

# INTERNATIONAL STANDARD

## NORME INTERNATIONALE

---

**Electronic railway equipment – Train communication network (TCN) –  
Part 3-1: Multifunction Vehicle Bus (MVB)**

**Matériel électronique ferroviaire – Réseau embarqué de train (TCN) –  
Partie 3-1: Bus de Véhicule Multifonctions (MVB)**



## THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2012 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester.

If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de la CEI ou du Comité national de la CEI du pays du demandeur.

Si vous avez des questions sur le copyright de la CEI ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de la CEI de votre pays de résidence.

IEC Central Office  
3, rue de Varembé  
CH-1211 Geneva 20  
Switzerland

Tel.: +41 22 919 02 11  
Fax: +41 22 919 03 00  
[info@iec.ch](mailto:info@iec.ch)  
[www.iec.ch](http://www.iec.ch)

### About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

### About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

#### Useful links:

IEC publications search - [www.iec.ch/searchpub](http://www.iec.ch/searchpub)

The advanced search enables you to find IEC publications by a variety of criteria (reference number, text, technical committee,...).

It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - [webstore.iec.ch/justpublished](http://webstore.iec.ch/justpublished)

Stay up to date on all new IEC publications. Just Published details all new publications released. Available on-line and also once a month by email.

Electropedia - [www.electropedia.org](http://www.electropedia.org)

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary (IEV) on-line.

Customer Service Centre - [webstore.iec.ch/csc](http://webstore.iec.ch/csc)

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: [csc@iec.ch](mailto:csc@iec.ch).

### A propos de la CEI

La Commission Electrotechnique Internationale (CEI) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

### A propos des publications CEI

Le contenu technique des publications de la CEI est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

#### Liens utiles:

Recherche de publications CEI - [www.iec.ch/searchpub](http://www.iec.ch/searchpub)

La recherche avancée vous permet de trouver des publications CEI en utilisant différents critères (numéro de référence, texte, comité d'études,...).

Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

Just Published CEI - [webstore.iec.ch/justpublished](http://webstore.iec.ch/justpublished)

Restez informé sur les nouvelles publications de la CEI. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - [www.electropedia.org](http://www.electropedia.org)

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (VEI) en ligne.

Service Clients - [webstore.iec.ch/csc](http://webstore.iec.ch/csc)

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: [csc@iec.ch](mailto:csc@iec.ch).



IEC 61375-3-1

Edition 1.0 2012-06

# INTERNATIONAL STANDARD

## NORME INTERNATIONALE

**Electronic railway equipment – Train communication network (TCN) –  
Part 3-1: Multifunction Vehicle Bus (MVB)**

**Matériel électronique ferroviaire – Réseau embarqué de train (TCN) –  
Partie 3-1: Bus de Véhicule Multifonctions (MVB)**

INTERNATIONAL  
ELECTROTECHNICAL  
COMMISSION

COMMISSION  
ELECTROTECHNIQUE  
INTERNATIONALE

PRICE CODE  
CODE PRIX

**XF**

ICS 45.060

ISBN 978-2-88912-070-3

**Warning! Make sure that you obtained this publication from an authorized distributor.  
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

# CONTENTS

FOREWORD.....	8
INTRODUCTION.....	10
1 Scope.....	12
2 Normative references.....	12
3 Terms and definitions, abbreviations and conventions .....	13
3.1 Terms and definitions .....	13
3.2 Abbreviations.....	30
3.3 Conventions .....	31
3.3.1 Base of numeric values.....	31
3.3.2 Naming conventions.....	32
3.3.3 Time naming conventions .....	32
3.3.4 Procedural interface conventions .....	32
3.3.5 Specification of transmitted data .....	35
3.3.6 State diagram conventions.....	37
4 Physical Layer .....	38
4.1 Topology .....	38
4.1.1 Segments .....	38
4.1.2 Couplers.....	38
4.1.3 Double-line segments .....	39
4.2 Device classes .....	39
4.2.1 Capabilities.....	39
4.2.2 Class 0 devices .....	40
4.2.3 Class 1 devices .....	40
4.2.4 Class 2 devices .....	40
4.2.5 Class 3 devices .....	40
4.2.6 Class 4 devices .....	40
4.2.7 Class 5 devices .....	40
4.2.8 Device Attachment.....	41
4.3 Specifications common to all media .....	41
4.3.1 Signalling speed .....	41
4.3.2 Propagation delays .....	41
4.3.3 Transceiver interface .....	41
4.3.4 Redundant medium (option) .....	42
4.4 Electrical Short Distance medium (choice) .....	43
4.4.1 ESD topology.....	43
4.4.2 ESD configuration rules .....	44
4.4.3 ESD section specifications.....	45
4.4.4 ESD shielding .....	45
4.4.5 ESD medium-dependent interface.....	46
4.4.6 ESD Line_Unit specifications .....	48
4.4.7 ESD signal wave form.....	48
4.4.8 ESD transmitter .....	49
4.4.9 ESD receiver .....	50
4.5 Electrical Middle Distance medium (choice) .....	51
4.5.1 EMD topology .....	51
4.5.2 EMD configuration rules.....	51

4.5.3	EMD terminator.....	52
4.5.4	Cable section.....	52
4.5.5	EMD shielding.....	53
4.5.6	EMD medium-dependent interface .....	54
4.5.7	EMD Line_Unit specifications.....	57
4.5.8	EMD signal waveform .....	58
4.5.9	EMD transmitter specifications.....	58
4.5.10	EMD receiver specifications.....	62
4.6	Optical Glass Fibre medium (choice).....	63
4.6.1	OGF topology .....	64
4.6.2	OGF optical cable and fibre.....	64
4.6.3	OGF medium-dependent interface .....	64
4.6.4	OGF test signal (guideline) .....	65
4.6.5	OGF transmitter specifications.....	66
4.6.6	OGF receiver specifications .....	66
4.6.7	OGF active star coupler .....	67
4.6.8	OGF double-line layout (option).....	68
5	Medium-dependent signalling.....	68
5.1	Frame encoding and decoding .....	68
5.1.1	Conventions.....	68
5.1.2	Bit encoding.....	69
5.1.3	Non-data symbols .....	69
5.1.4	Start Bit .....	69
5.1.5	Start Delimiter.....	69
5.1.6	End Delimiter .....	70
5.1.7	Valid frame (definition).....	71
5.1.8	Detection of line idle .....	71
5.1.9	Detection of collision.....	71
5.1.10	Receiver behaviour in case of error.....	71
5.1.11	Jabber halt .....	71
5.2	Line redundancy (option) .....	72
5.2.1	Principle .....	72
5.2.2	Redundant transmission.....	72
5.2.3	Redundant reception.....	72
5.2.4	Switchover.....	72
5.2.5	Redundancy status report .....	73
5.3	Repeater .....	73
5.3.1	Repeater between single-line segments .....	74
5.3.2	Repeater for redundant medium (option) .....	74
6	Frames and telegrams .....	75
6.1	Frame format.....	75
6.1.1	Master Frame format .....	75
6.1.2	Slave Frame format .....	75
6.1.3	Check Sequence.....	76
6.2	Telegram timing.....	77
6.2.1	Conventions.....	77
6.2.2	Reply delay (definition).....	77
6.2.3	Frame spacing at the source .....	79
6.2.4	Frame spacing at the destination.....	79

6.2.5	Frame spacing at the master.....	80
6.3	Detection of correct frames, collision and silence by the master .....	81
6.3.1	Correct frame (definition) .....	81
6.3.2	Detection of collision by the master .....	81
6.3.3	Detection of silence by the master.....	81
7	Link Layer Control.....	81
7.1	Addressing .....	81
7.1.1	Device Address.....	81
7.1.2	Logical_Address .....	81
7.1.3	Group_Address.....	82
7.2	Master Frames contents .....	82
7.2.1	Master Frame format .....	82
7.2.2	F_code encoding .....	82
7.3	Slave Frame contents.....	83
7.3.1	Slave Frame format .....	83
7.3.2	Size error.....	84
7.4	Telegram types.....	84
7.4.1	Process Data telegram.....	84
7.4.2	Message Data.....	85
7.4.3	Supervisory Data telegrams .....	86
8	Medium allocation.....	87
8.1	Organisation.....	87
8.1.1	Turn .....	87
8.1.2	Basic Period .....	87
8.1.3	Padding .....	88
8.2	Periodic Polling.....	88
8.2.1	Periodic List.....	88
8.2.2	Individual Period .....	88
8.2.3	Periodic Phase construction.....	89
8.3	Event Polling .....	90
8.3.1	Group_Address.....	90
8.3.2	Event_Round .....	91
8.3.3	Recommended event search algorithm.....	93
8.3.4	Supervisory Data frames for Event_Arbitration .....	94
8.4	Devices_Scan.....	96
8.4.1	Device_Status .....	96
8.4.2	Device_Status protocol .....	98
8.4.3	Devices_Scan protocol .....	99
9	Mastership transfer .....	99
9.1	Mastership transfer operation .....	99
9.1.1	Bus administrator configuration.....	99
9.2	Mastership transfer specifications .....	100
9.2.1	States.....	100
9.2.2	Time-outs for mastership transfer .....	103
9.3	Supervisory data frames for mastership transfer .....	103
9.3.1	Device_Status telegram .....	103
9.3.2	Mastership transfer telegram.....	104
10	Link Layer Interface .....	104
10.1	Link Layer layering.....	104

10.2	Link Process Data interface .....	105
10.3	Link Message Data interface.....	105
10.3.1	General .....	105
10.3.2	Priority.....	106
10.3.3	Packet size .....	106
10.3.4	Protocol_Type.....	106
10.3.5	Message Transport Protocol .....	106
10.4	Link Supervision Interface.....	106
10.4.1	General .....	106
10.4.2	Link Supervision Interface procedures.....	106
10.4.3	MVB_Status.....	107
10.4.4	MVB_Control .....	107
10.4.5	MVB_Devices .....	108
10.4.6	MVB_Administrator .....	108
10.4.7	MVB_Report .....	110
11	Real-Time Protocols .....	111
12	Gateway Function .....	111
13	Network Management .....	111
13.1	Contents of this clause .....	111
13.2	MVB Managed objects .....	111
13.2.1	MVB link objects .....	111
13.3	MVB Services and management messages.....	112
13.3.1	MVB link services .....	112
	Bibliography .....	126
	Figure 1 – Reference device and structure of the document.....	11
	Figure 2 – State transition example .....	37
	Figure 3 – MVB configuration .....	39
	Figure 4 – Transceiver interface .....	42
	Figure 5 – Example of ESD segment .....	43
	Figure 6 – Example of terminator.....	44
	Figure 7 – ESD backplane section (double-line).....	46
	Figure 8 – ESD connector arrangement.....	47
	Figure 9 – ESD terminator connector arrangement .....	48
	Figure 10 – Example of start of frame (ESD) .....	49
	Figure 11 – End of an ESD frame (both cases) .....	50
	Figure 12 – EMD medium .....	51
	Figure 13 – Shielding (single-line segment) .....	53
	Figure 14 – Single-line device attachment .....	54
	Figure 15 – Double-line device attachment to EMD .....	55
	Figure 16 – EMD connectors arrangement.....	56
	Figure 17 – EMD terminator strapping .....	57
	Figure 18 – Example of start of an EMD frame .....	58
	Figure 19 – Example of pulse waveform at EMD transmitter .....	60
	Figure 20 – Example of end of EMD frame .....	61
	Figure 21 – EMD receiver test signal .....	62

Figure 22 – Optical link .....	64
Figure 23 – Optical connector (dimensions in millimeters).....	65
Figure 24 – Example of start of OGF frame .....	66
Figure 25 – Edge jitter .....	67
Figure 26 – Example of active star coupler .....	67
Figure 27 – Example of a duplicated star coupler.....	68
Figure 28 – "0" and "1" data encoding.....	69
Figure 29 – Non_Data symbols encoding.....	69
Figure 30 – Master Start Delimiter .....	70
Figure 31 – Slave Start Delimiter .....	70
Figure 32 – Example of End Delimiter for EMD medium.....	70
Figure 33 – Example of a valid frame (OGF medium).....	71
Figure 34 – Signal skew .....	72
Figure 35 – Example of repeater for single-line attachment.....	74
Figure 36 – Example of repeater connecting a double-line to a single line segment.....	75
Figure 37 – Master Frame Format .....	75
Figure 38 – Slave Frames .....	76
Figure 39 – Telegram timing.....	77
Figure 40 – Example of Reply delay.....	78
Figure 41 – Frame spacing at the source side .....	79
Figure 42 – Frame spacing at the destination(s) .....	79
Figure 43 – Frame spacing at the master side .....	80
Figure 44 – Master Frame contents .....	82
Figure 45 – Word ordering in a Slave Frame.....	84
Figure 46 – Process Data telegram .....	84
Figure 47 – Message Data telegram .....	85
Figure 48 – Supervisory Data telegram .....	86
Figure 49 – Basic Periods .....	87
Figure 50 – Example of construction of the Macro_Cycle .....	90
Figure 51 – General_Event_Request frame format .....	94
Figure 52 – Group_Event_Request frame (M = 6, C = ABCDEF).....	94
Figure 53 – Single_Event_Request frame .....	95
Figure 54 – Event_Identifier_Response frame .....	95
Figure 55 – Device_Status_Request.....	96
Figure 56 – Device_Status_Response .....	96
Figure 57 – Device_Status of Class 1 device .....	97
Figure 58 – Device_Status of Class 2/3/4/5 device .....	97
Figure 59 – Device_Status of a device with Bus Administrator capability .....	97
Figure 60 – Device_Status of a device with Gateway capability .....	98
Figure 61 – Mastership Transfer states .....	102
Figure 62 – Device_Status_Request (sent by current master) .....	103
Figure 63 – Device_Status_Response (sent by proposed master) .....	103
Figure 64 – Mastership_Transfer_Request (sent by current master).....	104



Figure 65 – Mastership_Transfer_Response (sent by proposed next master) .....	104
Figure 66 – Link Layer Layering.....	105
Table 1 – Template for the specification of an interface procedure .....	34
Table 2 – Example of message structure .....	35
Table 3 – Example of textual message form (corresponding to Table 2).....	36
Table 4 – State transitions table .....	37
Table 5 – MVB devices capabilities .....	40
Table 6 – Pin assignment for the ESD connector .....	47
Table 7 – Pin assignment for the EMD connector.....	56
Table 8 – Master Frame types and F_code .....	83
Table 9 – LS_RESULT encoding .....	107
Table 10 – MVB_Status object .....	107
Table 11 – MVB_Control object .....	108
Table 12 – MVB_Devices object .....	108
Table 13 – MVB_Administrator object.....	109
Table 14 – LS_V_REPORT encoding.....	111
Table 15 – Example of mvb_administrator_list.....	123

## INTERNATIONAL ELECTROTECHNICAL COMMISSION

**ELECTRONIC RAILWAY EQUIPMENT –  
TRAIN COMMUNICATION NETWORK (TCN) –****Part 3-1: Multifunction Vehicle Bus (MVB)**

## FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as “IEC Publication(s)”). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 61375-3-1 has been prepared by IEC technical committee 9: Electrical equipment and systems for railways.

This first edition cancels and replaces the clauses of IEC 61375-1, second edition, published in 2007, relevant to the specification of MVB and constitutes a technical revision.

It was prepared taking into account IEC 61375-1 third edition.

The text of this standard is based on the following documents:

FDIS	Report on voting
9/1644/FDIS	9/1668/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts of IEC 61375 series, under the general title *Electronic railway equipment – Train communication network (TCN)*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

## INTRODUCTION

This part of IEC 61375 specifies one component of the Train Communication Network, the Multifunction Vehicle Bus (MVB), a serial data communication bus designed primarily, but not exclusively, for interconnecting equipment where interoperability and interchangeability are needed.

This part specifies:

- a) the physical media in single-line and double-line configurations;
- b) the signalling and the redundancy handling;
- c) the format and timing of the transmitted frame and telegrams;
- d) the organisation of the bus traffic;
- e) the allocation of Mastership;
- f) the management of the bus;
- g) the Link Layer interface and the layer management interface.

This part is structured following the OSI layers of a reference MVB device as shown in Figure 1:

### Clause 4 Physical Layer

- Electrical medium for short distance (RS-485, 20,0 m)
- Electrical medium for middle distance (transformer-coupled, 200,0 m)
- Optical fibre for long distances (glass fibres, 2,0 km)

### Clause 5 Medium-dependent signalling

- Frame encoding and decoding
- Line Unit interface
- Physical redundancy handling

### Clause 6 Frames and telegrams

- Master Frame and Slave Frame encoding, Telegram timing

### Clause 7 Link Layer Control

- Addressing
- Master Frame and Slave Frame format

### Clause 8 Medium allocation

- Periodic Polling
- Event Polling
- Devices Scan

### Clause 9 Mastership transfer

- Regular and exceptional mastership transfer

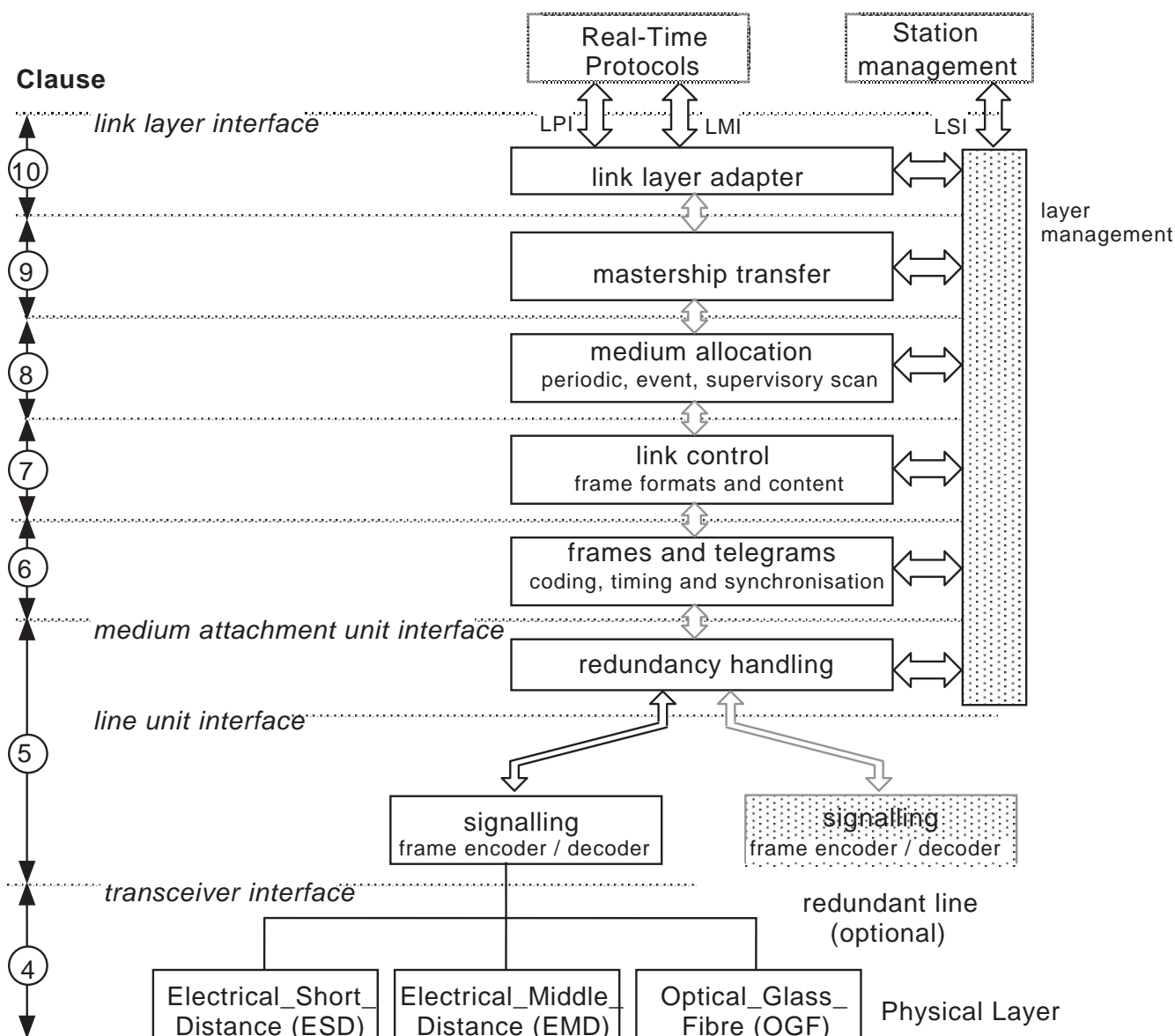
### Clause 10 Link Layer Interface

- Link Process Data Interface (LPI),
- Link Message Data Interface (LMI),
- Link Supervision Interface (LSI).

