50 ms to 200 ms (see 3.12.3). For multi-packet messages directed to a specific destination, the transmitting controller will maintain a maximum time between packets (where CTS allows more than one) of 200 ms. Receivers must be aware that the packets containing the data all have the same identifier.

Each Data Transfer packet (other than the last packet in a transmission sequence) shall include seven bytes of the original large message. The final packet will include a data field of eight bytes: the sequence number of the packet and at least one byte of data related to the Parameter Group and then any remaining unused bytes set to "FF_{16}".

3.10.1.4 Reassembly

Data packets are received sequentially. Each data packet of a multi-packet message shall be assembled, in order of sequence number, into a single string of bytes. This string of bytes is passed to the application entity responsible for the large message.

3.10.2 Connection management

Connection Management is concerned with the opening, use, and closure of virtual connections between controllers for destination-specific transfers. A virtual connection in the ISO 11783-specified environment may be considered to be a temporary association of two controllers for the purpose of transferring a single large message that is described by a single Parameter Group Number (see figures B.1 and B.2). In cases where the connection is from one to many, there is no flow control or closure provided (see figure B.3).

3.10.2.1 Multi-Packet broadcast

Multi-packet messages may be non-destination-specific; that is, they may be broadcast messages. To broadcast a multi-packet message, a controller first transmits a Broadcast Announce Message (BAM). This message, which must be transmitted to the global destination address, constitutes a large message warning to the controllers on the network. The BAM message contains the Parameter Group Number of the large message to be broadcast; its size and the number of packets into which it has been packeted. Controllers interested in the broadcast data are then required to allocate the resources necessary to receive and reassemble the message. The Data Transfer PGN (PGN = 60160) is then used to send the associated data.

3.10.2.2 Connection initiation

A connection is initiated when a controller transmits a Request to Send message to a destination address. The Request to Send message contains the size of the entire message in bytes, the number of separate messages in which it will be transferred, and its assigned Parameter Group Number.

Upon receipt of a Request to Send message, a controller may elect to accept the connection or to reject it. To accept the connection, the receiving controller transmits a Clear to Send message. The Clear to Send message contains the number of message packets it may accept, and the sequence number of the first packet it is expecting. The receiving controller must ensure that it has sufficient resources to handle the number of packets of which it is accepting delivery. The sequence number of the packet, in the instance of a freshly-opened connection, would be one.

NOTE — The Clear to Send message may not include provision for all the component frames of the message.

To reject the connection, the controller responds with a Connection Abort message. The connection may be rejected for any reason, although lack of resources, memory, etc., will likely be the cause.

3.10.2.3 Data transfer

Data transfer begins after the originator of the connection receives the Clear to Send message. An exception is if the Data Transfer was the result of the ”Broadcast Announce Message”; in this case the Clear to Send message is not used. The PGN for Data Transfer contains the CAN identifier field of each packet. The first byte of the data field contains the sequence number of the packet.

In the case of destination-specific messages, the receiving controller is responsible for co-ordinating flow control between the controllers. If the receiving controller wants to stop data flow momentarily, while a connection is open, it must used the Clear to Send message, setting the number of packets to send equal to zero. In the case where the flow must be stopped for several seconds, the receiving controller must repeat the transmission of the Clear to Send message once per 0,5 s to assure the transmitter the connection is not broken. All remaining bit fields are set to 1 ("Don’t care").

3.10.2.4 Connection closure

Two connection closure cases exist in the absence of errors. The first is when there is a global destination and the second is when there is a specific destination. In the case of the global destination no connection closure operation will be performed beyond the reception of the data itself (see 3.10.3 and 3.10.3.4). In the case of destination-specific transfer and upon receipt of the last packet in the message stream, the receiving controller will transmit an end-of-message acknowledgement to the originator of the message. This is the signal to the originator that the connection is considered closed by the receiver. The ”end-of-message ack” closure is required to free the connection for subsequent use by other controllers.