### Clause 11 Real-Time Protocols

### Clause 12 Gateway Function

### Clause 13 Network Management



**Figure 1 – Reference device and structure of the document**

# **ELECTRONIC RAILWAY EQUIPMENT – TRAIN COMMUNICATION NETWORK (TCN) –**

## **Part 3-1: Multifunction Vehicle Bus (MVB)**

### **1 Scope**

This part of IEC 61375 applies where MVB is required.

### **2 Normative references**

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60245-1, *Rubber insulated cables – Rated voltages up to and including 450/750 V – Part 1: General requirements*

IEC 60304, *Standard colours for insulation for low-frequency cables and wires*

IEC 60332-1-1, *Tests on electric and optical fibre cables under fire conditions – Part 1-1: Test for vertical flame propagation for a single insulated wire or cable – Apparatus*

IEC 60571, *Electronic equipment used on rail vehicles*

IEC 60794-1-1, *Optical fibre cables – Part 1-1: Generic specification – General*

IEC 60807 (all parts), *Rectangular connectors for frequencies below 3 MHz*

IEC 60870-5-1, *Telecontrol equipment and systems – Part 5: Transmission protocols – Section One: Transmission frame formats*

IEC 61375-2-1, *Electronic railway equipment – Train Communication Network (TCN) – Part 2-1: Wire Train Bus (WTB)*

ISO/IEC 8482, *Information technology – Telecommunications and information exchange between systems – Twisted pair multipoint interconnections*

ISO/IEC 8802-2, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 2: Logical link control*

ISO/IEC 8824 (all parts), *Information technology – Abstract Syntax Notation One (ASN.1)*

ISO/IEC 8825 (all parts), *Information technology – ASN.1 encoding rules*

ISO/IEC 9646 (all parts), *Information technology – Open Systems Interconnection – Conformance testing methodology and framework*

ISO/IEC 13239, *Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures*

ITU-T Recommendation Z.100, *Specification and Description Language (SDL)*

### 3 Terms and definitions, abbreviations and conventions

#### 3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

NOTE Keywords in this standard are written with the first letter of each word in upper case and, when they are composed of two or several words, these are joined by an underscore. This convention allows keywords to be tracked in the documents.

##### 3.1.1

##### **address**

identifier of a communication partner, of which several types exist, depending on the layer

##### 3.1.2

##### **agent**

application process in a Station which accesses the local managed objects on behalf of the Manager

##### 3.1.3

##### **Aperiodic Data**

transmission of Process Data on a demand basis. This service is not used

##### 3.1.4

##### **Application Layer**

upper layer in the OSI model, interfacing directly to the Application

##### 3.1.5

##### **Application Layer Interface**

definition of the services offered by the Application Layer

##### 3.1.6

##### **Application Messages Adapter**

code directly called by the application implementing the Messages services

##### 3.1.7

##### **Application Messages Interface**

definition of the Messages services

##### 3.1.8

##### **Application Process**

communicating entity, implemented for instance by a task

##### 3.1.9

##### **Application Processor**

processor which runs a communicating Application Process

##### 3.1.10

##### **Application Supervision Interface**

definition of the Supervision services available in particular to the Agent

### 3.1.11

#### **Application Variables Adapter**

code directly called by the application implementing the Variables services

### 3.1.12

#### **Application Variables Interface**

definition of the Variables services

### 3.1.13

#### **arbiter**

device, or common protocol followed by several devices, which selects one of several devices competing for mastership

### 3.1.14

#### **Auxiliary Channel**

channel used for detecting additional Nodes

### 3.1.15

#### **Basic Period**

bus activity is divided into periods. The shortest is the Basic Period, which consists of a Periodic Phase (for Periodic Data) and of a Sporadic Phase (for Message Data and Supervisory Data)

### 3.1.16

#### **big-endian**

ordering scheme for storing or transmitting data in which the most significant part of a multiple-octet data is stored at the lowest octet address, and transmitted first

### 3.1.17

#### **bit-stuffing**

method specified by ISO/IEC 13239 to prevent Frame Data from being misinterpreted as a Flag, consisting of inserting an additional "0" symbol after each string of five "1" symbols and removing this "0" at reception

### 3.1.18

#### **bridge**

device which stores and forwards frames from one bus to another on the base of their Link Layer addresses

### 3.1.19

#### **broadcast**

nearly simultaneous transmission of the same information to several destinations. Broadcast in the TCN is not considered reliable, i.e. some destinations may receive the information and others not

### 3.1.20

#### **bus**

communication medium which broadcasts the same information to all attached participants at nearly the same time, allowing all devices to obtain the same sight of its state, at least for the purpose of arbitration

### 3.1.21

#### **Bus Administrator**

device capable of becoming Master of the MVB

### 3.1.22

#### **Bus Controller**

processor or integrated circuit in charge of the Link Layer of communication



**3.1.23****Bus Switch**

switch or relay within a WTB Node which connects electrically the cable sections of the two directions

**3.1.24****Caller**

Application Process which initialises a message exchange

**3.1.25****Check Sequence**

method of error detection based on appending to the transmitted useful data a checksum or a cyclic redundancy check (CRC) calculated on the useful data

**3.1.26****Check Variable**

Process Variable of type antivalent boolean protecting another Process Variable

**3.1.27****Check Offset**

bit offset of a Check Variable within a Dataset

**3.1.28****Closed Train**

train consisting of a set of vehicles, where the composition does not change during normal operation, for instance metro, suburban train, or high-speed train units

**3.1.29****composition**

number and characteristics of the vehicles forming a train

**3.1.30****configuration**

definition of the topology of a bus, the devices connected to it, their capabilities and the traffic they produce; by extension, the operation of loading the devices with the configuration information before going to regular operation

**3.1.31****Connect Confirm**

response of the Consumer to the Connect Request of the Producer

**3.1.32****Connect Request**

first packet of a message sent from Producer to Consumer

**3.1.33****Consist**

Single vehicle or a group of vehicles which are not separated during normal operation. A Consist could contain no, one or several Consist networks.

**3.1.34****Consist network**

bus connecting equipment within a consist, e.g. the MVB, and which conforms or adapts to the TCN Real-Time protocols as described in this document

### **3.1.35**

#### **consistency**

Dataset consisting of several elements is consistent if all elements are read or written in one indivisible operation

### **3.1.36**

#### **Consumer**

receiver of a message at the Transport Layer (see: Producer)

### **3.1.37**

#### **continuity vehicle**

vehicle without an operational Train Bus Node, but carrying a section of the bus to connect passively the Train Bus of its adjacent vehicles.

### **3.1.38**

#### **conversation**

data exchange at the Application Layer, consisting of a Call Message and a Reply Message (the latter is missing in the multicast protocol). A conversation starts with the first Connect Request frame and ceases when the last acknowledgement for the Reply Message has been received or is no longer expected

### **3.1.39**

#### **datagram**

frame containing all information necessary to forward it to its final destination, without knowledge of previous frame's contents. Datagrams do not use a previous connection establishment and they are not acknowledged at the Link Layer

### **3.1.40**

#### **Dataset**

all Process Variables transmitted in one Process Data frame

### **3.1.41**

#### **delimiter**

sequence of signals which includes code violation symbols (neither "1" nor "0") which is used to delimit the start (Start Delimiter) and the end (End Delimiter) of a frame, as defined for instance in IEC 61158-2

### **3.1.42**

#### **Destination Device**

receiver of a frame at the Link Layer (see: Source Device)

### **3.1.43**

#### **device**

unit connected to one or more busses

### **3.1.44**

#### **Device Address**

Device Address identifies a device within a bus; On the MVB, the Device Address has 12 bits; On the WTB, the Device Address has 8 bits, the least significant 6 bits being the Node Address.

NOTE A device connected to several busses may have a different Device Address for each bus. Special devices such as repeaters only participate at the Physical Layer and have no Device Address.

### **3.1.45**

#### **Device Status**

16-bit word expressing the capabilities and status of an MVB device connection

**3.1.46****Direction 1**

one direction of a WTB Node

**3.1.47****Direction 2**

other direction of a WTB Node

**3.1.48****Electrical Middle Distance**

one of the media of the MVB

**3.1.49****Electrical Short Distance**

one of the media of the MVB

**3.1.50****End Delimiter**

sequence which ends a frame before the medium returns to idle

**3.1.51****End Node**

Node which terminates the two bus segments connected to it but does not establish continuity between them

**3.1.52****Event Round**

sequence of polls in which all events pending at the start are read

**3.1.53****extension box**

wiring box where the trunk cable is interrupted and passively extended by an extension cable to connect a device

**3.1.54****extension cable**

cable inserting a Node in a trunk cable, consisting of two separate twisted wire pairs per line, possibly of smaller cross-section than the trunk cable itself

**3.1.55****field device**

device attaching simple sensors and actuators to the bus, outside a rack

**3.1.56****final**

receiver of a packet (data or acknowledgement) at the Network Layer. When two devices communicate within the same bus, the final is located in the destination device (see: origin)

**3.1.57****Flag**

sequence of "1" and "0" symbols which serves to delimit the beginning or the end of a frame. Flags which would appear in the transmitted data are modified by bit-stuffing, as defined for instance in ISO/IEC 13239

**3.1.58**

**frame**

sequence of consecutive symbols sent in one time slot by a transmitter, between two slots where the line is idle.

**3.1.59**

**Frame Check Sequence**

16-bit FCS specified in ISO/IEC 13239

**3.1.60**

**Frame Data**

data transmitted between the Start Delimiter and the End Delimiter (on the MVB) or between the Preamble and the End Delimiter (on the WTB)

**3.1.61**

**fritting**

electrical cleaning of oxidised contacts by applying a breakdown voltage over the contact

**3.1.62**

**Function**

Application Process which exchanges messages with another Application Process

**3.1.63**

**Function Directory**

directory which maps a Function Identifier to a Station Identifier and vice-versa

**3.1.64**

**Function Identifier**

8-bit identifier of a Function

**3.1.65**

**F code**

in a Master Frame, indicates the request and the expected response Slave Frame size

**3.1.66**

**gateway**

connection between different busses at the Application Layer requiring application-dependent data analysis and protocol conversion

**3.1.67**

**Group Address**

address of a multicast group to which a Node belongs

**3.1.68**

**Group Directory**

directory which indicates to a Node to which multicast group it pertains

**3.1.69**

**hamming distance**

minimum number of bits of a given correct bit sequence, which, if inverted, create a false bit sequence indistinguishable from a correct one

**3.1.70**

**HDLC**

High-level Data Link Control, a set of standardised protocols, including ISO/IEC 13239 for data transmission

**3.1.71****HDLC Data**

data transmitted in an HDLC frame

**3.1.72****Inauguration**

operation executed in case of composition change, which gives all Nodes of the WTB their address relative to the Master, their orientation and the descriptor of all named Nodes on the same bus

**3.1.73****Individual Period**

interval between two successive transmissions of the same Process Data from the same source. The Individual Period is a power-of-2 multiple of the Basic Period

**3.1.74****instance**

- a) one of several objects which share the same definition (object instance)
- b) one of several (simultaneous or not) executions of the same program (process instance)

**3.1.75****integrity**

property of a system to recognise and to reject wrong data in case of malfunction of its parts

**3.1.76****Intermediate Node**

Node which establishes continuity between two bus sections connected to it, but does not terminate them

**3.1.77****jumper cable**

cable connecting the trunk cables of two consecutive vehicles, possibly of a larger cross-section than the trunk cable, and which is plugged by hand in the case of the UIC-cable. There are generally two jumper cables between vehicles

**3.1.78****Line**

non-redundant bus. A dual-thread bus consists of two lines

**3.1.79****Line Unit**

all circuits providing the electrical attachment to a line

**3.1.80****Link Address**

address supplied to the Link Layer to identify to which Bus and to which Device Address a packet is sent or received

**3.1.81****Link Control**

field in the HDLC frame which indicates the type of frame

**3.1.82****Link Data**

data transported by the Link Layer, but not relevant to it

**3.1.83**

**Link Header**

part of a Message Data frame relevant to the Link Layer

**3.1.84**

**Link Layer**

layer in the OSI model establishing point-to-point and broadcast connections between devices attached to the same bus

**3.1.85**

**Link Layer Interface**

interface between Link Layer and higher communication layers

**3.1.86**

**Link Layer Management**

interface controlling the Link Layer for management purposes

**3.1.87**

**little-endian**

ordering scheme for storing or transmitting data in which the least significant part of a multiple-octet data is stored at the lowest octet address, and transmitted first

**3.1.88**

**local area network**

part of a network characterised by a common medium access and address space

**3.1.89**

**logical link control**

protocols and associated frame formats which serve to control the Link Layer

**3.1.90**

**Logical Address**

address which is not bound to a specific device (e.g. the Process Data address)

**3.1.91**

**Logical Port**

ports of a device used for the Process Data traffic and addressed by the Logical Address

**3.1.92**

**Macro Cycle**

number of Basic Periods corresponding to a Macro Period

**3.1.93**

**Macro Period**

longest Individual Period, after which the periodic traffic returns to the same pattern, counted in milliseconds

**3.1.94**

**Main Channel**

channel over which the main bus traffic is received

**3.1.95**

**Management Message**

message exchanged between a Manager and an Agent for Network Management.

**3.1.96****Manager**

Function in a Station which is dedicated to Network Management and which sends management Call Messages through System Addresses

**3.1.97****marshalling**

allocation of application addresses or names to the Process Variables of a dataset, that, on the WTB, depends on the Node Type and Version

**3.1.98****Master**

device which spontaneously sends information on a bus to a number of slave devices. It may give a Slave the right to transmit for one Slave Frame only within a limited time

**3.1.99****Master Frame**

frame sent by a Master

**3.1.100****Master Start Delimiter**

Start Delimiter of a Master Frame

**3.1.101****medium access control**

sublayer of the Link Layer, which controls the access to the medium (arbitration, mastership transfer, polling)

**3.1.102****medium dependent interface**

mechanical and electrical interface between the transmission medium and a Medium Attachment Unit

**3.1.103****medium**

physical carrier of the signal: electrical wires, optical fibres, etc.

**3.1.104****Medium Attachment Unit**

device used as a coupler to the transmission medium

**3.1.105****message**

data item transmitted in one or several packets

**3.1.106****Messages**

transmission service of the TCN

**3.1.107****Message Data**

data transmitted sporadically by the Link Layer in relation to message transmission; the corresponding Link Layer service

**3.1.108**

**messenger**

communication stack caring for end-to-end message communication and interfacing to the application

**3.1.109**

**multicast**

transmission of the same message to a group of Repliers, identified by their Group Address. The word "multicast" is used even if the group includes all Repliers

**3.1.110**

**Multifunction Vehicle Bus**

**MVB**

Vehicle Bus to be used for connecting programmable stations and simple sensors/actors.

**3.1.111**

**Multiple Unit Train**

train consisting of a set of closed trains, where the composition of the set may change during normal operation

**3.1.112**

**network**

set of possibly different communication systems which interchange information in a commonly agreed way

**3.1.113**

**Network Address**

address which identifies a Function or a Station within the network. It can be either a User Address or a System Address

**3.1.114**

**Network Header**

part of a Message Data frame relevant to the Network Layer

**3.1.115**

**Network Layer**

layer in the OSI model responsible for routing between different busses

**3.1.116**

**Network Management**

operations necessary to remotely configure, monitor, diagnose and maintain the network

**3.1.117**

**Node**

device on the Wire Train Bus, which may act as a gateway between Train Bus and Vehicle Bus

**3.1.118**

**Node Address**

address of a Node on the Train Bus (6 bits). It is equal to the least significant 6 bits of the 8-bit Device Address on the WTB

**3.1.119**

**Node Descriptor**

24-bit data structure which indicates for a Node its Node Period and its Node Key



**3.1.120****Node Directory**

directory which maps the Node Address to the Device Address (one-to-one mapping in WTB)

**3.1.121****Node Key**

16-bit identifier selected by the application to identify a Node's type and version. The Master distributes it to all other Nodes after each composition change and before exchanging data

**3.1.122****Node Period**

on the WTB, desired Individual Period of a Node (identical to Individual Period except if overload occurs)

**3.1.123****octet**

8-bit word stored in memory or transmitted as a unit \*

**3.1.124****Open Train**

train consisting of a set of vehicles where the configuration may change during normal operation, for instance international UIC trains

**3.1.125****Optical Glass Fibre**

one of the media of the MVB

**3.1.126****origin**

sender of a packet (data or acknowledgement) at the Network Layer. When two devices communicate within the same bus, the Origin is located on the source device (see: final)

**3.1.127****packet**

unit of a message (information, acknowledgement or control) transmitted in exactly one Message Data frame

**3.1.128****period**

time unit after which a periodic pattern repeats itself

**3.1.129****Periodic Data**

Process Data transmitted periodically, at an interval which is the Individual Period

**3.1.130****Periodic List**

list of Nodes, addresses or devices to be polled in each period of a Macro Cycle

**3.1.131****Periodic Phase**

phase during which the Master polls for Periodic Data according to its Periodic List

---

\* IEC prescribes 'octet' instead of 'byte'.

**3.1.132**

**Physical Address**

Device Address on the MVB and the Node Address on the WTB which identify communicating devices on the same bus

**3.1.133**

**Physical Port**

Port used for the Message Data or the Supervisory Data traffic and addressed by the Device Address

**3.1.134**

**pitch**

distance between adjacent devices on the same electrical bus required to avoid clustering of bus loads

**3.1.135**

**polling**

sending of a Master Frame in order to receive a Slave Frame

**3.1.136**

**Port**

memory structure which contains data for transmission or reception, and in which a new value overwrites the former value (buffer, not queue). A Port provides means for simultaneous access by the bus and the application(s)

**3.1.137**

**Port Index Table**

look-up table which deduces the memory address of a port from the Logical Address of the Process Data

**3.1.138**

**Preamble**

sequence of signals heading a frame for the purpose of synchronising the receiver, used on the WTB

**3.1.139**

**Presentation Layer**

layer in the OSI model responsible for data representation and conversion

**3.1.140**

**Process Data**

source-addressed data broadcast periodically by the link layer in relation with Process Variables transmission; the corresponding Link Layer service

**3.1.141**

**Process Variable**

variable expressing the state of a process (e.g. speed, brake command)

**3.1.142**

**Producer**

sender of a message at the Transport Layer (see: Consumer)

**3.1.143**

**Publisher**

source of a Dataset for broadcasting (see: Subscriber)

**3.1.144****PV Name**

identifier of a Process Variable

**3.1.145****PV Set**

set of Process Variables belonging to the same Dataset

**3.1.146****queue**

memory storing an ordered set of frames in a first-in, first-out fashion

**3.1.147****rack**

equipment containing one or more devices, attached to the same segment

**3.1.148****reassembly**

act of regenerating a long message from several packets generated by segmentation

**3.1.149****receiver**

electronic device which may receive signals from the physical medium

**3.1.150****Receive Queue**

queue for receiving Message Data in a device

**3.1.151****regular operation**

normal bus activity as opposed to Inauguration (WTB) or configuration (MVB)

**3.1.152****repeater**

connection at the Physical Layer between bus segments, providing an extension of the bus beyond the limits permitted by passive means. The connected segments operate at the same speed and with the same protocol. The delay introduced by a repeater is in the order of one bit duration

**3.1.153****Replier**

Application Process which has been requested by the Caller to receive a Call Message and to reply with a Reply Message

**3.1.154****residual error rate**

probability of integrity breach (unrecognised wrong bit) per transmitted bit

**3.1.155****router**

connection between two busses at the Network Layer, which forwards datagrams from one bus to another on the base of their Network Address

**3.1.156****scan**

polling of devices in a certain sequence for supervisory purposes

**3.1.157**  
**section**

part of a segment, which is passively connected to another section without terminator in between

**3.1.158**  
**segment**

piece of cable to which devices are attached, terminated at both ends by its characteristic impedance. Segments may consist of several sections (non-terminated) connected by connectors

**3.1.159**  
**segmentation**

division of a long message into several shorter frames for transmission

**3.1.160**  
**Send Queue**

queue for sending Message Data in a device

**3.1.161**  
**service**

capabilities and features of a sub-system (e.g. a communication layer) provided to a user

**3.1.162**  
**Session Header**

part of a Message Data frame relevant to the Session Layer

**3.1.163**  
**Session Layer**

OSI layer in charge of establishing and closing communication

**3.1.164**  
**Side A**

one side of a vehicle with respect to a WTB Node

**3.1.165**  
**Side B**

other side of a vehicle with respect to a WTB Node

**3.1.166**  
**Slave**

device which receives information from the bus or sends information on the bus in response to a request (also called a poll) from the Master

**3.1.167**  
**Slave Frame**

frame sent by a Slave

**3.1.168**  
**Slave Start Delimiter**

Start Delimiter of a Slave Frame on the MVB

**3.1.169**  
**Source Device**

sender of a frame at the Link Layer (see: destination device)

**3.1.170****sporadic transmission**

transmission which is made upon demand, when an event external to the network requires it (also called aperiodic, event-driven, demand-driven transmission)

**3.1.171****Sporadic Data**

data frames transmitted on demand to carry Message Data or Supervisory Data

**3.1.172****Sporadic Phase**

second half of a Basic Period, dedicated to the demand-driven transmission of messages and bus management data