The "connection abort” message is not allowed to be used by receiving controllers in the case of a global destination (see 3.10.3 and 3.10.3.4). In the case of a destination-specific transfer either controller can, at any time, use the ”connection abort” to terminate the connection. If the receiving controller should, for example, determine that there are no resources available for processing the message, it may simply abort the connection by issuing the connection abort message. Upon receipt of the connection abort message, any message packets already passed must be abandoned.

A failure of either controller may also cause closure of a connection due to a time-out. For example,

— a delay of more than 250 ms (T1) from receipt of the last packet when more were expected (CTS allowed more),

— a delay of more than 1250 ms (T2) after a CTS was transmitted (originator failure),

— a lack of CTS or ACK for more than 1250 ms (T3) after the last packet was transmitted (receiver failure),

— a lack of CTS for more than 550 ms (T4) after CTS (0) message to ”hold the connection open”,

will cause a connection closure to occur (see 3.12.3 and figure B.1 regarding time-outs).

3.10.3 Transport Protocol — Connection Management messages

Connection Management Messages are used to initiate and close connections and also to control flow. Transport protocol provides the following five transport protocol connection management messages: the Connection Mode Request to Send, the Connection Mode Clear to Send, the End of Message Acknowledgement, the Connection

Abort, and the Broadcast Announce Message. The format of these messages are shown below in the Parameter Group definition for ”Transport Protocol - Connection Management”.

Parameter Group Name:	TRANSPORT PROTOCOL - CONNECTION MANAGEMENT (TP.CM)
Definition:	Used for the transfer of Parameter Groups that have 9 or more bytes of data. A definition of each specific message defined as part of the transport protocol is contained in the following section .
Transmission Repetition Rate:	Per the Parameter Group Number to be transferred
Data Length:	8 bytes
Data Page:	0
PDU Format:	236
PDU Specific field:	Destination Address
Default Priority:	6
Parameter Group Number:	60416 ($00EC00_{16}$)

Data ranges for parameters used by this Group Function:

Control byte:	16,17,19,32, 255 (Bytes 0-15, 18, 20-31, 33-254 are reserved for assignment in a future International Standard.)
Total Message Size, bytes:	9 to 1785 (2 bytes) (Sizes 0 to 8 and 1786 to 65535 are not allowed.)
Total Number of Packets:	1 to 255 (1 byte), zero not allowed
Number of Packets that can be sent:	0 to 255 (1 byte)
Next Packet Number to be Sent:	1 to 255 (1 byte), zero not allowed
Sequence Number:	1 to 255 (1 byte), zero not allowed

Connection Mode Request To Send (TP.CM_RTS): Destination Specific

Byte:	1	Control byte = 16, Destination Specific Request_To_Send (RTS)
	2-3	Total Message Size, number of bytes
	4	Total number of packets
	5	Reserved for assignment in a future International Standard, this byte should be sent as ”FF_{16}”
	6-8	Parameter Group Number of packeted message

Connection Mode Clear To Send (TP.CM_CTS): Destination Specific

Byte:	1	Control byte = 17, Destination Specific Clear_To_Send (CTS)
	2	Number of packets that can be sent
	3	Next packet number to be sent
	4-5	Reserved for assignment in a future International Standard, this byte should be sent as ”FF_{16}”
	6-8	Parameter Group Number of packeted message

End of Message Acknowledgement (TP.CM_EndofMsgACK): Destination Specific

Byte:	1	Control byte = 19, End_of_Message Acknowledge
	2-3	Total Message Size, number of bytes
	4	Total number of packets
	5	Reserved for assignment in a future International Standard, this byte should be sent as ”FF_{16}”
	6-8	Parameter Group Number of packeted message

Connection Abort (TP.Conn_Abort): Destination Specific

Byte:	1	Control byte = 255, Connection Abort
	2-5	Reserved for assignment in a future International Standard, this byte should be sent as ”FF_{16}”

	6-8	Parameter Group Number of packeted message

Broadcast Announce Message (TP.CM_BAM): Global Destination

Byte:	1	Control byte = 32, Broadcast Announce Message
	2-3	Total Message Size, number of bytes
	4	Total number of packets
	5	Reserved for assignment in a future International Standard, this byte should be sent as ”FF_{16}”
	6-8	Parameter Group Number of packeted message

3.10.3.1 Connection Mode Request To Send (TP.CM_RTS)

The TP.CM_RTS message informs a controller that another controller on the network wishes to open a virtual connection with it. It consists of a message with the source address field equal to that of the originating controller, destination address set to that of the intended recipient of a large message, and the remaining fields set appropriately for the Parameter Group Number being sent.