**3.1.173****star coupler**

device which takes the light of an optical fibre and redistributes it to several other fibres

**3.1.174****Start Delimiter**

delimiter which announces the beginning of a frame on the MVB

**3.1.175****Station**

device capable of message communication, by contrast to simple devices, and which supports an Agent Function

**3.1.176****Station Directory**

directory which maps a Station Identifier to a Link Address and vice-versa

**3.1.177****Station Identifier**

8-bit identifier of a Station

**3.1.178****Station Status Word**

16-bit descriptor of the status and capabilities of a Station

**3.1.179****Strong Master**

Strong Node is currently Master and will not relinquish mastership until demoted to Weak Node status

**3.1.180****Strong Node**

Node selected by the application to become Strong Master. There may be only one Strong Master on a bus segment

**3.1.181****stub**

T-connection branching from an electrical bus line (at the tap), connecting a device to the line

**3.1.182****Subscriber**

one of the sinks of a broadcast Dataset (see: Publisher)

**3.1.183**

**Supervisory Data**

data transmitted within one bus only for the purpose of Link Layer supervision (e.g. arbitration on the MVB or Inauguration on the WTB)

**3.1.184**

**System Address**

Network Address of a Management Message exchanged between Manager and Agent, consisting of Node Address and Station Identifier

**3.1.185**

**tap**

place where a segment is tapped. A tap is a three-way electrical fork

**3.1.186**

**Telegram**

Master Frame and the corresponding Slave Frame, treated as a whole

**3.1.187**

**terminator**

circuit which closes an electrical transmission line, ideally by its characteristic impedance

**3.1.188**

**Terminator Switch**

switch which inserts the Terminator at the end of a segment on the WTB

**3.1.189**

**Topography**

data structure describing the Nodes attached to the Train Bus, including their address, orientation, position and Node Descriptor

**3.1.190**

**topology**

possible cable interconnection and number of devices a given network supports

**3.1.191**

**Topography Counter**

counter in a Node which is incremented at each new Inauguration

**3.1.192**

**Traffic Store**

shared memory accessed both by the network and the user, which contains the Process Data Port

**3.1.193**

**Train Communication Network**

data communication network for connecting programmable electronic equipment on-board rail vehicles

**3.1.194**

**Train Bus**

bus connecting the vehicles of a train, in particular, the WTB, and which conforms to the TCN protocols

**3.1.195**

**Train Network Management**

services of the Network Management for TCN

**3.1.196****transceiver**

combination of a transmitter and of a receiver

**3.1.197****transmitter**

electronic device which can transmit a signal on the physical medium

**3.1.198****Transport Data**

data carried by the Transport Layer, but not relevant to it

**3.1.199****Transport Header**

part of a Message Data frame relevant to the Transport Layer

**3.1.200****Transport Layer**

layer of the OSI model responsible for end-to-end flow control and error recovery

**3.1.201****trunk cable**

cable which runs along the vehicles, as opposed to extension cable or jumper cable

**3.1.202****User Address**

Network Address of a User Message exchanged between Functions, consisting of Node Address (or Group Address) and Function Identifier

**3.1.203****User Message**

messages exchanged between user Functions

**3.1.204****Variables**

transmission service of the TCN

**3.1.205****Var\_Offset**

bit offset of a Process Variable within a Dataset

**3.1.206****Vehicle Descriptor**

application-dependent information about a particular vehicle, such as length and weight

**3.1.207****Weak Master**

Weak Node is currently Master and which will relinquish mastership if it finds another, stronger Master

**3.1.208****Weak Node**

Node which may take over the bus mastership spontaneously, but which releases it if it detects a stronger Node

### 3.1.209

#### Wire Train Bus (WTB)

Train Bus for frequently coupled and uncoupled vehicles, such as international UIC trains

### 3.2 Abbreviations

ALI	Application Layer Interface, the definition of the semantics of all network services used by the application (a set of primitives, expressed as procedures, constants and data types)
AMA	Application Messages Adapter, the code directly called by the application which implements the Messages service
AMI	Application Messages Interface, the definition of the message services
ANSI	American National Standard Institute, a standardisation body in the United States
ASI	Application Supervision Interface, the definition of the Management services
ASN.1	Abstract Syntax Notation Number 1 on data presentation (ISO/IEC 8824)
AVA	Application Variables Adapter, the code directly called by the application implementing the Process Variable services
AVI	Application Variables Interface, the definition of the Process Variable services
BER	Basic Encoding Rules, a transfer syntax for ASN.1 data types (ISO/IEC 8825)
BR	Bit Rate, the rate of data throughput on the medium expressed in bits per second (bit/s) or in hertz (Hz), whichever is appropriate
BT	Bit Time, the duration of the transmission of one bit, expressed in $\mu$ s
ITU	International Telecommunication Union, the international standardisation body for telecommunications based in Geneva
CRC	Cyclic Redundancy Check, a data integrity check based on polynomial division
DIN	Deutsches Institut für Normung, the German national standardisation body
EIA	Electronics Industries Association, a standardisation body in the United States
EMD	Electrical Middle Distance, one of the media of the MVB
ESD	Electrical Short Distance, one of the media of the MVB
EP	Electro-Pneumatic brake cable as described in UIC leaflet 648
ERRI	European Railways Research Institute, laboratory based in Utrecht, Netherlands
FCS	Frame Check Sequence, an error detection code appended to the transmitted data, as specified in ISO/IEC 13239
GCT	Guidelines for Conformance Test, the conformance testing specified in Annex B of this Standard
HDLC	High-level Data Link Control, a Link Layer protocol whose frame format is defined in ISO/IEC 13239
IEC	International Electrotechnical Commission, Geneva
IEEE	Institute of Electrical and Electronics Engineers, New York
ISO	International Standard Organisation, Geneva



LLC	Logical Link Control, a sub-layer within the Link Layer ruling the data exchange
LME	Layer Management Entity, the entity in charge of supervising a layer on behalf of Network Management
LMI	Layer Management Interface, the services provided by the LME
MAC	Medium Access Control, a sub-layer within the Link Layer ruling which device is entitled to send on the bus
MAU	Medium Attachment Unit, the part of a Node which interfaces electrically to the bus and which provides/accepts binary logic signals
MIB	Management Information Base, the set of all objects accessed by Network Management
MVB	Multifunction Vehicle Bus, a Consist Network.
NRZ	Non-Return to Zero, the simplest encoding scheme in which one bit is represented by one level for a "1" and the other level for a "0", or vice-versa, with a separate clocking
OGF	Optical Glass Fibre, one of the media of the MVB
ORE	Office de Recherches et d'Essais, a UIC laboratory based in Utrecht, Netherlands
OSI	Open System Interconnection, a universal communication model defined in the ISO/IEC 7498
PCTR	Protocol Conformance Test Report, defined in ISO/IEC 9646
PICS	Protocol Implementation Conformance Statement, defined in ISO/IEC 9646
PTA	Process Data to Traffic Store Adapter, the component which accesses one of the Traffic Stores
RIC	Regulation for the reciprocal use of coaches and vans in international traffic, issued by UIC.
RTP	Real-Time Protocols, the common communication protocols given in Clause 11 of this standard
SDL	Specification and Description Language, a specification language defined by ITU-T Z100 Annex D for communication protocols
TCN	Train Communication Network, a set of communicating consist networks and Train Backbones.
TNM	Train Network Management
UIC	International Union of Railways , the international railways operators association
WTB	Wire Train Bus

### 3.3 Conventions

#### 3.3.1 Base of numeric values

This standard uses a decimal representation for all numeric values unless otherwise noted.

Analog and fractional values include a comma.

EXAMPLE 1 The voltage is 20,0 V.

Binary and hexadecimal values are represented using the ASN.1 (ISO/IEC 8824) convention.

EXAMPLE 2 Decimal 20 coded on 8 bits = '0001 0100'B = '14'H

### 3.3.2 Naming conventions

Keywords in the TCN specifications are written with a capital letter at the beginning.

If the word is composed, the different parts of the name are united with a space.

When a data structure is associated with a keyword, its type consists of the same basic words separated by an underscore.

When the value corresponding to the keyword is transmitted in a message, the corresponding field has the same name as the type, but in lower case.

When the value is passed as parameter, the parameter name has the same name as the field in a message.

In the SDL diagrams, the corresponding variable has the same name as the type, but without underscores.

#### EXAMPLES

Topo Counter is a counter of the link layer;

It is of the type Topo\_Counter, which is an UNSIGNED6.

When its value is transmitted in a message, the corresponding field is called 'topo\_counter'.

When its value is transmitted across a procedural interface, the parameter is called 'topo\_counter', its C-type is Type\_Topo\_Counter.

In the SDL diagrams, the variable representing the counter is called TopoCounter.

### 3.3.3 Time naming conventions

Time values beginning with a lower case (e.g. t mm) are measurable time intervals.

Time values beginning with an uppercase (e.g. T reply) are parameters or time-out values.

### 3.3.4 Procedural interface conventions

A procedural interface is defined by a set of service primitives, which represent an abstract, implementation-independent interaction between the service user and the service provider.

These primitives are expressed in this standard as procedures in the ANSI C syntax with typed parameters.

This ought to be considered as a semantic description only, which does not imply a particular implementation or language. Any interface which provides the same semantics is allowed.

Conformance to the syntax of this interface cannot be claimed. Implementations are free to change the procedure or parameters names, to add parameters or to split procedures, as long as the specified service is provided.

Interface procedures are defined in the ANSI C syntax using the Courier font.

Procedure names, variables and parameters all appear in lower case.

## EXAMPLE 1 Im\_send\_request

Constants and type definitions all appear in upper case.

## EXAMPLE 2 UNSIGNED32

The name of a procedure or of a type is prefixed:

for the Variables service by

- lp\_ or LP\_ Link Layer
- ap\_ or AP\_ Application Layer

for the Messages service by

- MD\_ messages in general
- lm\_ or LM\_ Link Layer
- nm\_ or NM\_ Network Layer
- tm\_ or TM\_ Transport Layer
- sm\_ or SM\_ Session Layer
- am\_ or AM\_ Application Layer

Table 1 shows a template used for procedures and types.

**Table 1 – Template for the specification of an interface procedure**

Definition	<i>The service or data type is expressed here.</i> <i>In case of an indication procedure, the event which triggers the call is indicated here, beginning with "When"</i> <i>The name and parameters of the service procedure are defined here.</i> <i>In case of an indication procedure, the type of the procedure is specified.</i> <i>Input parameters, output parameters and return parameters are distinguished</i>	
Syntax	<pre>MD_RESULT          lm_send_request                     (                         /* example */                         unsigned,      destination,                         UNSIGNED8      link_control,                         MD_PACKET *    p_packet                         ENUM8 *        status                     )</pre>	
Input	<i>Input parameters are supplied to the procedure, which is not allowed to modify them</i>	
	destination	The data type "unsigned" is compiler-dependent
	link_control	parameter passed by reference, not modified by the procedure. The data type is an 8-bit word
	p_packet	The "*" denotes a pointer to the p_packet data structure, which is of type MD_PACKET, defined elsewhere in this standard
Output	<i>Output parameters are expected to be modified by the call</i>	
	status	The type ENUM8 is an 8-bit enumeration type
Result	<i>The Result parameter is an optional output parameter which expresses success or failure of the call, but not necessarily of the service</i>	
	MD_RESULT	<p>The Result parameter is of the type: AM_RESULT for the AMI, MD_RESULT for the LMI, LP_RESULT for the LPI, AP_RESULT for the AVI, The template specifies the error codes expected for each procedure individually. The Result parameter is not explicitly described if the only two values expected are: xx_OK = 0    successful completion; xx_FAILURE    &lt;&gt; 0 some problem. The result can also be returned as an output parameter in the parameter list, depending on the implementation</p>
Usage	The rules listed after the procedure template indicate how the procedure should be used. Although usage rules are not mandatory, not following them produces unpredictable results	
NOTE    Data structures represented in this table are interface specifications which should not be confused with formats of the same data structures when transmitted over a bus, see 3.3.5.		

3.3.5 Specification of transmitted data

The format of transmitted data, single frames as well as whole messages, is specified in two forms:

- a) a graphical form, which is not normative, but shows the message structure at a glance.
- b) a textual form based on ASN.1, with encoding rules specified in 6.3 of IEC 61375-2-1.

EXAMPLE 1 A graphical form of a message is shown in Table 2, the corresponding textual form is shown in Table 3.

Table 2 – Example of message structure

first transmitted octet								next transmitted octet								
bit->	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	snu	gni	node_id						station_id							
2	next_station_id								rsv1	Tvd	topo_counter					
4	tnm_key								sif_code (= 3)							
6	parameter 1															
8	parameter 2								parameter 3						par4	
	parameter 5: ARRAY [n] OF (repeat following field (n) times)															
	parameter 5.1															
	parameter 5.2															
	parameter5.3: STRING32															
	(CHARACTER8)								last character or 0							
^octet																

In the graphical form, one line is used for each word of 16 bits, but in Clause 4 (WTB), lines are 8-bit oriented.

Arrays of parameters are preceded by a repetition frame on the top and to its left.

Repetitions can be nested (see parameter 5.3 in Table 4).

If the size of a parameter may be longer than three words, three lines are allocated for it and the middle one has a shaded border.

**Table 3 – Example of textual message form (corresponding to Table 2)**

```

Call_Mgt_Message ::= RECORD
{
    snu                BOOLEAN1 (=1)      -- this '1' means that message
                                           uses system addressing
    gni                BOOLEAN1 (=0)      -- this '0' means that final
                                           is an individual device
    node_id            UNSIGNED6          -- 6-bit node address of final
    station_id         UNSIGNED8          -- 8-bit identifier of the
                                           station
    next_station_id    UNSIGNED8          -- 8-bit identifier of next
                                           station
    rsv1              BOOLEAN1 (=0)      -- this bit is always 0
    tvd               BOOLEAN1          -- this bit indicates if
                                           topography counter is valid
    topo_counter       UNSIGNED6          -- 6-bit topography counter
    tnm_key            UNSIGNED8,        -- announces a network
                                           management call message.
    sif_code           UNSIGNED8,        -- there is a different
                                           SIF_code for each
                                           Management Message
    parameter1         INTEGER16,        -- a 16-bit value. If the
                                           parameter has less than 16
                                           bits, the value is right-
                                           justified and sign-extended
                                           (e.g. one single octet is
                                           transmitted as the second
                                           octet).
    parameter2         INTEGER8,        -- this value is transmitted
                                           in the most significant
                                           part of a word.
    parameter3         UNSIGNED6        -- this value is transmitted
                                           in the least significant
                                           octet, but the lower two
                                           bits of that octet are
                                           reserved for parameter4.
    par4              ANTIVALENT2        -- parameter4 has two bits
    parameter5         ARRAY [n] OF     -- parameter5 represents a
                                           structured data to be
                                           {
                                           -- repeated n times,
                                           containing:
                                           parameter5.1         INTEGER16        -- first parameter of the
                                           repeated field
                                           parameter5.2         UNIPOLAR4.16      -- second parameter of the
                                           repeated field
                                           parameter5.3         STRING32        -- third parameter is a string
                                           (array of up to 32 8-bit
                                           characters);
                                           - a string is closed by a
                                           '0' character, or by two
                                           such '0' characters to
                                           align on a 16-bit word
                                           boundary;
                                           - a void string consists of
                                           32 '0' characters;
                                           - the actual size of a
                                           string is deduced from the
                                           number of significant
                                           characters before the zero.
                                           },
}

```

Fields names begin with a lower case letter, their type begins with an upper case. Sometimes, the same type is used as a transmission format, in which case only the first letter is upper case, and as a C-type, in which case the whole type is in upper case.

EXAMPLE 2 Am\_Result (transmission format) and AM\_RESULT (C-type of an interface procedure).

3.3.6 State diagram conventions

The transport protocol state machine is described as in ISO/IEC 8802-2 (Logical Link Layer) in the form of a table, which specifies the transitions between the possible states in which a state machine may be.

Transitions between states are governed by events, coming from the Network Layer (inbound packets), from the Session Layer (commands) or from time-outs.

An action depending on the event is executed before leaving the state. This action defines the next state.

Figure 2 shows an example of a state transition diagram.

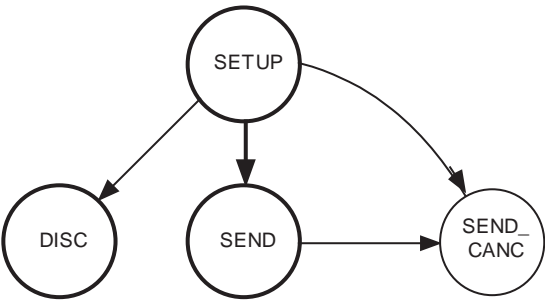


Figure 2 – State transition example

From “SETUP”, the machine may go to three different states, DISC, SEND or SEND\_CANC.

The transition between these states are governed as Table 4 shows.

Table 4 – State transitions table

Current state	Event	Action(s)	Next state (if <> current)
SETUP	rcv_DR	close_send (DR_reason);	DISC
	rcv_CC AND (conn_ref = CC_conn_ref)	IF (eot) THEN close_send (AM_OK); ELSE credit:= CC_credit; send_not_yet:= credit; send_data_or_cancel; END;	DISC  SEND or SEND_CANC
	TMO AND (rep_cnt = MAX_REP_CNT)	close_send (AM_CONN_TMO_ERR);	DISC

According to Table 4, there are three events which cause a transition from SETUP to the DISC state:

- a) rcv\_DR (received Disconnect\_Request, a network event). The corresponding action consists in closing the connection (close\_send) before going to state DISC;
- b) another network event (received Connect\_Confirm with correct reference), leading either to state DISC, SEND or SEND\_CANC, depending on the outcome of the send\_data or cancel procedure;
- c) A time-out conditioned by the predicate (rep\_cnt = MAX\_REP\_CNT), which also causes the closing of the connection.

## 4 Physical Layer

### 4.1 Topology

#### 4.1.1 Segments

An MVB shall consist of one or several bus segments of the following media:

- a) ESD: an Electrical Short Distance medium using a pair of wires with differential transmission according to the RS-485 standard, supporting up to 32 devices over a distance of 20,0 m without need for galvanic separation, and over longer distances when using galvanic isolation, as specified in 4.4.
- b) EMD: an Electrical Middle Distance medium, using a shielded, twisted wire pair, supporting up to 32 devices over a distance of 200,0 m and allowing transformers for galvanic isolation, as specified in 4.5.
- c) OGF: an Optical Glass Fibre medium, using optical fibres bussed through a star coupler, over a distance up to 2,0 km, intended for critical environments such as locomotives, as specified in 4.6.

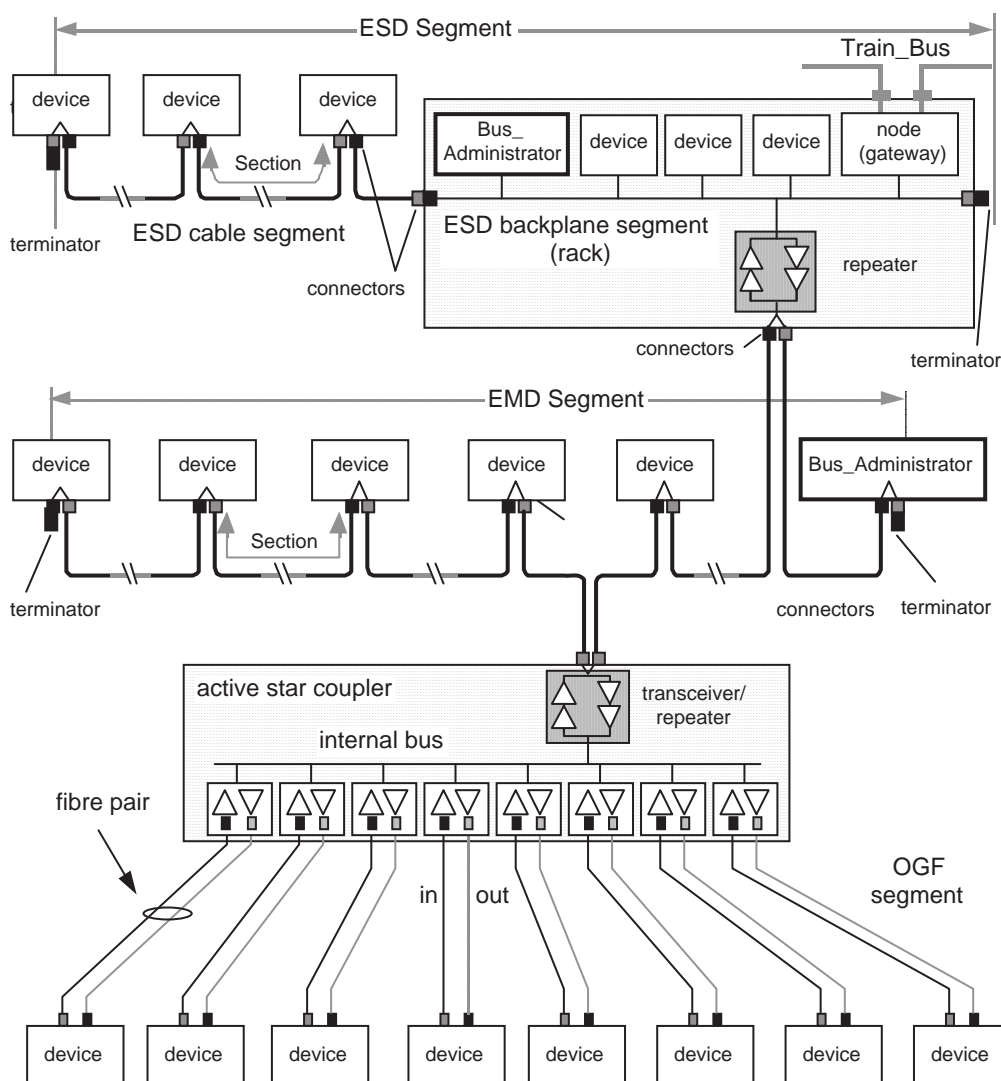
#### 4.1.2 Couplers

MVB segments shall be interconnected by couplers of one of the following types:

- a) repeater for connecting different media;
- b) star couplers for forming optical fibres into a bus.

**EXAMPLE** An MVB configuration, shown in Figure 3, consists of an ESD segment supporting a bus administrator, several devices within and outside of a rack and a gateway, an EMD segment supporting several devices and a bus administrator, and an OGF segment supporting a number of devices. Segments are interconnected by repeaters. The star coupler consists of an ESD segment which, in this example, supports no devices. Each segment in Figure 3 could be duplicated to increase availability.





**Figure 3 – MVB configuration**

#### 4.1.3 Double-line segments

As an option, a segment may be duplicated to increase availability (double-line segment).

An MVB may consist of a combination of double-line segments and single-line segments interconnected by bus couplers.

## 4.2 Device classes

### 4.2.1 Capabilities

There exist five classes of devices attached to the MVB, which differ in their capabilities.

MVB devices shall provide a subset of the following six capabilities listed in Table 5.

**Table 5 – MVB devices capabilities**

Capability	Description	Class
Device_Status	the device is able to send its Device_Status when polled	1, 2, 3, 4, 5
Process_Data	the device is able to send Process Data when polled and to receive Process Data from other devices	1, 2, 3, 4, 5
Message_Data	the device is able to send Message Data when polled and to receive Message Data from other devices. This capability implies that the devices are able to execute the Real-Time Protocols and have a Network Management Agent.	2, 3, 4, 5
User_Programmable	the device is able to be down-loaded with User Application programs. This capability implies the Message_Data capability.	3, 4, 5
Bus_Administrator	the device is able to act as a master. This capability implies the Message_Data, Process_Data and Device_Status capabilities and the ability to read the Device_Status of all other devices.	4, 5
TCN_Gateway	the device is able to access at least another bus (MVB or other). This capability implies the Device_Status, Process_Data and Message_Data capabilities, and the presence of a router if at least two of the busses obey to the Real-Time Protocols.	5

#### 4.2.2 Class 0 devices

Class 0 devices are not requested to support any capability.

NOTE Class 0 includes special devices, such as repeaters or star couplers, which do not participate in the application data exchange or participate in another way (for instance with another protocol).

#### 4.2.3 Class 1 devices

Class 1 devices shall offer the capabilities: Device\_Status and Process\_Data.

NOTE In these devices, the Port\_Address of the Process Data may be tied to their Device\_Address, for instance, the Port\_Address may be identical to the Device\_Address.

#### 4.2.4 Class 2 devices

Class 2 devices shall offer the capabilities: Device\_Status, Process\_Data and Message\_Data.

NOTE Class 2 devices are intelligent field devices which can be configured over the bus, but not programmed.

#### 4.2.5 Class 3 devices

Class 3 devices shall offer the capabilities: Device\_Status, Process\_Data, Message\_Data and User\_Programmable.

#### 4.2.6 Class 4 devices

Class 4 devices shall offer the capabilities: Device\_Status, Process\_Data, Message\_Data and Bus\_Administrator.

NOTE The User\_Programmable capability is optional.

#### 4.2.7 Class 5 devices

Class 5 devices shall offer the capabilities: Device\_Status, Process\_Data, Message\_Data and TCN\_Gateway.

Class 5 devices may offer the Bus\_Administrator capability.

NOTE Gateways with Bus\_Administrator capability can synchronise the busses.

#### 4.2.8 Device Attachment

Three different device attachments are considered:

- a) ESD: Devices equipped with the connectors carrying ESD signals according to 4.4;
- b) EMD: Devices equipped with the connectors carrying EMD signals according to 4.5;
- c) OGF: Devices equipped with the connectors carrying ESD signals according to 4.6.

NOTE A device may support more than one attachment.

### 4.3 Specifications common to all media

#### 4.3.1 Signalling speed

The signalling speed shall be 1,5 Mbit/s  $\pm$  0,01 %, using Manchester encoding (BR = 1,5 MHz or 1,5 Mbit/s, BT = 666,7 ns).

#### 4.3.2 Propagation delays

Between any two devices, the reply delay, including propagation delays and repeater delays and assuming a worst-case source delay of T\_source\_max (see 6.2.3), shall not be larger than T\_reply\_def = 42,7  $\mu$ s (see 6.2.2), except if the bus is operated in Extended Reply Mode.

If the worst case propagation delay is larger than T\_reply\_def, the bus may be operated in Extended Reply Delay mode, provided that the worst case round-trip delay T\_reply\_max is entered as a configuration parameter of all connected devices.

The difference between the maximum and the minimum propagation delay on two consecutive telegrams, considering overall propagation and repeater delays, shall at no place exceed 4,0 BT.

The sum of the interframe spacings from one master frame to another master frame shall be larger than 9,0 BT.

NOTE 1 The default value of T\_reply\_def of 42,7  $\mu$ s limits the number of repeaters to four, and the transmitted distance to 2,0 km. Using a value larger than T\_reply\_def is possible with certain precautions explained in the Extended Reply Delay mode (6.2.4.2).

NOTE 2 Fluctuations in propagation delay could reduce the interframe spacing and cause frame overlap. Repeaters may increase the interframe spacing, but overrun may occur if the above requirement is exceeded.

NOTE 3 Imposing a minimum sum of interframe spacings prevents a pair of fast master and slave from overrunning slower listeners.

#### 4.3.3 Transceiver interface

The transceiver interface specifies for all types of media the interface between the Bus\_Controller and the transceivers within a device.

This interface can remain internal to a device, but it is recommended to make it available for testing. This interface is not covered by conformance testing.

The transceiver interface is defined as an electrical interface operating with binary signals.

The transceiver interface assumes that the medium takes two distinct levels, a HIGH level and a LOW level. The HIGH and LOW levels are defined for each media.

If exposed, the transceiver interface shall consist of the following signals:

- a) TxS: Transmitter\_Signal

This signal controls the level of the medium. It is "0" when the medium is at the LOW level, or a "1" when the medium is at the HIGH level.

b) TxE: Transmitter\_Enable

This signal becomes "1" to enable the transmitter. This signal is not needed for optical transmission. Its timing is defined for each medium.

c) RxS: Receiver\_Signal

This signal expresses the state of the medium. It is "0" when the line is at the LOW level, "1" when the line is at the HIGH level. The receiver interprets undefined levels, either as LOW or as HIGH. There is no defined level when no transmitter is active, although some media define an idle level (generally LOW).

Figure 4 illustrates the transceiver interface for all three media.

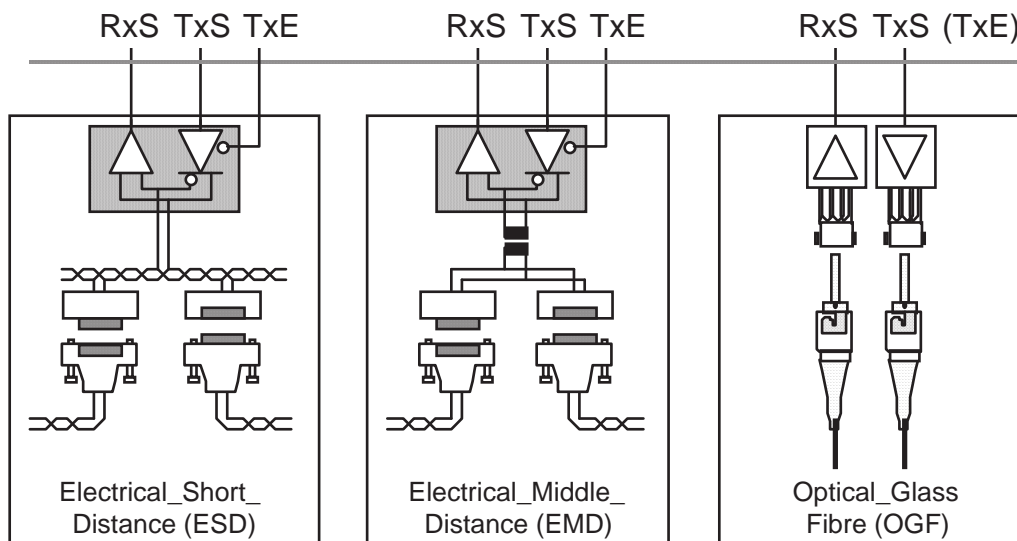


Figure 4 – Transceiver interface

#### 4.3.4 Redundant medium (option)

For applications which need a high availability, this subclause defines a redundancy scheme, in which the bus may be either fully duplicated (all segments are double-line) or partially duplicated (include single-line segments and double-line segments). In case this option is used, the following specifications apply.

##### 4.3.4.1 Identification

A double-line segment shall consist of two lines, operated in parallel.

All devices connected to a double-line segment shall identify the same line as Line\_A, and the other line as Line\_B.

##### 4.3.4.2 Layout

The layout of the busses and their loading shall ensure close propagation conditions on both lines of a double-line segment, to meet 5.2.2.2.

The layout shall take precautions to ensure fail-independence of the redundant lines.

4.3.4.3 Connection between single-line and double-line segments

When a double-line segment is connected to another double-line segment by a repeater, there shall be an independent repeater for each Line\_A and Line\_B, and the identification of Line\_A and Line\_B shall be maintained over this connection.

When a double-line segment is connected to a single-line segment by one repeater, the repeater shall be connected to both lines on one side and to one line on the other, as specified in 5.3.2.

4.4 Electrical Short Distance medium (choice)

The specifications of the Electrical Short Distance medium allow segments with up to 32 devices over a distance of up to 20 m without galvanic separation. Actually, the covered distance and the number of devices are limited by the distortion caused by cables, connectors and devices, by the ground quality and by the disturbance levels, and can be increased by providing galvanic isolation.

4.4.1 ESD topology

The ESD medium shall consist of two conductors (with an additional equipotential conductor), terminated and biased at each end, to which the devices are attached by short stubs.

EXAMPLE The principle connection of an ESD segment is shown in Figure 5.

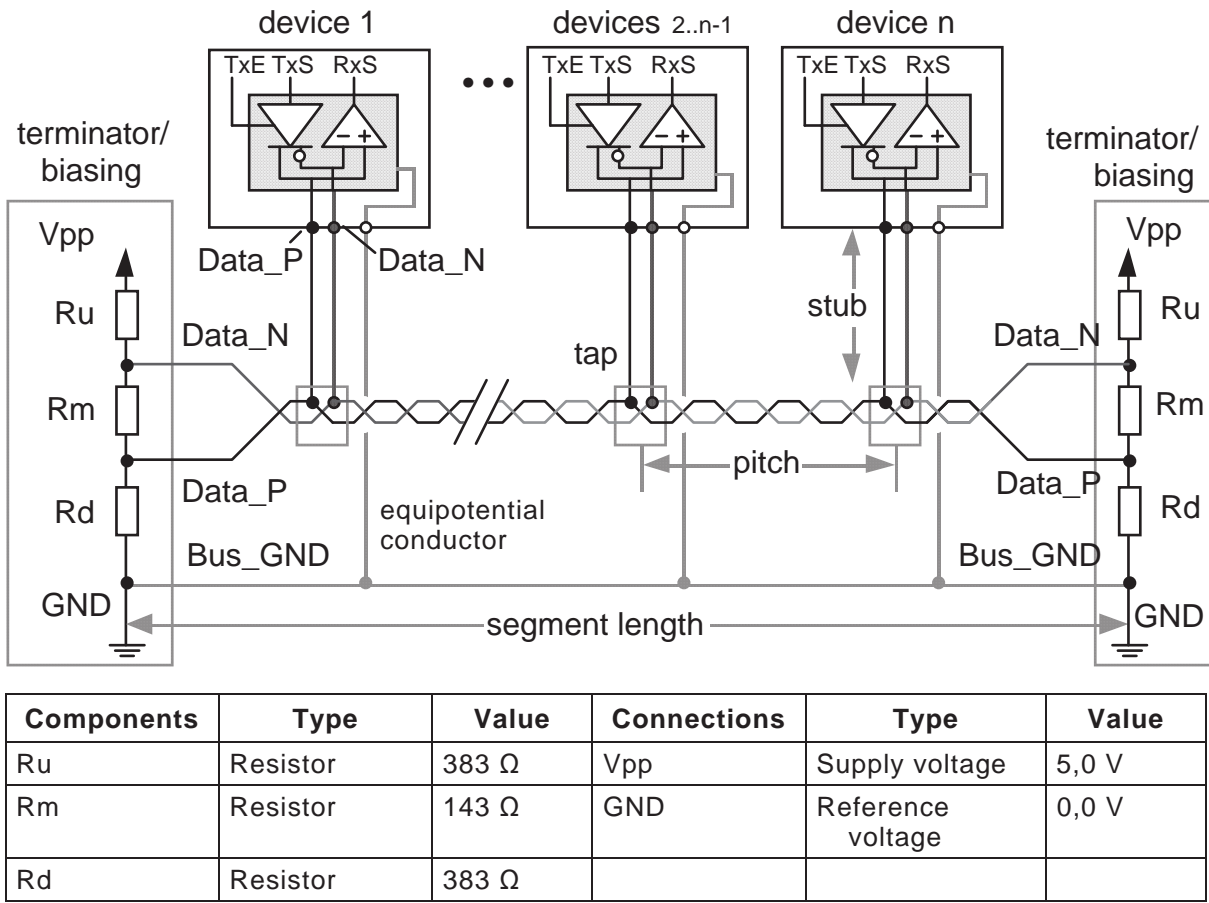


Figure 5 – Example of ESD segment

## 4.4.2 ESD configuration rules

### 4.4.2.1 ESD transmission medium

An ESD line shall consist of a pair of conductors, preferably twisted and shielded, with an equipotential conductor running in parallel.

### 4.4.2.2 ESD identification

The two conductors of a line shall be named Data\_P and Data\_N.

Throughout a line, the conductors Data\_P and Data\_N shall be marked distinctly.

The identity of the conductors shall be maintained at all connection or splicing points.

The equipotential conductor shall be called Bus\_GND and marked distinctly.

### 4.4.2.3 ESD characteristic impedance

The characteristic impedance of the unloaded line shall be  $120,0 \Omega \pm 10 \%$  at 1,0 BR.

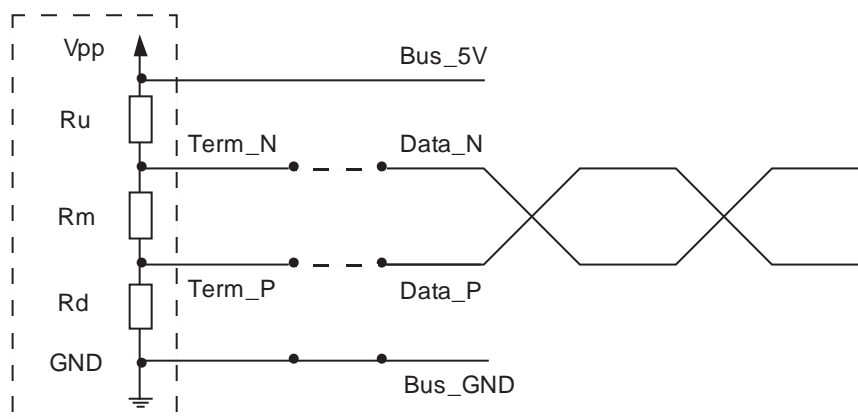
### 4.4.2.4 ESD terminator

An ESD segment shall be terminated electrically at each end by a terminator.

The terminator shall present an impedance of  $120,0 \Omega \pm 2 \%$  measured at 1,0 BR.

The terminator shall bias the wire pair, so that the potential of the Data\_P wire is lower by at least  $0,750 \text{ V} \pm 10 \%$  with respect to the Data\_N wire when no transmitter is active.

**EXAMPLE** The terminator shown in Figure 6 presents an impedance of  $120,5 \Omega$  and biases the line with about 0,786 V. Note that the Vpp source shall have a very low inner impedance in the frequency range of 0,5 BT until 2 BT in order that the equivalent terminator impedance seen by the line is within the specified tolerance. Resistors are belonging from the E96 series 1 % tolerance.



Components	Type	Value	Connections	Type	Value
Ru	Resistor	$383 \Omega$	Vpp	supply voltage	5,0 V
Rm	Resistor	$143 \Omega$	GND	reference voltage	0,0 V
Rd	Resistor	$383 \Omega$			

Figure 6 – Example of terminator

#### 4.4.2.5 ESD attenuation due to the medium

The total voltage attenuation between any two devices located on the same line shall not exceed 8,0 dB.

For this test, the attenuation is measured by applying at one terminator a sinusoidal signal of differential amplitude of  $4,0 \text{ Vpp} \pm 10 \%$  ( $1,414 \text{ V r.m.s.}$ ) at 2,0 BR, and by measuring the signal at the other terminator.

#### 4.4.2.6 ESD jitter due to devices, connectors and cabling

A line, with all devices and connectors in place, terminated by its characteristic impedance shall add no more than  $\pm 0,1 \text{ BT}$  of edge jitter, referenced to the idealised zero-crossings.

For this test, jitter is measured by driving the line at one terminator by a source of differential amplitude  $4,0 \text{ Vpp} \pm 10 \%$ , centred on 0,0 V, through a source impedance of  $22,0 \Omega \pm 10 \%$  and generating as signal a pseudo-random sequence of "0" and "1" Manchester symbols with a repetition period of at least 511 bits, the signal being measured at the other terminator.

NOTE 1 Interference and reflection due to impedance mismatches between the line sections, stubs, connectors or load clustering can introduce jitter in the timing of the zero-crossings.

NOTE 2 This test method is specified in ISO/IEC 8802-3.

#### 4.4.3 ESD section specifications

The ESD specifications apply to two kinds of sections:

- a) the backplane section is used in enclosures, such as racks and cabinets, where no galvanic separation is required, over distances of up to 20,0 m;
- b) the cable section is used to connect devices outside of enclosures, with or without galvanic separation. In case galvanic separation is used, distances of up to 200,0 m can be covered.

##### 4.4.3.1 ESD backplane section

If a backplane bus is used as a medium, the following specifications apply.

The maximum extension of a stub shall be 10,0 cm, measured from the track to the transceiver input.

The pitch between adjacent taps shall be not less than 2,0 cm to avoid load clustering.

##### 4.4.3.2 ESD cable section

If a cable section is used as a medium, the specifications of the EMD cable shall apply.

In addition, the following specifications hold:

- a) a cable used for ESD shall carry at least one equipotential wire per line pair;
- b) the equipotential wire of an ESD cable shall be identified as A.Bus\_GND (for Line\_A) and B.Bus\_GND (for Line\_B, in case two redundant lines are used).

NOTE The cross-sectional area of the equipotential wire may be different from that of the data wire pair.

#### 4.4.4 ESD shielding

The shield of the cable shall be connected to the casing of each connector, which shall be of conductive material.

The casing of the connector shall make electrical contact with the receptacle of the device when inserted.

The shield of the cable shall not be used in place of Bus\_GND.

#### 4.4.5 ESD medium-dependent interface

NOTE The Medium-dependent interface is described for a double-line segment, even when this option is not used.

##### 4.4.5.1 ESD attachment

A device shall be attached to Line\_A and/or Line\_B through independent connection points, identified as shown in Figure 7 as:

- A.Data\_P and A.Data\_N; respectively
- B.Data\_P and B.Data\_N.