3.10.3.2 Connection Mode Clear To Send (TP.CM_CTS)

The TP.CM_CTS message is used to respond to the Request To Send message. It informs the peer controller that it is ready for a certain amount of large message data.

3.10.3.3 End of Message Acknowledgement (TP.CM_EndofMsgACK)

The TP.CM_EndofMsgACK message is passed from the recipient of a large message to its originator indicating that the entire message was received and reassembled correctly.

3.10.3.4 Connection Abort (TP.Conn_Abort)

The TP.Conn_Abort message is used by either controller involved in a virtual connection to close the connection without completing the transfer of the message.

Upon receipt of a Connection Mode Request To Send message, a node must determine if there are sufficient resources available to deal with the message for which this connection is sought. For example, if the controller must acquire memory from the system heap, it may not be able to claim enough to accept the entire message; or a controller may simply be too occupied doing other tasks to expend processor cycles handling a large message. In these cases a Connection Abort message would be sent.

3.10.3.5 Broadcast Announce message (TP.CM_BAM)

The TP.CM_BAM is used to inform all the nodes of the network that a large message is about to be broadcast. After the transmission of this message, Data Transport messages passed from the originator of this message will contain the packetized broadcast data.

3.10.4 Transport Protocol — Data Transfer messages (TP.DT)

The TP.DT message is used to communicate the data associated with a Parameter Group. The TP.DT message is an individual packet of a multi-packet message transfer. For example, if a large message has to be divided into 5 packets in order to be communicated, then there would be five TP.DT messages. Examples of the use of TP.DT messages are given in annex B.

Parameter Group Name:	TRANSPORT PROTOCOL - DATA TRANSFER (TP.DT)
Definition:	Used for the transfer of data associated with Parameter Groups that have more than 8 bytes of data.
Transmission Repetition Rate:	Per the Parameter Group Number to be transferred

Data Length:	8 bytes
Data Page:	0
PDU Format:	235
PDU Specific field:	Destination Address
Default Priority:	6
Parameter Group Number:	60160 ($00EB00_{16}$)

Data ranges for parameters used by this Message Type:

Sequence Number:		1 to 255 (1 byte)
Byte:	1	Sequence Number
	2-8	Packetized Data (7 bytes).

NOTE — The last packet of a multi-packet Parameter Group may require less than 8 bytes. The extra bytes should be filled with FF_{16}.

3.11 PDU processing requirements

Processing of the PDUs requires specific procedures to be followed. A suggested sequence for interpreting PDUs is given in annex A. An example of the use of Parameter Groups is given in annex C.

Controllers must be able to process data link messages fast enough to prevent losing messages when the data link is at 100 % utilization. This also means that in low utilization situations when there are back-to-back messages, each controller must be able to process the messages fast enough in order not to lose messages due to their back-to-back nature. Processing the messages fast enough does not mean that a message has to be immediately generated but that a new message must not overrun the previous messages.

3.12 Application notes

3.12.1 High data rates

Data that is to be updated at a high rate and has tight latency requirements should, if possible, allow hardware-based message filtering to be used.

3.12.2 Request scheduling

The scheduling of a request should be cancelled if the information that is about to be requested is received before the request is sent. That is, if the information is received 50 ms prior to request scheduling then the request should not be issued. Parameter Group should not be requested if they are recommended to be broadcast. Exceptions may arise when the recommended broadcast time exceeds a special case need.