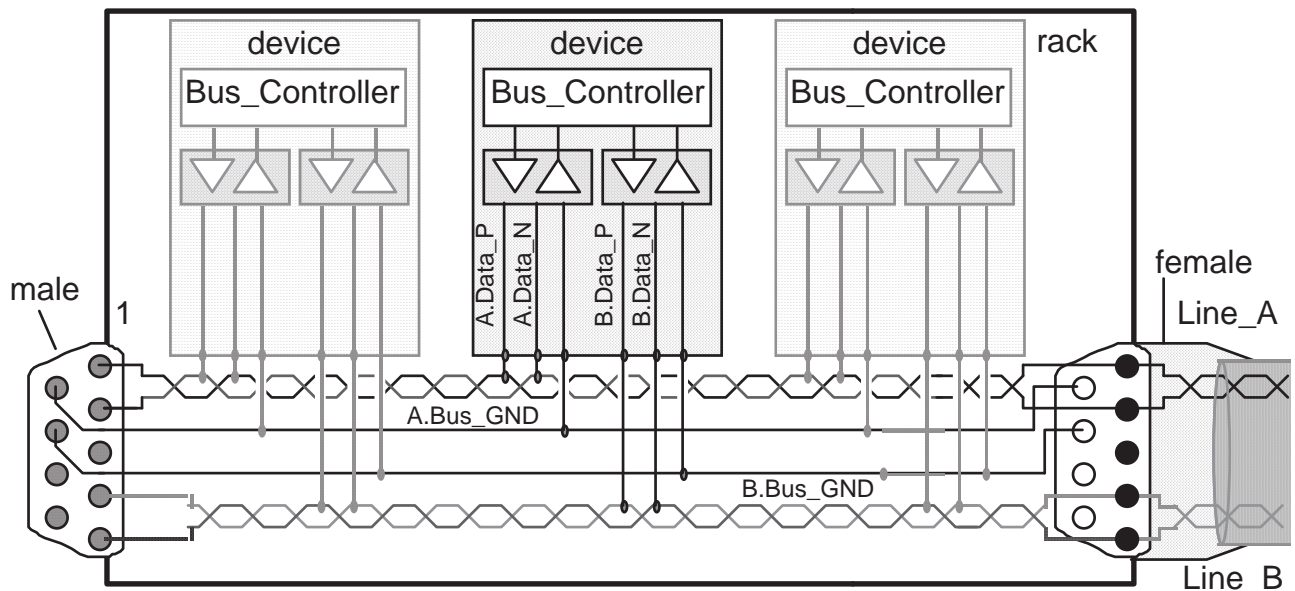


Figure 7 – ESD backplane section (double-line)

NOTE 1 This scheme encompasses a redundancy scheme in which the two redundant lines use independent twisted pairs sharing the same cable and connector. This scheme offers protection against wire, pin contact and transceiver failure, but not against cable rupture or connector removal. Field experience shows that the risks of the latter are minor.

NOTE 2 In applications where separate connectors and cables are required, the connector pin-out should be the same as for the two-connector attachment and it should have the same polarity (male or female) as a single line section.

##### 4.4.5.2 ESD connector

Where interchangeability is required, devices shall be attached to the cable as follows:

- each device shall be inserted into the cable by two 9-pin Sub-D 9 connectors using metric screws (IEC 60807), called Connector\_1 and Connector\_2;
- the connector shall have a shielded, conductive casing connected to the cable shield and which makes an electrical contact with the receptacle when fastened;
- it shall be possible to connect and fasten cable connectors together, so as to provide cable and shield continuity;
- the connector (male or female) shall have the pin assignment indicated in Table 6;



- e) the connector receptacles shall be marked with “MVB-S1” and “MVB-S2” to identify the ESD medium;
- f) a cable section or a backplane section shall bear a male connector at one extremity and a female connector at the other extremity;
- g) the connectors shall have the polarity and arrangement shown in Figure 8;
- h) Connector\_1 shall use the male connector on the device and the female on the cable;
- i) Connector\_2 shall use the female connector on the device and the male on the cable;
- j) a device or a rack shall provide through the connector for each line a power supply of 5,0 V ± 5 % with a minimum supply capability of 20,0 mA and limited to 300,0 mA, whose ground is connected to the equipotential conductor;

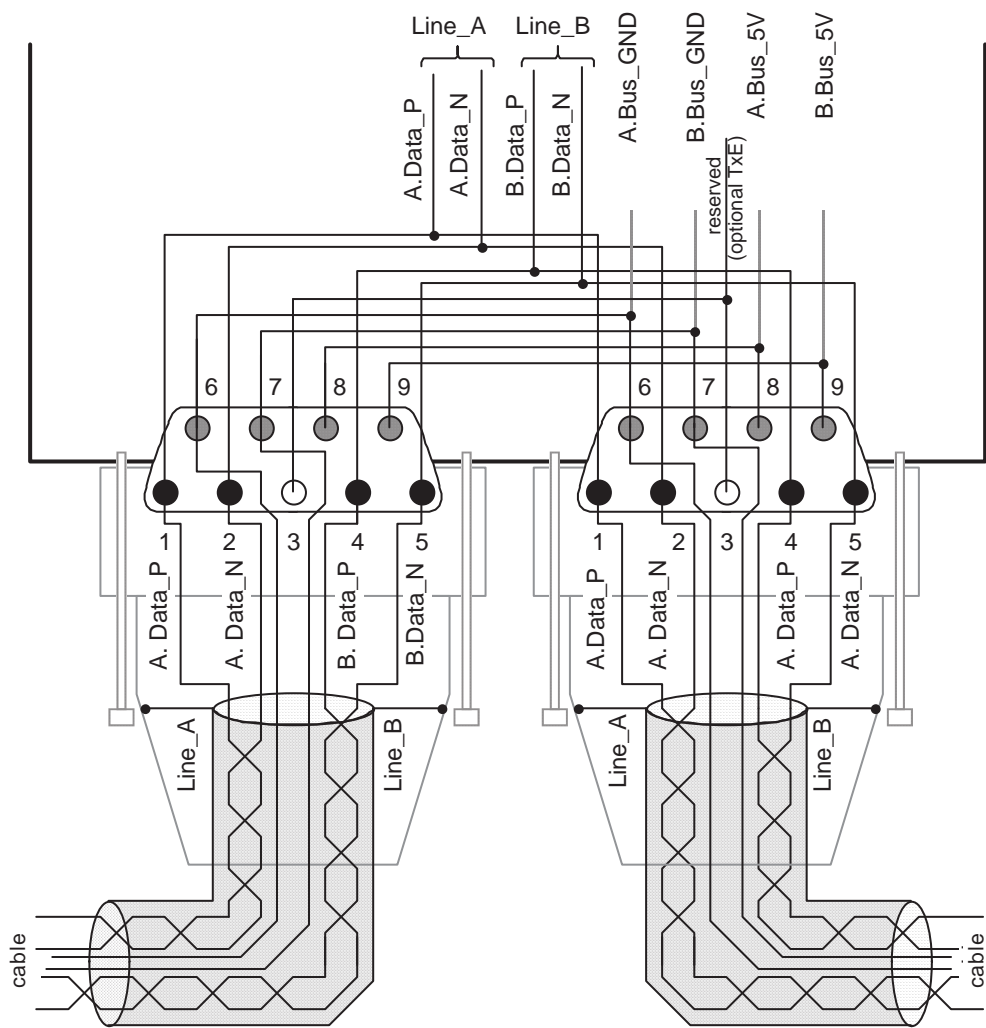


Figure 8 – ESD connector arrangement

Table 6 – Pin assignment for the ESD connector

1	A.Data_P, positive wire Line_A	6	A.Bus_GND, ground Line_A
2	A.Data_N, negative wire Line_A	7	B.Bus_GND, ground Line_B
3	TxE, see 4.3.3 (optional)	8	A.Bus_5V, positive supply Line_A
4	B.Data_P, positive wire Line_B	9	B.Bus_5V, positive supply Line_B
5	B.Data_N, negative wire Line_B		

NOTE 1 It is recommended to arrange the two connectors side by side on the front plate of a device, Connector\_1 on the left and Connector\_2 on the right, to thread the wiring neatly.

NOTE 2 It may be necessary to ensure that the connector case is isolated from the device case if connection of the device cases is not advisable, for instance when large stray currents are expected.

#### 4.4.5.3 ESD connector of the terminator

The terminator shall be included into a connector, to be plugged into the empty receptacle of a device located at the end of the segment, according to the wiring shown in Figure 9:

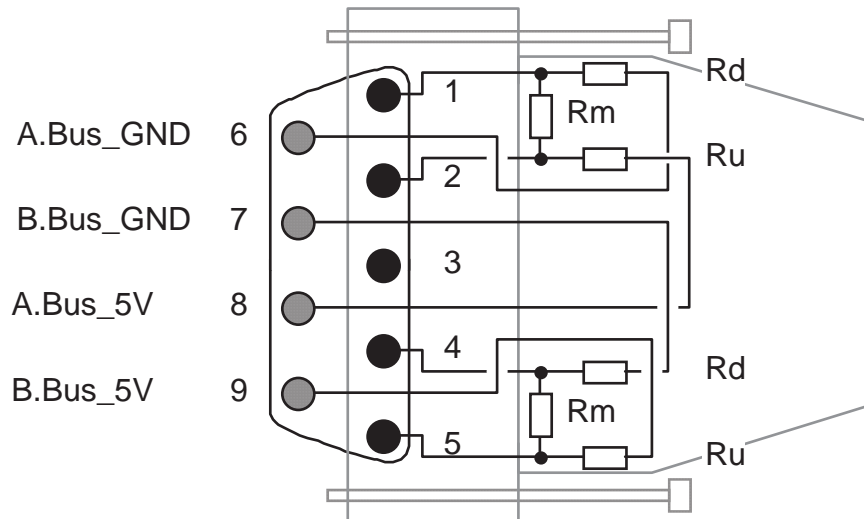


Figure 9 – ESD terminator connector arrangement

The connector containing the terminator for ESD shall be marked with a “MVB-S1”, respectively “MVB-S2”.

NOTE There are two kinds of terminator connector, male and female. The connectors for terminating ESD and EMD are distinct.

#### 4.4.6 ESD Line\_Unit specifications

##### 4.4.6.1 ESD conventions (guideline)

The characteristics of each device are measured at the points where the line is attached to the device, Data\_P, Data\_N and Bus\_GND, as shown in Figure 7.

When measuring a transmitter, the circuit of the receiver is in the normal receiving state. When measuring a receiver, the circuit of its transmitter is in a high impedance state.

If the device is attached through connectors, these are included into the measurement.

##### 4.4.6.2 ESD insertion losses of a device

The load presented by a device shall conform to ISO/IEC 8482 (RS-485).

NOTE ISO/IEC 8482 specifies this value as 12 k at 1,0 BR.

#### 4.4.7 ESD signal wave form

The medium shall take two distinct levels:

- a) a HIGH level in which voltage  $U_p$  on Data\_P is greater than voltage  $U_n$  on Data\_N, which shall correspond to a HIGH level of the TxS or RxS signals;

- b) a LOW level in which voltage  $U_p$  on Data\_P is smaller than voltage  $U_n$  on Data\_N, which shall correspond to a LOW level of the TxS or RxS signals.

Biasing shall ensure that the line takes the LOW level when it is not driven.

EXAMPLE The start of a frame as seen at the transmitter (TxS and TxE), on the line ( $U_p - U_n$ ) and at the receiver (RxS), without considering timing delays, is shown in Figure 10.

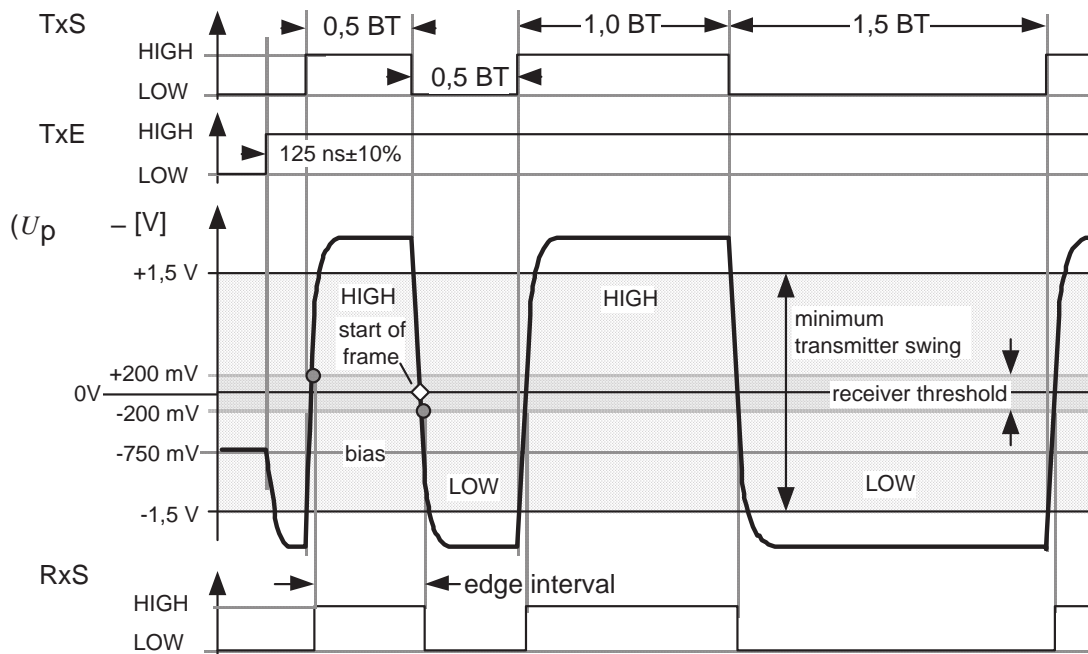


Figure 10 – Example of start of frame (ESD)

NOTE The first HIGH-to-LOW zero-crossing in the frame defines the start of the frame.

#### 4.4.8 ESD transmitter

##### 4.4.8.1 ESD transmitter signal

A transmitter shall conform to ISO/IEC 8482 (RS-485), with the following restrictions:

- the rise time of the signal (10 % – 90 %) shall be less than 0,03 BT (20,0 ns at 1,5 Mb/s), when driving a load of 54,0  $\Omega$  in parallel with 50,0 pF;
- the transmitter shall provide a low impedance differential voltage source with two active levels:
  - HIGH, when the voltage difference ( $U_p - U_n$ ) is within:
    - +1,5 V < ( $U_p - U_n$ ) < +5,0 V when driving a resistive load of 54,0  $\Omega$ , and
    - +1,5 V < ( $U_p - U_n$ ) < +6,0 V when unloaded;
  - LOW, when the voltage difference ( $U_p - U_n$ ) is within:
    - 1,5 V > ( $U_p - U_n$ ) > -5,0 V when driving a resistive load of 54,0  $\Omega$ , and
    - 1,5 V > ( $U_p - U_n$ ) > -6,0 V when unloaded.

NOTE Since this specification is stricter than ISO/IEC 8482, care should be taken in selecting commercial transceivers. This specification is fulfilled by transmitters which conform to IEC 61158-2.

##### 4.4.8.2 ESD transmitter jitter

When sending a frame, the jitter between two consecutive edges between Start\_Bit and End\_Delimiter shall not exceed 10,0 ns.

#### 4.4.8.3 ESD transmitter start of frame

Before sending the Start\_Bit, the transmitter shall drive the line actively towards the LOW state for at least  $0,125 \mu\text{s} \pm 0,010 \mu\text{s}$ , as shown in Figure 10.

NOTE Driving the line LOW before the Start\_Bit is good practice to achieve a sharper starting edge at the receiver.

#### 4.4.8.4 ESD transmitter end of frame

After sending the last bit of a frame, the transmitter shall drive the line actively towards the LOW state for at least  $0,125 \mu\text{s}$  and at most 1,0 BT, as shown in Figure 11.

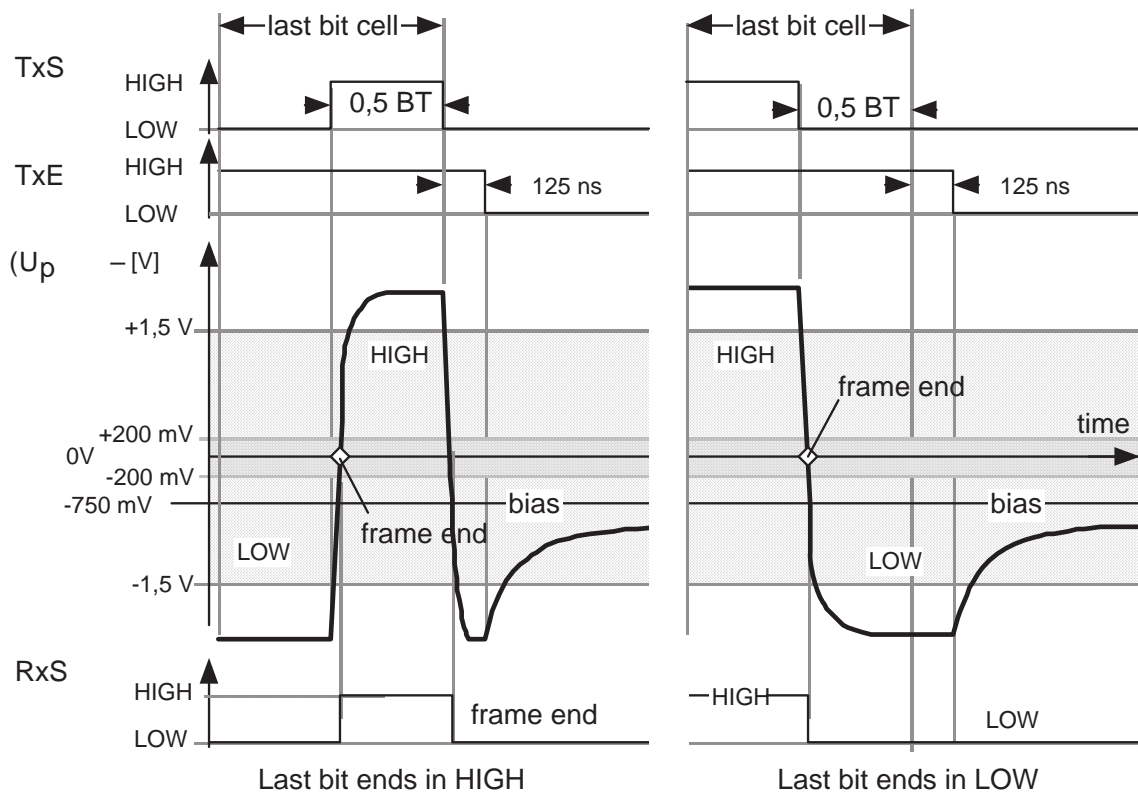


Figure 11 – End of an ESD frame (both cases)

#### 4.4.9 ESD receiver

A receiver shall conform to ISO/IEC 8482 (RS-485), with the following restrictions:

- depending on the differential line voltage, a receiver shall generate two levels on its RxS output:
  - a HIGH level if the voltage difference ( $U_p - U_n$ ) is higher than  $+0,200 \text{ V}$ , when the line is driven toward the HIGH level;
  - a LOW level if the voltage difference ( $U_p - U_n$ ) is lower than  $-0,200 \text{ V}$ , when the line is driven toward the LOW level or when the line is not driven and only the bias voltage appears;
- the receiver shall have a hysteresis of at least  $0,050 \text{ V}$ , but of no more than  $0,200 \text{ V}$ ;
- the receiver shall operate correctly in the presence of the common mode voltage specified in RS-485 with respect to the Bus\_GND line.

## 4.5 Electrical Middle Distance medium (choice)

The specifications of the Electrical Middle Distance medium allow segments with up to 32 devices over a distance of up to 200 m when using the specified medium. The actual distance and the number of devices is limited by the distortion caused by cables, connectors and devices, by the ground quality and by the disturbance levels. Although the EMD has been designed for transformer-coupling, other attachments are possible.

### 4.5.1 EMD topology

The EMD medium shall consist of two conductors, chained from device to device, and terminated at each end, as shown in Figure 12.

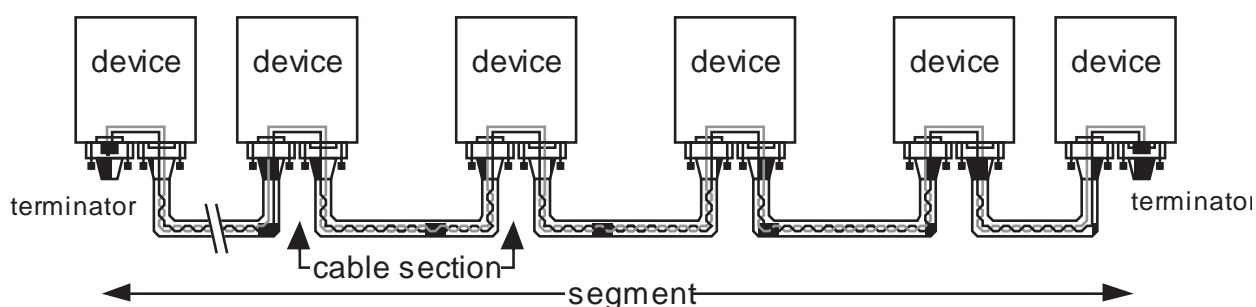


Figure 12 – EMD medium

### 4.5.2 EMD configuration rules

#### 4.5.2.1 EMD transmission medium

The EMD medium shall be a shielded, twisted pair of wires.

The transmitted signal shall be a differential, polarity-sensitive voltage between these wires.

#### 4.5.2.2 EMD identification

Throughout a segment, the wires shall be marked distinctly.

The identity of the wires shall be maintained at all connection or splicing points.

#### 4.5.2.3 EMD attenuation due to the medium

The total signal attenuation over the medium between any two devices, expressed as the ratio of the voltage at the transmitting device to the voltage at the receiving device, shall not exceed 8,0 dB, measured at a frequency between 0,5 BT and 2,0 BT.

NOTE Attenuation limits the length of a segment for a given number of devices. With a cable attenuation of 15,0 dB/km and an insertion loss of 0,15 dB/device at 1,5 MHz, a distance of 200 m can be covered.

#### 4.5.2.4 EMD jitter due to devices, connectors and cabling

A line, with all devices and connectors in place, terminated by its characteristic impedance shall add no more than  $\pm 0,1$  BT of edge jitter, referenced to the idealised zero-crossings.

For this test, the jitter is measured by driving the line at one terminator by a source of differential amplitude  $4,0 \text{ V}_{pp} \pm 10 \%$ , centred on 0,0 V, through a source impedance of  $22,0 \Omega \pm 10 \%$  and generating as signal a pseudo-random sequence of "0" and "1" Manchester symbols with a repetition period of at least 511 bits, the signal being measured at the other terminator.

NOTE 1 Interference and reflection due to impedance mismatches between the line sections, stubs, connectors or load clustering can introduce jitter in the timing of the zero-crossings.

NOTE 2 This test method is specified in ISO/IEC 8802-3.

NOTE 3 It is a good practice to cluster no more than 4 devices over a cable length of 10,0 m and to include at least 10,0 m of cable from such a cluster to the next cluster.

#### **4.5.3 EMD terminator**

A line shall be terminated at each end by a terminator presenting an impedance of  $Z_w = 120,0 \Omega \pm 2 \%$  and a phase angle of less than 0,087 rad over the frequency range of 0,5 BR to 2,0 BR.

#### **4.5.4 Cable section**

The cable specifications apply to both ESD and EMD, allowing to use only one type of cable. The particularities of the ESD cable are mentioned in 4.4.3.2.

##### **4.5.4.1 Mechanical**

All sections shall consist of a shielded, jacketed cable carrying at least one twisted wire pair.

The cable shall have no less than 12 twists per metre.

NOTE A cross-sectional area of each wire between 0,34 mm<sup>2</sup> (AWG 22) and 0,56 mm<sup>2</sup> (AWG 20) is recommended.

##### **4.5.4.2 Marking**

The individual wires of the twisted pair shall be identified:

- a) for Line\_A (even if only one pair is used) as: A.Data\_P and A.Data\_N;
- b) for Line\_B (in case two line pairs are used) as: B.Data\_P and B.Data\_N.

The individual wires of the cable shall be marked distinctly.

Where the cable carries two lines, the redundant twisted pairs shall be marked distinctly.

NOTE The two pairs of wires may be geometrically laid out as a quadruple, in this case, diagonal wires shall form one pair.

##### **4.5.4.3 Characteristic impedance**

The bus sections shall present a differential characteristic impedance of  $Z_w = 120,0 \Omega (\pm 10 \%)$  to the data wires, measured with a sinusoidal signal at a frequency between 0,5 BR and 2,0 BR.

##### **4.5.4.4 Cable attenuation**

The cable shall attenuate a sinusoidal signal sent between the data wires by less than 15,0 dB/km at 1,0 BR, and by less than 20,0 dB/km at 2,0 BR.

##### **4.5.4.5 Distributed capacitance**

The differential (wire-to-wire) distributed capacitance of the cable shall not exceed 46,0 pF/m at 1,0 BR.

##### **4.5.4.6 Capacitive unbalance to shield**

The capacitive unbalance to shield shall not exceed 1,5 pF/m at 1,0 BR.

#### 4.5.4.7 Crosstalk rejection

Where two pairs of wires are carried in the same cable, the signal rejection from one redundant pair to the other shall be superior to 45,0 dB in the range from 0,5 BR to 2,0 BR.

NOTE Cable for double-line segments may carry two wire pairs in the same cable.

#### 4.5.4.8 Shield quality

The transfer impedance of the cable, shall, at 20,0 MHz, be less than 0,020  $\Omega$ /m.

The differential transfer impedance of the cable shall be less than 0,002  $\Omega$ /m.

#### 4.5.4.9 Cable connector quality

All cable connections shall provide continuity of wires and shielding, with a resistance of less than 0,010  $\Omega$ .

The transfer impedance of the connector, measured at 20,0 MHz, shall be less than 0,020  $\Omega$  between one pin and shield, respectively 0,002  $\Omega$  between two pins.

NOTE 1 These requirements do not apply to connectors between vehicles.

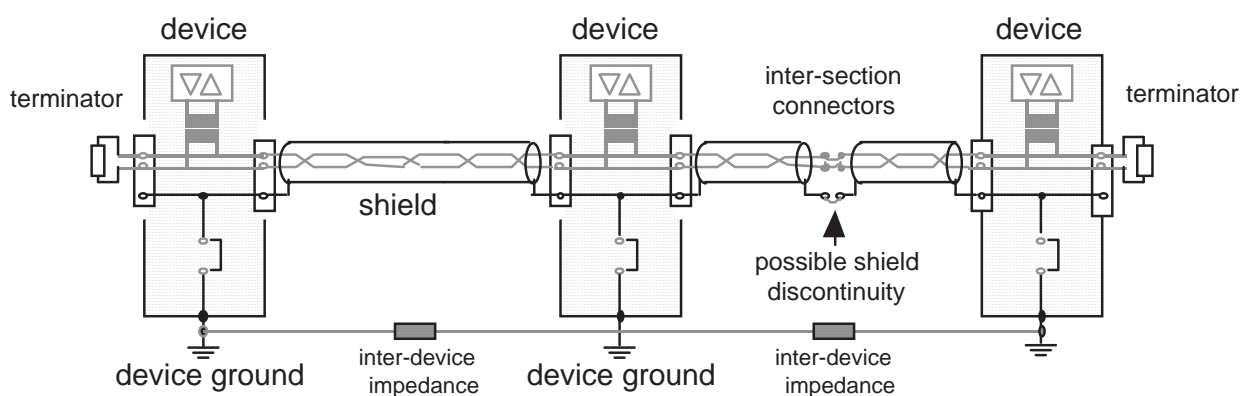
### 4.5.5 EMD shielding

#### 4.5.5.1 EMD shield continuity

A device shall connect the shields of the two cable sections to which it is attached.

When a device is removed, it shall be possible to connect the shields together, for example through the connectors.

The device shall provide means to connect the shield to its device ground so that a suitable place can be found for earthing the shield, or to implement a grounding concept in which all casings are connected together, as shown in Figure 13.



**Figure 13 – Shielding (single-line segment)**

NOTE The casing of a device is not necessarily connected to the receptacle of its connectors, although this is the recommended arrangement.

#### 4.5.5.2 EMD shield discontinuity

In applications where large voltage differences between devices are expected, cable sections may be connected at selected places without establishing shield continuity, as shown in Figure 13.

At these places, non-standard cables, for instance using double shielding, may be used to comply with the EMC requirements.

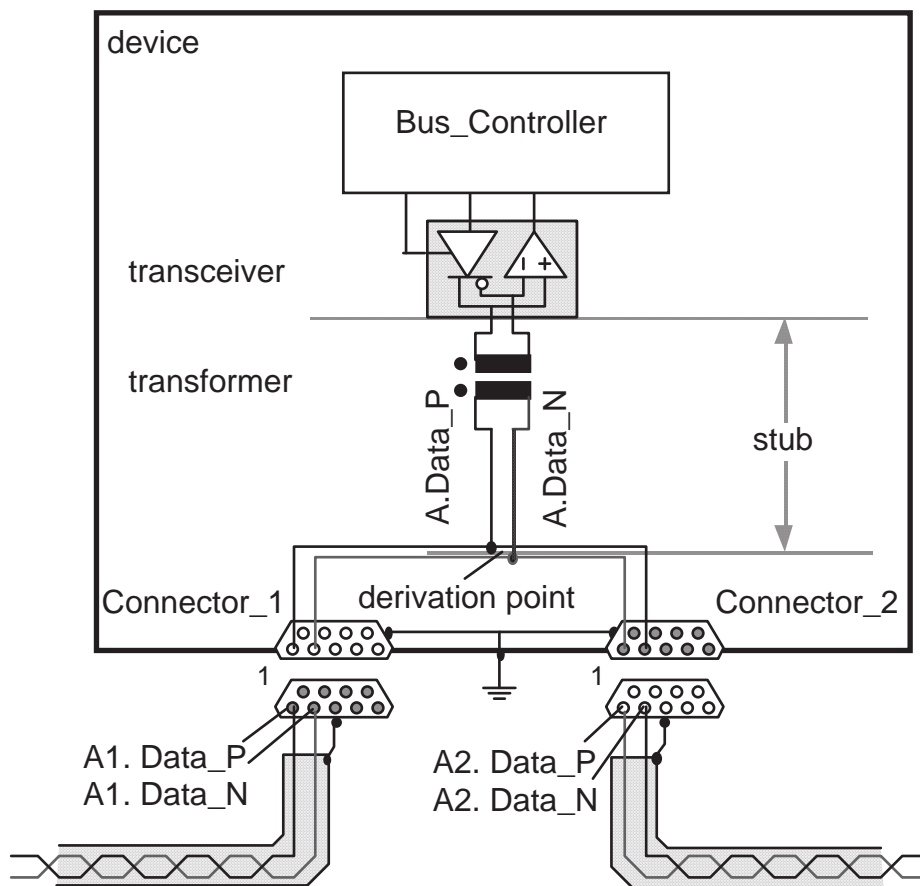
NOTE This exception accounts for jumper cable or automatic couplers between vehicles where high stray currents are expected.

#### 4.5.6 EMD medium-dependent interface

##### 4.5.6.1 EMD single line attachment

A device shall be attached to the line by a passive tap.

The length of the stub, measured from the derivation point (see Figure 14) to the transceiver, shall not exceed 10,0 cm.



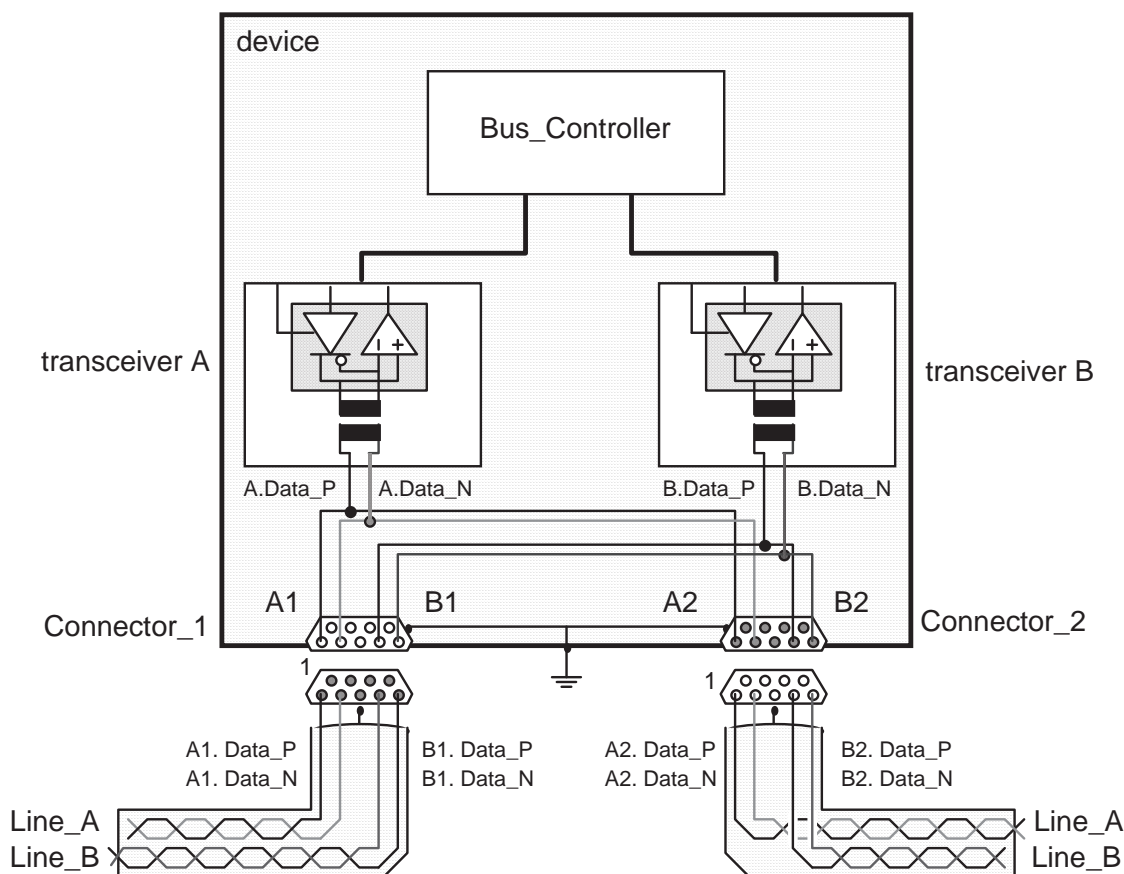
**Figure 14 – Single-line device attachment**

NOTE To conform to this specification, it is advisable that the device is attached by two connectors, as shown in Figure 14.

##### 4.5.6.2 EMD double-line attachment (option)

A device with double-line attachment shall be capable of being attached to both a single-line or to a double-line EMD segment, as shown in Figure 15.





NOTE 1 This scheme encompasses a redundancy scheme in which the two redundant lines use independent twisted pairs sharing the same cable and connector. This scheme offers protection against wire, pin contact and transceiver failure, but not against cable rupture or connector removal. Field experience shows that the risks of the latter are minor.

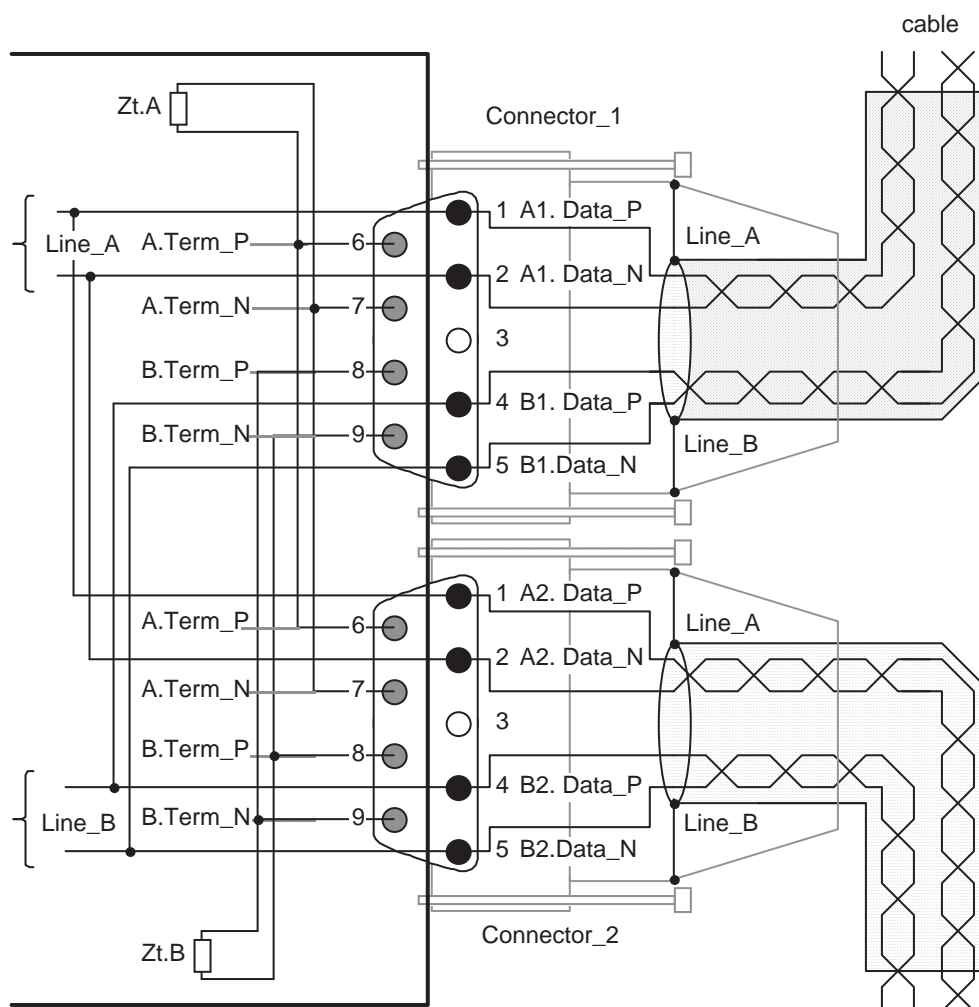
NOTE 2 In applications where the redundant lines should be run over separate connectors, in a four connector configuration, the connector pin-out should be the same as for the two-connector attachment and it should have the same polarity (male or female) at one extremity of a cable section.

**Figure 15 – Double-line device attachment to EMD**

#### 4.5.6.3 MD connector

Where interchangeability is required, devices shall be attached to the cable as follows:

- each device shall be inserted into the cable by two 9-pin Sub-D 9 connectors using metric screws (IEC 60807), called Connector\_1 and Connector\_2;
- the connectors shall have a shielded, conductive casing connected to the cable shield and which makes an electrical contact with the receptacle when fastened;
- it shall be possible to connect and fasten cable connectors together, so as to provide cable and shield continuity;
- the connector (male or female) shall have the pin assignment shown in Table 7;
- the connector receptacle for EMD shall be marked with "MVB-M1" (for Connector\_1) and "MVB-M2" (for Connector\_2) to identify the EMD medium;
- a cable section shall bear a male connector at one extremity and a female connector at the other extremity;
- the connectors shall have the following polarity and arrangement, as shown in Figure 16;
- Connector\_1 shall use the male connector on the device and the female on the cable;
- Connector\_2 shall use the female connector on the device and the male on the cable.



**Figure 16 – EMD connectors arrangement**

**Table 7 – Pin assignment for the EMD connector**

1	A.Data_P positive wire of Line_A	6	A.Term_P, positive pole of Terminator Line_A
2	A.Data_Nnegative wire of Line_A	7	A.Term_N, negative pole of Terminator Line_A
3	reserved for TxE. see 4.3.3 (option)	8	B.Term_P, positive pole of Terminator Line_B
4	B.Data_P positive wire of Line_B	9	B.Term_N, negative pole of Terminator Line_B
5	B.Data_Nnegative wire of Line_B		

NOTE 1 It is recommended to arrange the two connectors side by side on a device, Connector\_1 on the left and Connector\_2 on the right, to thread the wiring neatly.

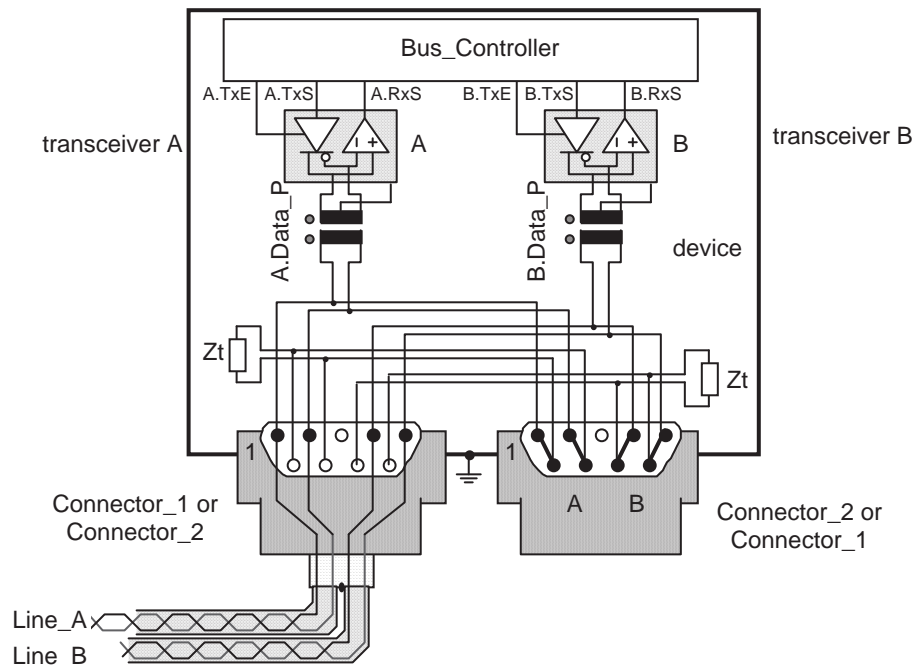
NOTE 2 It may be necessary to ensure that the connector case is isolated from the device case if connection of the device cases is not advisable, for instance when large stray currents are expected.

#### 4.5.6.3.1 MD female connector (option)

Where no interchangeability is required, devices can be attached to the cable as follows: both Connector\_1 and Connector\_2 can use the female connector on the device and the male on the cable.