3.12.3 Controller response time and time out defaults

All controllers, when required to provide a response, must do so within 0,20 s. All controllers expecting a response must wait at least 1,25 s (T2 or T3) before giving up or retrying. These times ensure that any latencies due to bus access or message forwarding across bridges will not cause unwanted time-outs. Different time values can be used for specific applications when required. For example, a 20 ms response may be expected for high speed control messages. Reordering of any buffered messages may be necessary to reach a faster response. There is no restriction on minimum response time.

Time between packets of a multi-packet message directed to a specific destination is 0 ms to 200 ms. This means that back-to-back messages can occur and they may contain the same identifier. The CTS mechanism can be used to assure a given time spacing between packets (flow control). The required time interval between packets of a Multi-packet Broadcast message is 50 ms to 200 ms. A minimum time of 50 ms ensures the receiver has time to pull the message from the CAN hardware. The receiver shall use a time-out (i.e. T1) of 250 ms. This provides a time-out that is greater than the maximum transmit spacing of 200 ms.

Maximum forward delay time within a bridge is 50 ms. The total number of bridges is equal to 10 (i.e. 1 tractor + 5 trailers + 4 dollies = 10 bridges). Therefore the total network delay is 500 ms in one direction.

The number of request retries is equal to 2 (3 requests total), this applies to the CTS when re-transmitting the data packet(s).

The margin for time-outs is 50 ms.

Figures B.1 and B.3 have the timing requirements identified. In figure B.1, the time numbers are computed assuming the worst case number of bridges, that is 10 bridges. The time-out numbers for receivers are identified as a time value while transmitter requirements are specified as a less than or equal to time value.

3.12.4 Required responses

A response is required for a global request from all controllers that have the requested PG, even the requester. Acknowledgements are not allowed for global requests.

A controller which uses a global DA (255) for a request (e.g. ”address request"), shall itself send a response if it has the data requested. This is a requirement because all controllers are expected to respond. If the controller issuing the request does not respond then the other network controllers may draw the wrong conclusion about the requested information.

Annex A
(normative)

ISO 11783-specified PDU processing — Typical receive routine

RECEIVE INTERRUPT:

When a message is received by the microprocessor via the CAN IC, several tests are performed in order to parse it and determine if and where it should be stored. The three priority bits are used only for bus arbitration and therefore are not needed (used) by the receiving controller.

NOTE — A given controller may have more than one address if it performs multiple functions.

```
IF PGN = REQUEST PGN AND THE DESTINATION IS SPECIFIC  ; specific request

THEN

   IF DA = ASSIGNED ADDRESS (destination)

   THEN

      SAVE 4 BYTE ID AND 3 DATA BYTES IN REQUEST QUEUE
IF PGN = REQUEST PGN AND THE DESTINATION IS GLOBAL    ; global request

THEN

   SAVE 4 BYTE ID AND 3 DATA BYTES IN REQUEST QUEUE
IF PF < 240

THEN

   IF DA = GLOBAL                                    ; PDU1 Format (DA = global)

   THEN

      USE JUMP TABLE FOR PGN VALUES OF INTEREST AND

      IF SA = ID OF SPECIAL INTEREST

      THEN

         SAVE 8 BYTES OF DATA IN DEDICATED BUFFER

      ELSE

         SAVE 12 BYTE MESSAGE (ID AND DATA) IN CIRCULAR QUEUE
   ELSE DA = SPECIFIC                                ; PDU1 Format (DA = specific)

      USE JUMP TABLE FOR PGN VALUES OF INTEREST AND

      IF SA = ID OF SPECIAL INTEREST VALUES

      THEN

         SAVE 8 BYTES OF DATA IN DEDICATED BUFFER
```

```
      ELSE
         SAVE 12 BYTE MESSAGE (ID AND DATA) IN CIRCULAR QUEUE
IF PF > OR = 240                                     ; PDU2 Format
THEN
   USE JUMP TABLE FOR PGN VALUES OF INTEREST AND
      IF SA = ID OF SPECIAL INTEREST
      THEN
         SAVE 8 BYTES OF DATA IN DEDICATED BUFFER
      ELSE
         SAVE 12 BYTE MESSAGE (ID AND DATA) IN CIRCULAR QUEUE
```

Annex B
(normative)

Transport Protocol Transfer Sequences

Under normal circumstances, the flow model for Connection Mode data transfer follows figure B.1. A TP.CM_RTS is transferred indicating that there are 23 bytes in the packeted message which will be transferred in four packets. The PGN for the data in the transfer is 65259, component identification.

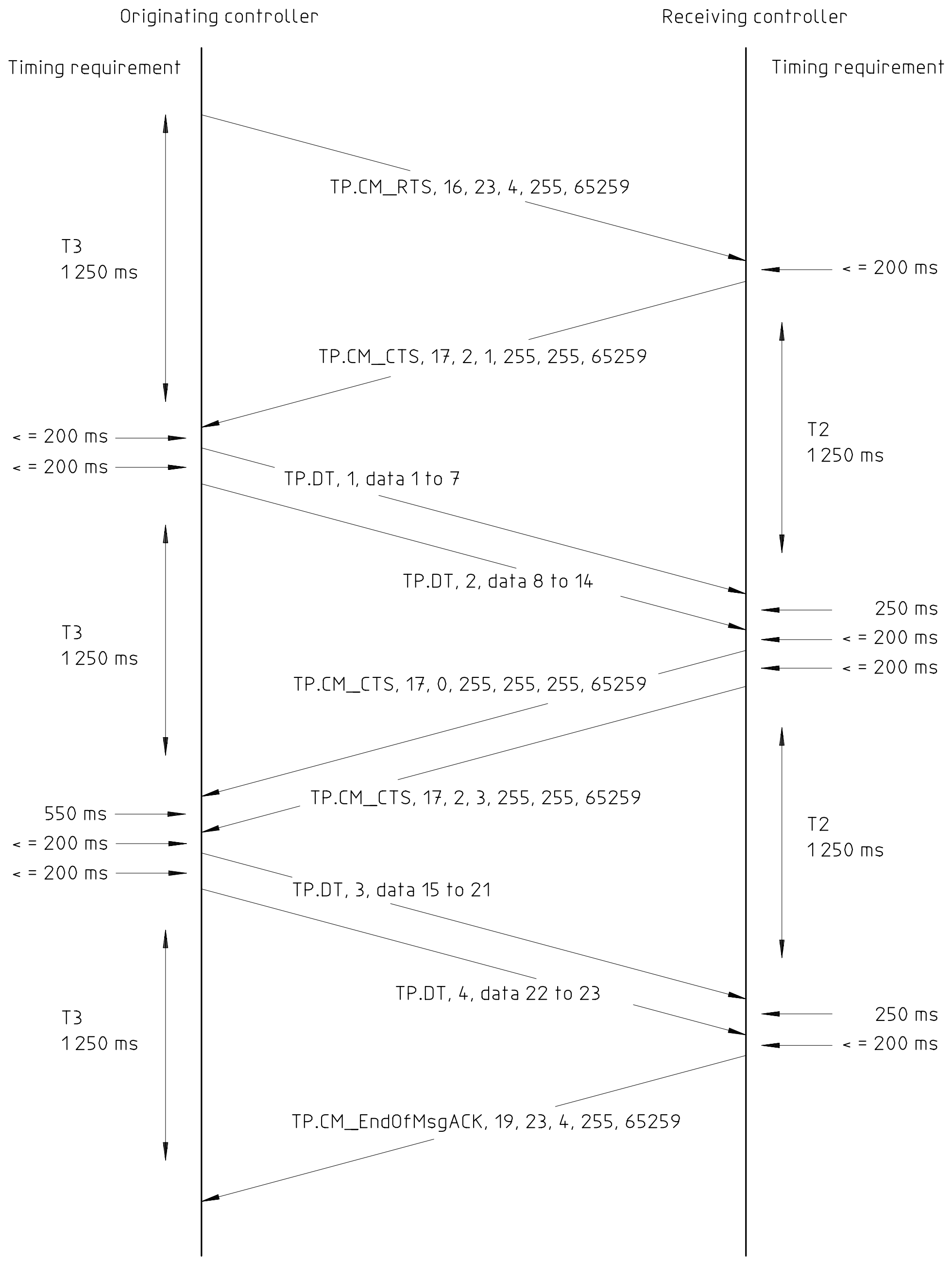
The receiving controller replies with a TP.CM_CTS indication that it is ready to process two packets, beginning with packet 1.