#### 4.5.6.4 EMD terminator strapping

To connect the terminator in the extremity device of a segment, the connector shall be strapped by connecting pin 1 to pin 6, pin 2 to 7, pin 4 to pin 8, and pin 5 to pin 9, as shown in Figure 17.



**Figure 17 – EMD terminator strapping**

The terminator connectors for EMD shall be marked with “MVB-M1” (to be plugged as Connector\_1) and “MVB-M2” (to be plugged as Connector\_2).

#### 4.5.7 EMD Line\_Unit specifications

Even when only Line\_Unit A is mentioned, this subclause also applies to Line\_Unit B.

##### 4.5.7.1 EMD insertion losses of a device

A device either:

- a) with receiver in normal operation and with transmitter in the high-impedance state; or
- b) with no power applied to either receiver or transmitter,

shall introduce less than 0,15 dB of attenuation at the frequencies of 0,5 BR and 2,0 BR.

##### 4.5.7.2 EMD galvanic separation

Between the device ground and any of the points:

- a) A1.Data\_P,
- b) A1.Data\_N,
- c) A2.Data\_P, and
- d) A2.Data\_N,

the isolation voltage and resistance shall exceed the value specified by IEC 60571.

NOTE These values are, in the present edition, 0,500 kV r.m.s. and 1,0 MΩ respectively.

#### 4.5.8 EMD signal waveform

The medium shall take two distinct levels:

- a HIGH level when the potential difference ( $U_p - U_n$ ) between Data\_P and Data\_N is positive, which shall correspond to a HIGH level of the TxS or RxS signals;
- a LOW level when the potential difference ( $U_p - U_n$ ) between Data\_P and Data\_N is negative, which shall correspond to a LOW level of the TxS or RxS signals.

The start of the frame is defined by the first HIGH-to-LOW zero-crossing in the frame.

NOTE The state of the line is undefined when the line is not driven.

EXAMPLE The start of a frame as seen at the transmitter (TxS), on the line ( $U_p - U_n$ ) and at the receiver (RxS), without considering the delays on the line or in the devices is shown in Figure 18.

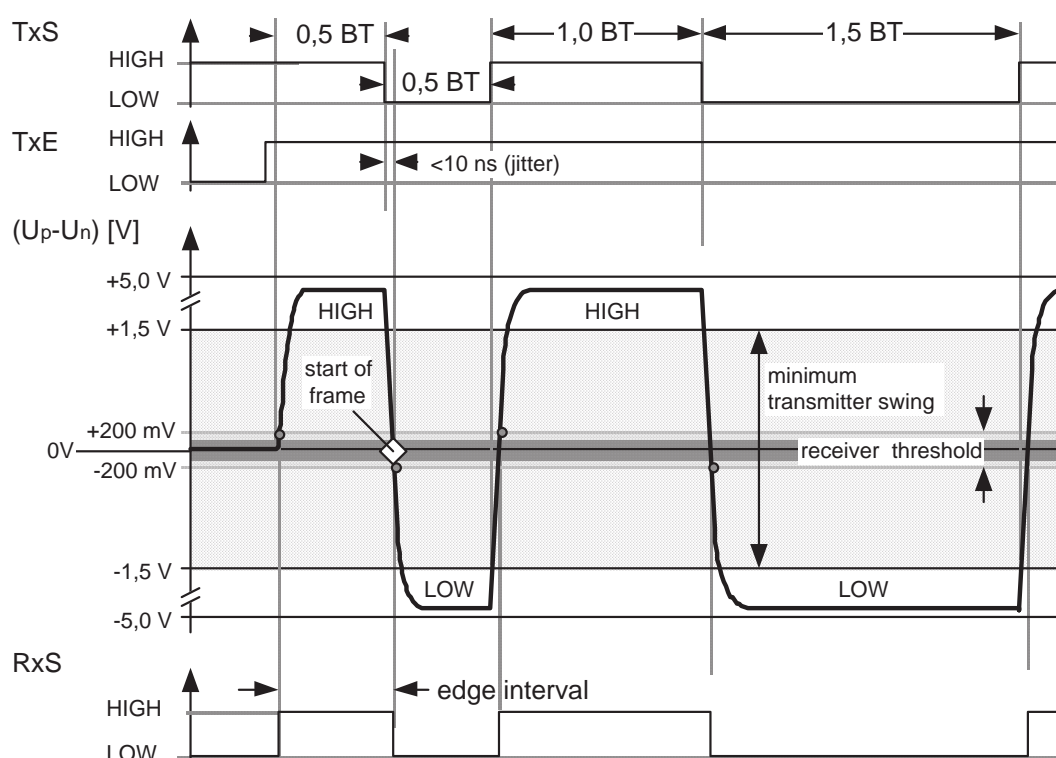


Figure 18 – Example of start of an EMD frame

#### 4.5.9 EMD transmitter specifications

##### 4.5.9.1 EMD transmitter output signal

The transmitter shall be a differential driver.

NOTE Due to the data encoding (5.1), the transmitter generates pulses which are either one bit in length (1,0 BT), a half-bit in length (0,5 BT) or one-and-a-half bits in length (1,5 BT).

These specifications apply to 0,5 BT, 1,0 BT and 1,5 BT pulses, positive or negative.

When connected to both the heavy and the light test circuit, the transmitter shall comply with the following requirements:

- a) the output signal shall be alternately positive and negative;
- b) the amplitude of the output signal shall be at least  $\pm 1,5$  V when connected to the heavy test circuit and at most  $\pm 5,5$  V when connected to the light test circuit;
- c) the difference between the positive and the negative stationary amplitude in two consecutive pulses shall not exceed 0,10 V;
- d) the slew rate of the output signal shall be more than 15 mV/ns within 0,100  $\mu$ s of the zero-crossing;
- e) the overshoot of the output signal, defined as the ratio of the maximum amplitude to the stationary amplitude shall not exceed 10 % of its stationary amplitude;
- f) the edge distortion of the output signal, defined as the time difference between the idealised and the actual zero crossing, shall not exceed  $\pm 2$  % of one Bit Time.

As an option, in order to improve signal quality, the transmitter may use a technique called Pre-Emphasis. In such a case, the transmitter shall comply with the above-listed requirements except requirement e) that is replaced by the following requirements:

- g) the ratio of the Pre-Emphasis amplitude to the steady state amplitude shall be in the range from 165 % up to 235 %;
- h) the duration of the Pre-Emphasis pulse, measured from the front edge of the waveform, shall be 330 ns;
- i) the difference between the positive and the negative stationary amplitude in two consecutive pulses shall not exceed 0,20 V.

**EXAMPLE** A typical waveform in the middle of a frame is shown in Figure 19, the upper diagram refers to the case where no pre-emphasis option is used, the lower diagram refers to the case where pre-emphasis option is used, minimum values are reported .

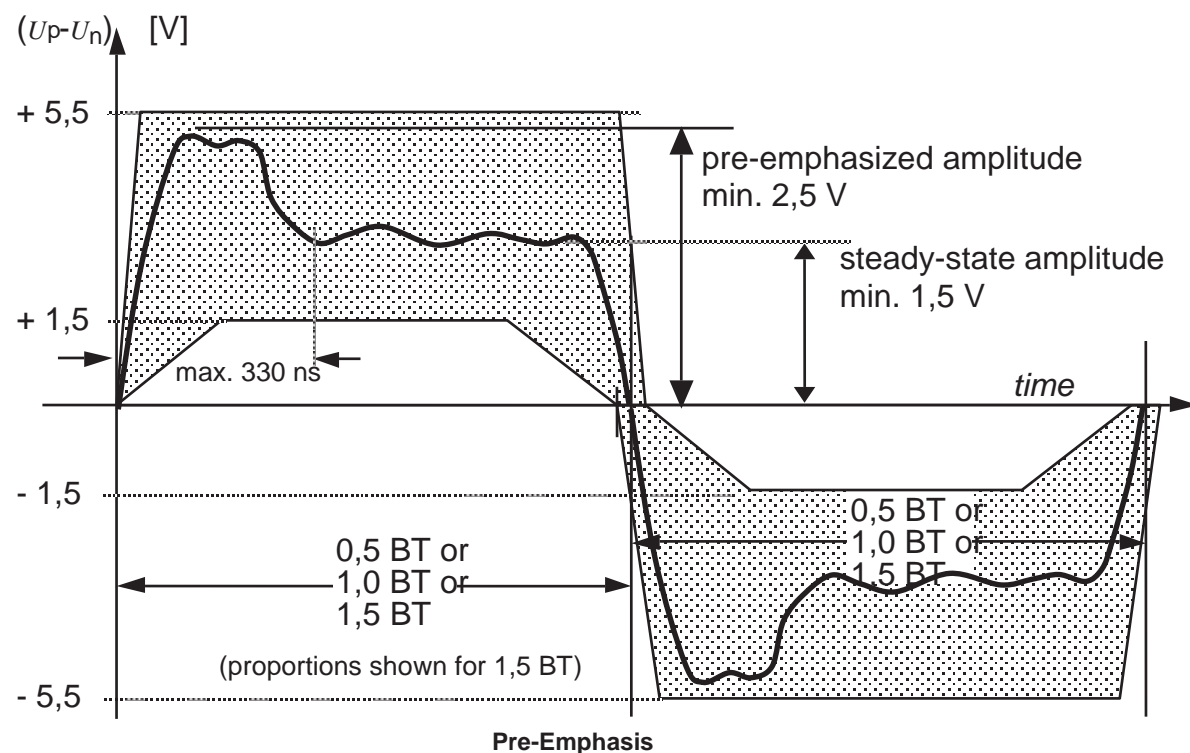
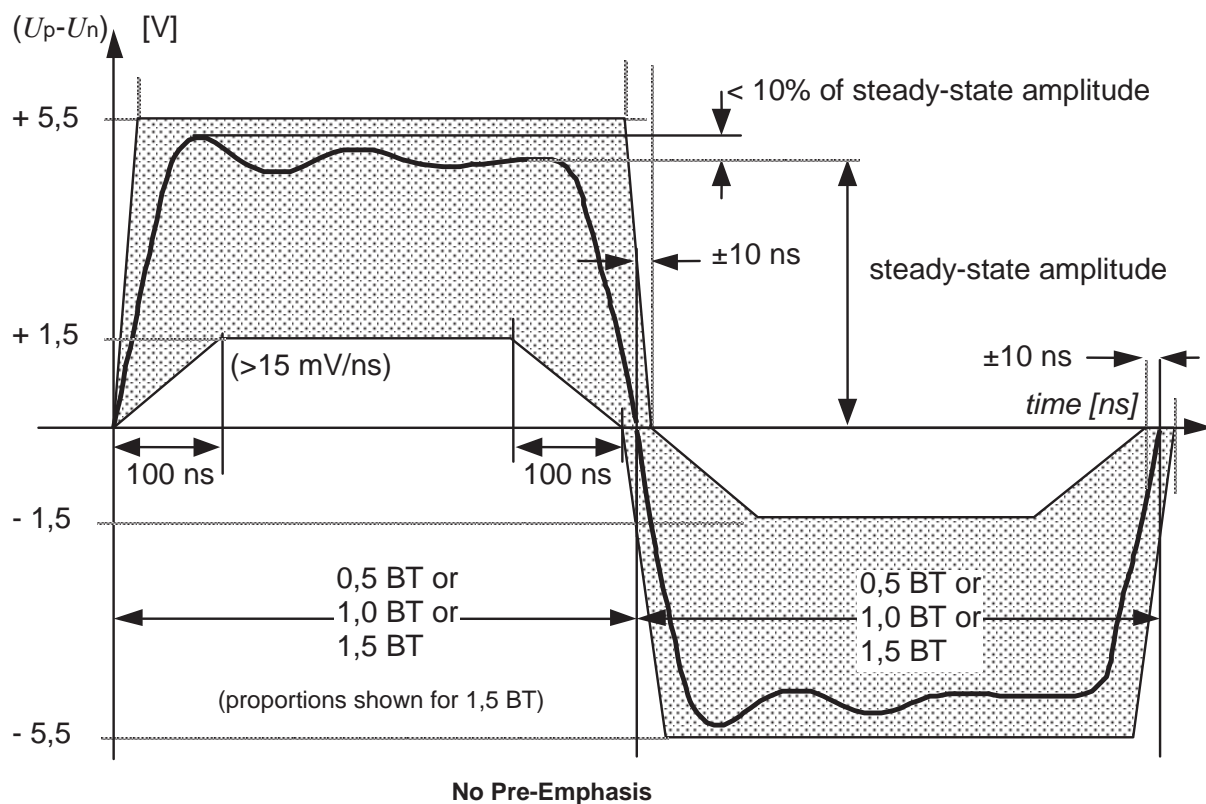


Figure 19 – Example of pulse waveform at EMD transmitter

#### 4.5.9.2 EMD transmitter noise

Any noise generated by a transmitter which is not transmitting shall not exceed a value of 5,0 mV r.m.s. over the frequency range 1,0 kHz to 4,0 BR.

#### 4.5.9.3 EMD transmitter end of frame

After the End Delimiter, the output signal shall reach 0,10 V within 0,30  $\mu$ s and at no time after the first zero-crossing exceed 0,20 V under the following test conditions:

- the transmitter transmits the longest possible frame;
- the frame data bits are a random sequence of "1" and "0";
- the frame is closed with the End Delimiter as specified in 5.1.6;
- the transmitter drives the idling test circuit;
- the output signal amplitude is greater than 4,5 V before the transmitter is disabled.

EXAMPLE The end of frame when the last data bit is a "1" is shown in Figure 20.

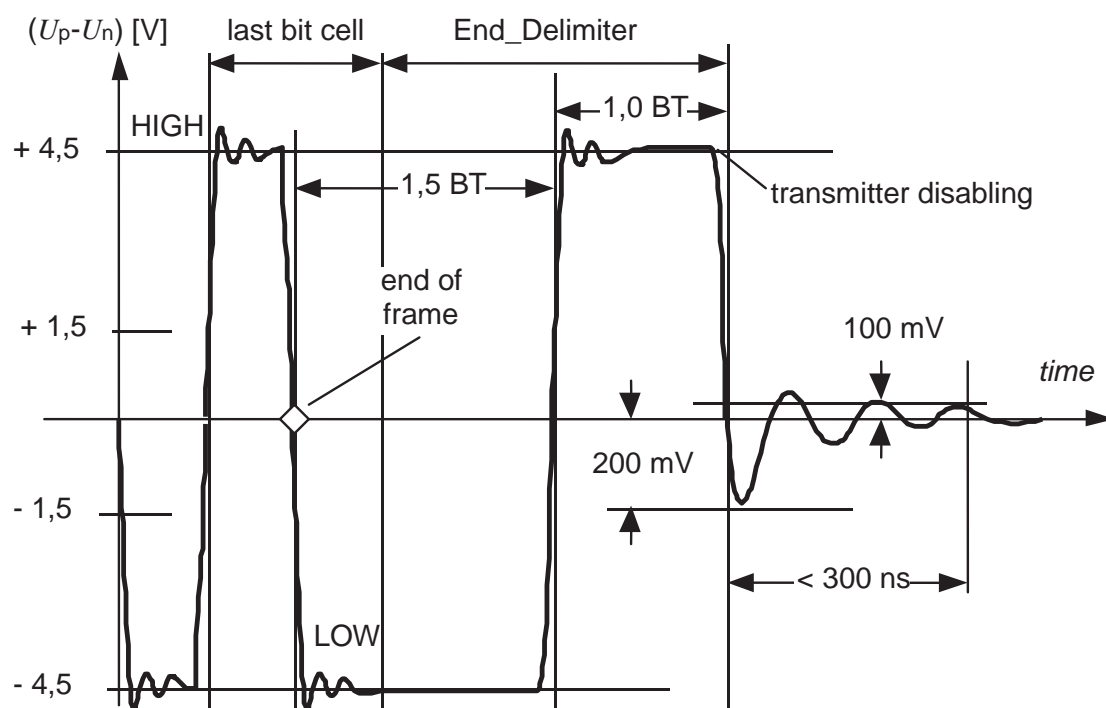


Figure 20 – Example of end of EMD frame

NOTE Line ringing after transmitter idling can be minimised by balancing the d.c. components in each bit cell.

The last zero-crossing before the End Delimiter shall define the end of frame.

#### 4.5.9.4 EMD transmitter fault tolerance

A transmitter, either when enabled or not, shall tolerate the application of the short test circuit at the connection point until thermal stability is reached, and shall resume normal operation after the short test circuit is removed.

The short-circuit current shall not exceed 1,0 A.

NOTE For conformance testing, thermal stability should be reached after 1 h.

#### 4.5.10 EMD receiver specifications

##### 4.5.10.1 EMD receiver test signal (guideline)

The characteristics of a receiver are tested by applying a test signal to the connection points and observing the frames received, either through direct access to the device, or through a device-internal response, such as response to a master frame.

To test the receiver, a test signal with the following shape, as shown in Figure 21, is applied at the connection point:

- when the amplitude of the test signal is less than 0,20 V, the slope of the test signal exceeds 4,0 mV/ns (0,20 V / 50,0 ns);
- to test sensitivity, the test signal remains above 0,30 V for a time period that starts after 0,1  $\mu$ s of the preceding zero-crossing and that lasts at least (0,5 BT – 0,2  $\mu$ s), respectively (1,0 BT – 0,2  $\mu$ s) and (1,5 BT – 0,2  $\mu$ s), whilst its peak amplitude will vary between 0,33 V and 5,00 V;
- to test insensitivity, the test signal amplitude does not exceed a prescribed value;
- to test edge distortion, the crossing of the zero voltage is varied by a prescribed value.

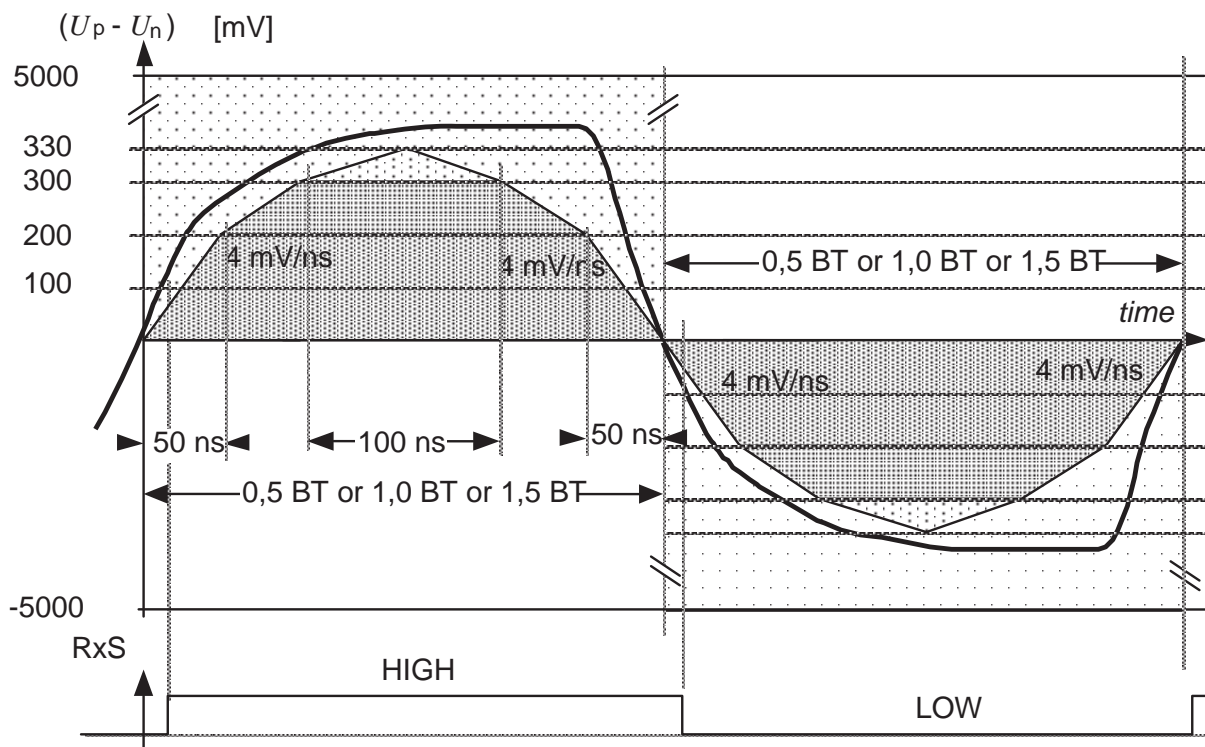


Figure 21 – EMD receiver test signal

The receiver detects a frame error as either a silence, a collision, a not valid frame or a not correct frame (incorrect frame size, incorrect Frame Data bits or incorrect CRC).

##### 4.5.10.2 EMD receiver polarity

NOTE This specification is not part of the conformance testing, since the Line\_Unit interface is not necessarily exposed. It is part of the type-testing of the transceiver.