The originating controller passes the first two packets across the network using TP.DT. The receiving controller then issues a TP.CM_CTS indicating that it wants to hold the connection open but cannot receive any packets right now. A maximum of 500 ms later it must send another TP.CM_CTS message to hold the connection. In this example it sends another TP.CM_CTS indicating that it can take two more packets, beginning with packet 3. Once packets 3 and 4 have been transferred, the receiving controller transmits a TP.CM_EndofMsgACK message indicating that all the packets expected were transmitted and that the connection is now considered closed. Note that packet 4 contains 2 bytes of valid data, bytes 22 and 23, the remaining data characters in this packet are transmitted as 255 (FF16), data not available, such that the message is 8 bytes in length.

Message transfer in the event of an error on the link is shown in figure B.2. The TP.CM_RTS is transferred and responded to properly, then data is lost during the data transfer phase.

In this situation, the request to send is sent in the same manner as the earlier example. The first two packets are transferred, but packet 2 fails checksum, or was considered in error by the receiving controller. The receiver then transfers a TP.CM_CTS indicating that it wants a single packet, and that packet is packet 2. The originator complies, transferring packet 2. The receiver then passes a CTS indicating it wants two packets, starting with packet 3. This TP.CM_CTS is acknowledgement that packets 1 and 2 were received correctly. Once the last packet is received correctly, the receiver passes a TP.CM_EndofMsgACK signalling that the entire message has been correctly received.

In the situation shown in figure B.3, a controller indicates to the network that it is about to transfer a multi-packet message utilizing the services of the transport protocol. In this example, PGN 65260, tractor identification is being broadcast to the network. The originating controller first transmits a TP.CM_BAM (Broadcast Announce Message), followed by the data packets. No acknowledgement is performed by any of the receiving controllers.



NOTE — Time-outs (T1, T2, T3, T4) are described in 3.10.2.4.

Figure B.1 — Connection Mode Data Transfer Sequence without errors

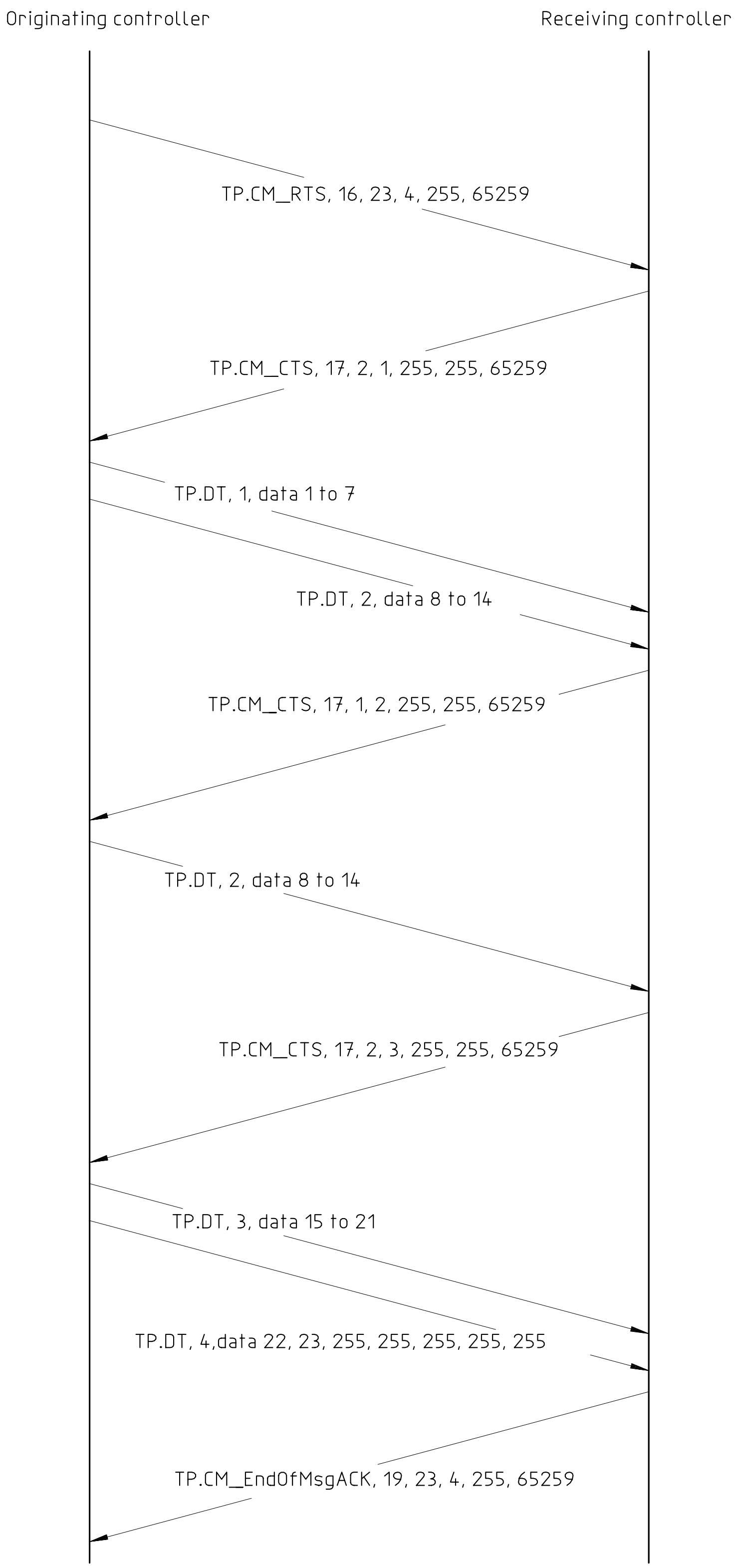


Figure B.2 — Connection Mode Data Transfer Sequence with errors

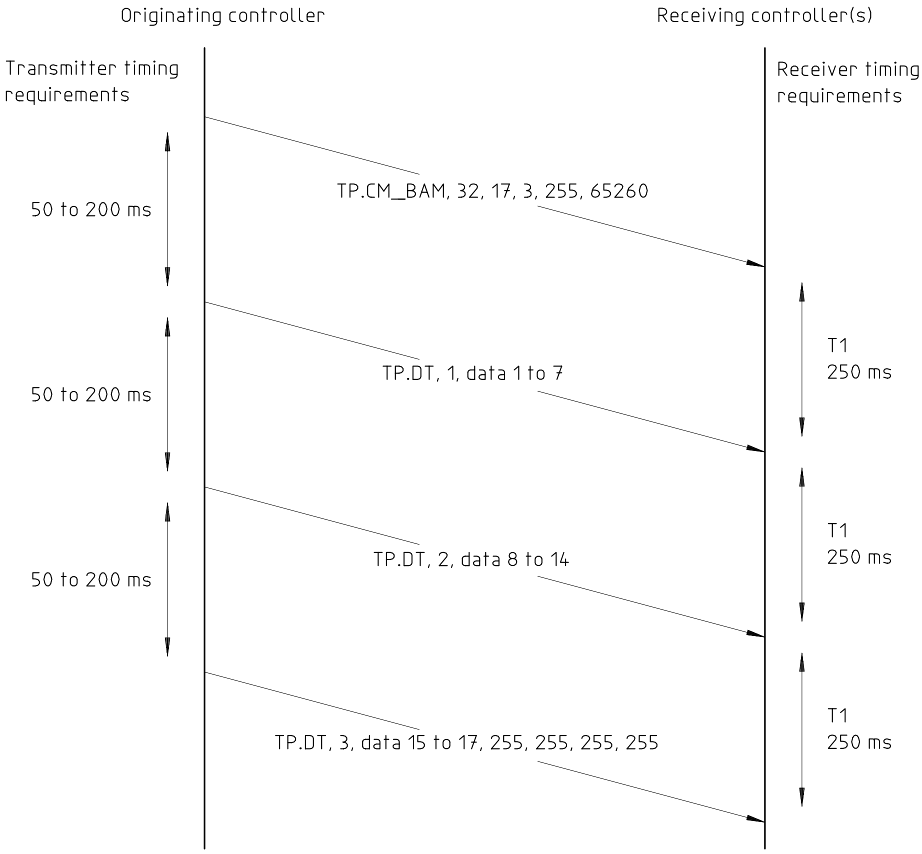


Figure B.3 — Broadcast Data Transfer Sequence

Annex C
(informative)

Communication Mode examples

This annex provides examples of how an engine would typically respond in the following cases.

1) BROADCAST/RESPONSE/ACK

Send the engine serial number (Component ID Parameter Group Number = 65259 {$00FEEB_{16}$}).

2) DESTINATION SPECIFIC REQUEST

Receive a specific request for the engine serial number. The message sent back is either a RESPONSE with the data, or a NACK (see item 4 below in the examples).

2A) GLOBAL REQUEST

Receive a global request for the engine serial number. The message sent back is a RESPONSE from a specific controller which has the data. Acknowledgements are not used on global requests.

3) COMMAND

For some commands it may be desirable to have a specific acknowledgement that the task has been completed. When this is the case a message can be sent back as either an ACK = COMMAND COMPLETE or a NACK = COMMAND NOT ABLE TO BE COMPLETED. The example below uses ”CF” as the command that will be Acknowledged with an ACK or NACK.

4) NACK

Send the NACK message to indicate that the command or request could not be acted upon (invalid request). The NACK message contains the offending Parameter Group Number (PGN) in the data field. If the Parameter Group Number in a COMMAND or REQUEST is not recognised by the destination (addressed controller), it should still NACK. If the Parameter Group Number is recognised, but the parameter(s) are not available, a normal response is sent back but with the data value(s) set to 255.

EXAMPLES

	PF	PS(DA)	SA	DATA
1) BROADCAST	254	235(GE)	000	236912
2) SPECIFIC REQUEST	234	000(DA)	003	PGN 65259
1) RESPONSE	254	235(GE)	000	236912
or				
4) NACK	232	255(DA)	000	01,255,255,255,255,PGN 65259
2A) GLOBAL REQUEST)	234	255(DA)	003	PGN 65259
1) RESPONSE	254	235(GE)	000	236912
3) COMMAND	CF	000(DA)	240	
1) ACK	232	255(DA)	000	00,255,255,255,255,PGN for CF
or				
4) NACK	232	255(DA)	000	01,255,255,255,255,PGN for CF
or OTHER[1)]				

1) COMMANDs must always have a mechanism for confirming that the action was successful or not. An ACK message is not required if another means is available. This helps to minimize bus traffic. For example, a torque command to the engine can be confirmed by looking at the torque mode bits as well as the torque value coming from the engine.

Annex D
(informative)

Bibliography

[1] ISO 11519-1:1994, *Road vehicles — Low-speed serial data communications — Part 1: General and definitions*.

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

[3] ISO 11783-5:—[1)], *Tractors, machinery for agriculture and forestry — Serial control and communications data network — Part 5: Network management*.

[4] ISO 11898:1993, *Road vehicles — Interchange of digital information — Controller area network (CAN) for high-speed communication*.

[5] CAN Specification, version 2.0 B. Robert Bosch GmbH. September 1991.

[6] SAE J 1939-21:1994, *Recommended practice for serial control and communications vehicle network — Part 21: Data link layer.*

1) To be published.

ICS 65.060.99

Descriptors: agricultural machinery, agricultural tractors, forest equipment, mobile equipment, data processing, information interchange, local area networks, controller area networks, data link layer, data transfer, data formats.

Price based on 29 pages