Depending on the state of the line, a receiver shall deliver a binary RxS signal with two states, HIGH and LOW, as follows:

- a) a HIGH level if the differential line voltage ( $U_p - U_n$ ) exceeds +0,200 V;
- b) a LOW level if the differential line voltage ( $U_p - U_n$ ) becomes less than –0,200 V.

The receiver shall have a hysteresis of at least 50 mV, but of no more than 0,25 V.

As an option, in environments with high electromagnetic interference, the requirements a) and b) shall read:

- c) a HIGH level if the differential line voltage ( $U_p - U_n$ ) exceeds the positive receive threshold;
- d) a LOW level if the differential line voltage ( $U_p - U_n$ ) becomes less than the negative receive threshold.

The positive receive threshold shall be in the range 0,200 V to 0,500 V.

The negative receive threshold shall be in the range –0,200 V to –0,500 V.

It shall be noted that the maximum length of the segment may not be achieved.

#### **4.5.10.3 EMD receiver sensitivity**

A receiver receiving frames containing 64 random data bits, at the rate of 1 000 frames per second shall detect no more than three frame errors in  $3 \times 10^{+6}$  frames, when the amplitude of the test signal defined in 4.5.10.1 varies between its minimum and maximum value.

#### **4.5.10.4 EMD receiver insensitivity**

The receiver shall not detect a valid frame when the test signal amplitude is less than 0,100 V.

NOTE This threshold may be raised in certain applications.

#### **4.5.10.5 EMD receiver edge distortion**

A receiver receiving frames containing 64 random data bits, at the rate of 1 000 frames per second shall detect no more than three frame errors in  $3 \times 10^{+6}$  frames when a test signal is applied, whose edges cross the zero voltage at random within  $\pm 10\%$  of 1,0 BT around the expected crossing position.

#### **4.5.10.6 EMD receiver noise rejection**

A receiver receiving frames containing 64 random data bits, at the rate of 1 000 frames per second, and with a test signal amplitude of 0,700 V (1,400 V peak-to-peak) shall detect no more than three frame errors in  $3 \times 10^{+6}$  frames when operating:

- a) in the presence of a common-mode sinusoidal signal (applied between the device ground and both Data\_P and Data\_N) with an amplitude of 4,0 V r.m.s. in the frequency range of 65,0 Hz to 1,5 MHz or
- b) in the presence of an additive quasi-white Gaussian noise (applied between Data\_P and Data\_N) distributed over a bandwidth of 1,0 kHz to 4,0 MHz at an amplitude of 0,140 V r.m.s.

### **4.6 Optical Glass Fibre medium (choice)**

The specifications of the Optical Glass Fibre medium allow the use of optical fibres bussed through star couplers over distances up to 2 000 m when the specified medium is used. The

actual distance is limited by the distortions in fibres and connectors and by the characteristics of the repeaters and star couplers.

#### 4.6.1 OGF topology

The optical medium shall consist of a pair of fibres forming a full-duplex point-to-point link, as shown in Figure 22.

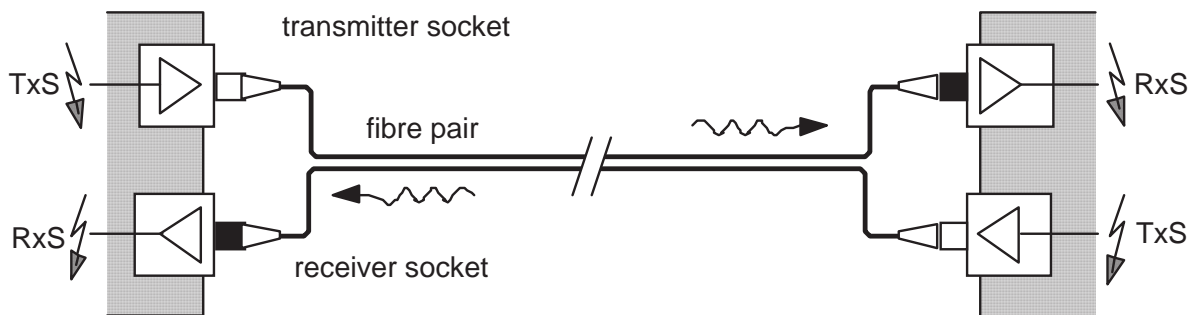


Figure 22 – Optical link

Optical fibres may be connected from device to device or may be bussed through a star coupler, preferably of the active type (4.6.7).

#### 4.6.2 OGF optical cable and fibre

The following optical cable and fibre specification applies to a mounted cable with a connector at each end of a fibre:

- the fibre shall be a multiple-mode step index silica fibre with a core of 200/230  $\mu\text{m}$  and a cladding eccentricity error of less than 5,0  $\mu\text{m}$ ;
- the attenuation of a cable of 10,0 m shall be less than 1,8 dB from connector to connector, measured at a wavelength of 840,0 nm  $\pm$  30,0 nm. For longer distances, the additional attenuation shall be less than 6,0 dB /km;
- the minimum bending radius of a cable shall be 50,0 mm;
- the cable shall conform to IEC 60571 with respect to vibration test;
- the cable shall conform to IEC 60794-1-1 (E3, E6, E8) with respect to mechanical stability;
- the cable shall conform to IEC 60332-1-1 regarding fire resistance;
- the cable shall conform to IEC 60245-1 with respect to oil resistance;
- the cable shall conform to IEC 60304 with respect to the markings.

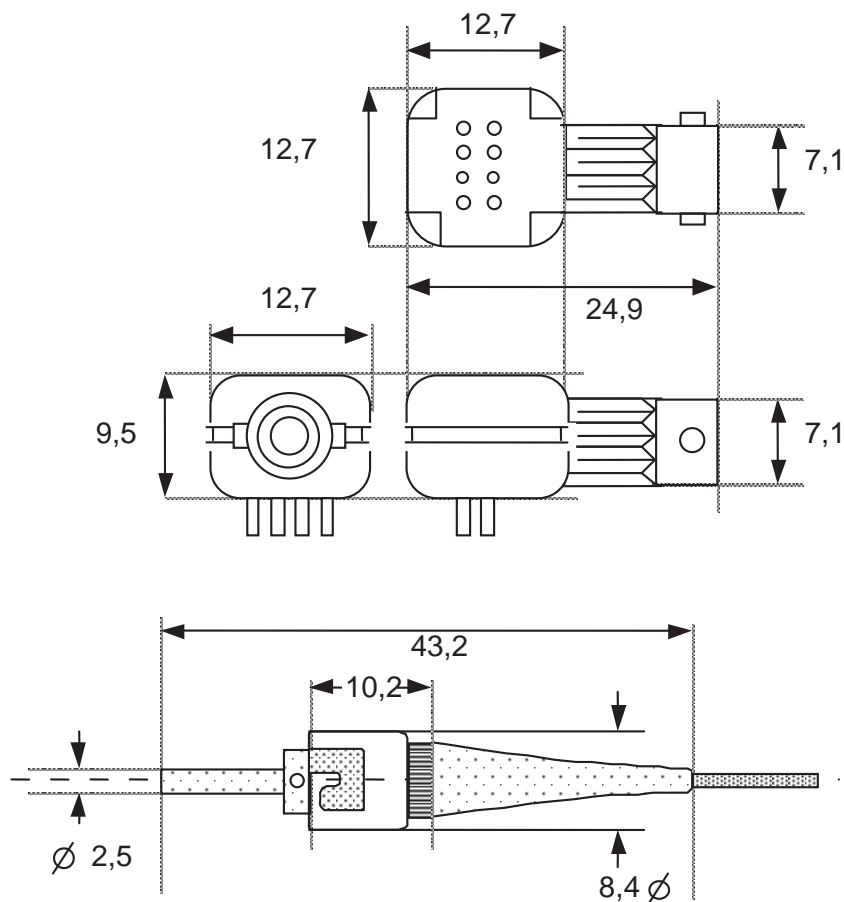
Two or more fibres may be located in the same cable.

When using multifibre cables, colour coding of boots is allowed.

NOTE It is recommended to identify distinctly the two extremities of a fibre, one for the transmitter side and one for the receiver side.

#### 4.6.3 OGF medium-dependent interface

This component should preferably integrate the transmitter and the receiver as shown in Figure 23.



**Figure 23 – Optical connector (dimensions in mm)**

The transmitter and the receiver connector shall be adequately identified, preferably:

- light grey for the transmitter;
- dark grey for the receiver.

NOTE The components and tools of the ST-connector and fibre can be used for multiple fibre assembly. The connector (ST-bayonet) for single fibres is specified with respect to the fibre type in the following standards that are withdrawn: IEC 60874-10-1, IEC 60874-10-2 or IEC 60874-10-3.

#### 4.6.4 OGF test signal (guideline)

The binary transmission uses two optical levels, the least energy level being termed LOW and the highest energy level being termed HIGH.

The LOW level corresponds to the LOW level on RxS or TxS, the HIGH level corresponds to the HIGH level on RxS or TxS.

The test signal is a well-formed frame consisting of pulses of varying amplitude with edges at a distance of 0,5 BT, 1,0 BT and 1,5 BT, with an adjustable jitter between two consecutive edges, as shown in Figure 24.

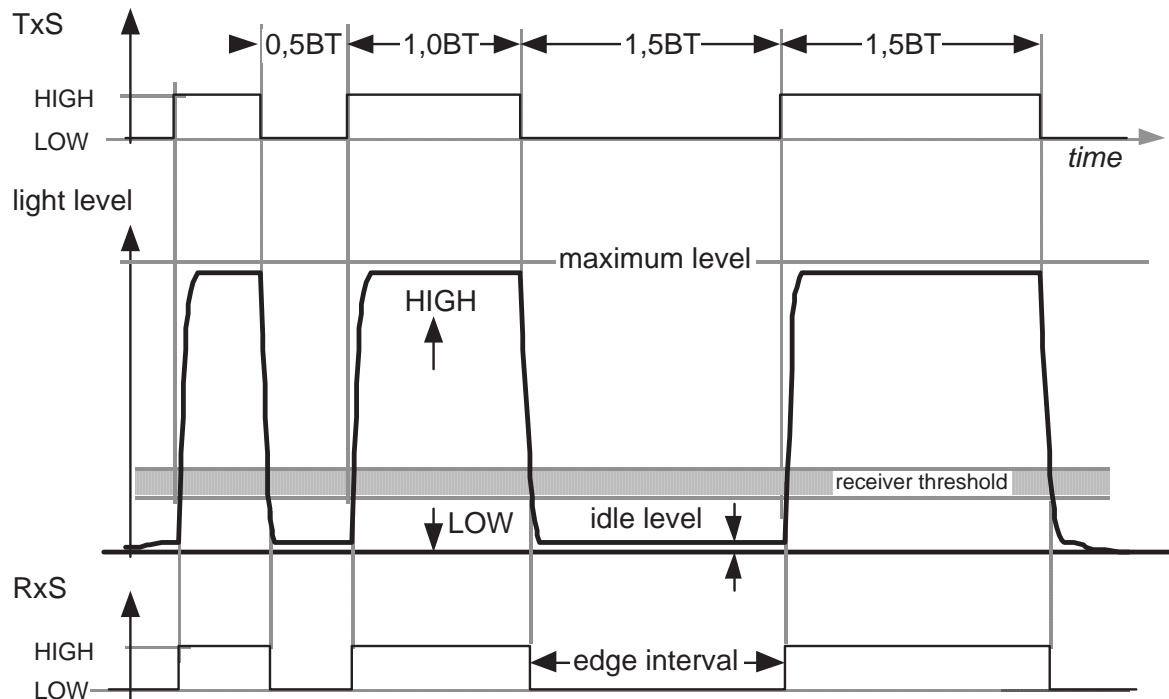


Figure 24 – Example of start of OGF frame

#### 4.6.5 OGF transmitter specifications

##### 4.6.5.1 OGF optical levels

The optical transmitter shall comply to the following specifications:

- the transmitter shall transmit two distinct optical levels;
- the peak emission wavelength shall be  $840,0 \text{ nm} \pm 30,0 \text{ nm}$ ;
- the spectral width shall be smaller than  $40,0 \text{ nm}$ ;
- the peak output power measured after  $1,0 \text{ m}$  of fibre, corresponding to the HIGH state shall be comprised between  $-7,5 \text{ dBm}$  and  $-4,5 \text{ dBm}$ ;
- the minimum power, corresponding to the LOW state, shall be less than  $-40,0 \text{ dBm}$ ;
- the rise time of the signals from LOW to HIGH (10 % to 90 %) shall be less than  $10,0 \text{ ns}$ ;
- the fall time of the signals from HIGH to LOW (90 % to 10 %) shall be less than  $10,0 \text{ ns}$ ;
- the difference between fall and rise time shall be less than  $10,0 \text{ ns}$ .

##### 4.6.5.2 OGF transmitter edge distortion

The edge distortion of the output signal, defined as the time difference between the idealised and the actual transitions, shall not exceed  $10,0 \text{ ns}$  over a whole frame.

##### 4.6.5.3 OGF end of frame

The transmitter shall switch off after the End Delimiter, as specified in 5.1.6.

#### 4.6.6 OGF receiver specifications

##### 4.6.6.1 OGF receiver

NOTE The test of these specifications assume that the electrical side of the receiver is accessible. If the optical receiver is type-tested according to this specification, this test may be omitted in testing conformance of the device.

The receiver shall comply to the following specifications:

- a) the input power corresponding to a LOW signal shall be  $< -40,0$  dBm ( $0,1 \mu\text{W}$ ) at  $840,0$  nm;
- b) the input power corresponding to a HIGH signal shall be  $> -24,0$  dBm ( $4,0 \mu\text{W}$ ) at  $840,0$  nm;
- c) the receiver shall operate correctly up to a light level of  $-10,0$  dBm ( $63,0 \mu\text{W}$ );
- d) the rise time of the output signal (LOW to HIGH) shall be less than  $10,0$  ns;
- e) the fall time of the output signal (HIGH to LOW) shall be less than  $10,0$  ns;
- f) the difference between fall and rise time shall be less than  $10,0$  ns.

#### 4.6.6.2 OGF receiver edge distortion

The receiver shall recognise a frame correctly even if the next edge appears  $0,125 \mu\text{s}$  too late or too early with respect to the expected edge position, as shown in Figure 25.

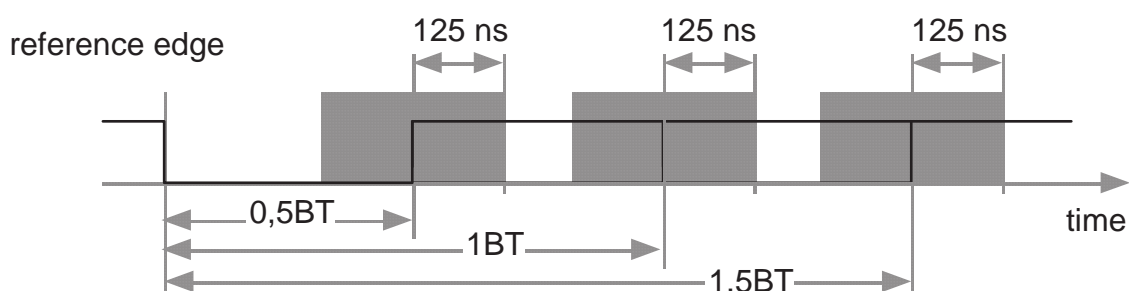


Figure 25 – Edge jitter

#### 4.6.7 OGF active star coupler

Bussing of optical fibres is preferably realised by an active star coupler, as illustrated in Figure 26.

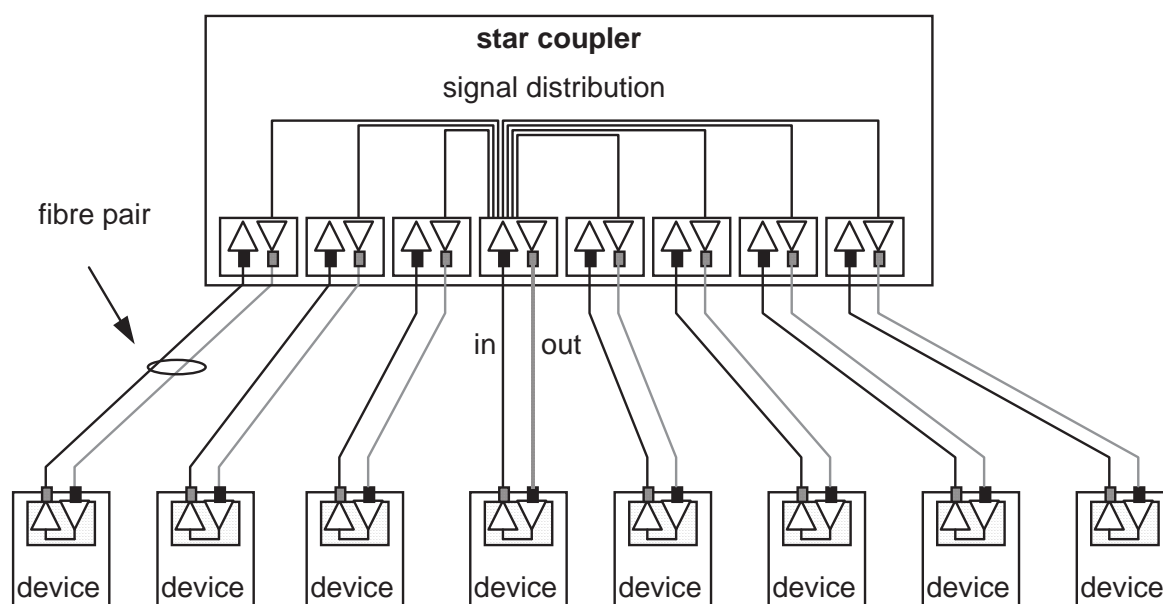


Figure 26 – Example of active star coupler

NOTE When the number of attached devices is small, bussing fibres by means of passive stars or taps is possible, provided the receiver's dynamic range permits it.

An active star coupler shall redistribute an incoming light signal over all outputs, except the output corresponding to the incoming signal.

The input and outputs of the active star shall conform to the light levels prescribed in 4.6.5 and 4.6.6.

The maximum delay between a change in an input signal and the corresponding change in the output signals shall be two repeater delays ( $2,0 T_{\text{repeat}}$ ).

#### 4.6.8 OGF double-line layout (option)

In case of a double-line optical segment, a star coupler shall be connected to either Line\_A or Line\_B, but not to both.

NOTE This rules out inserting devices on a star coupler used in a double-line segment.

EXAMPLE A redundant layout with optical fibres is shown in Figure 27.

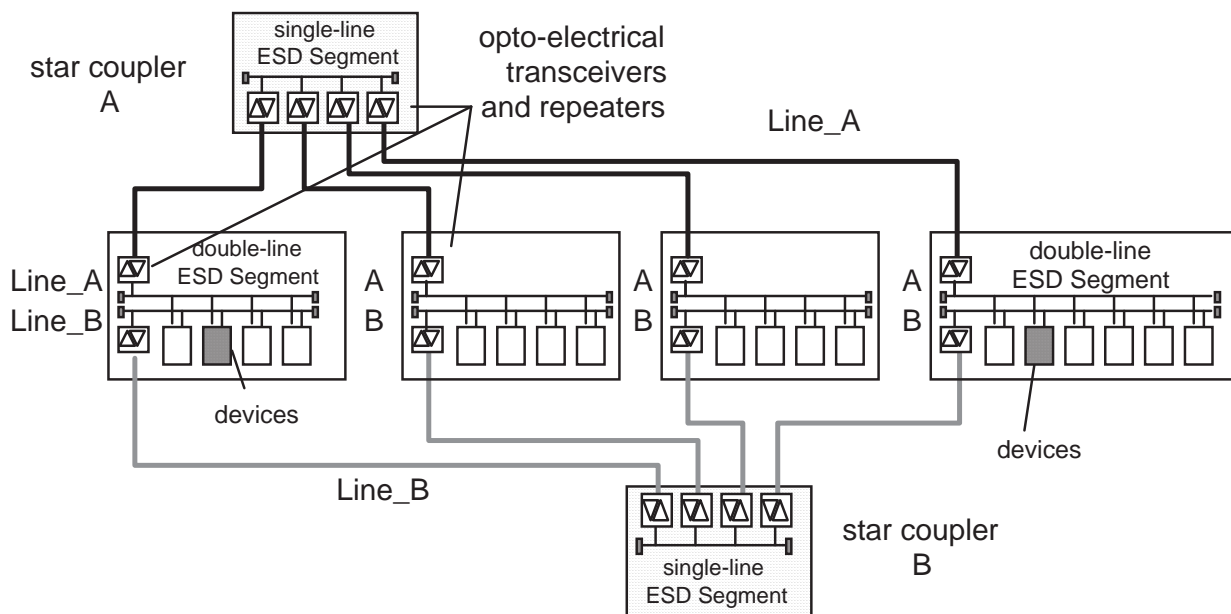


Figure 27 – Example of a duplicated star coupler

## 5 Medium-dependent signalling

This subclause specifies the signalling, which is common to electrical and optical media. Medium-dependent differences are specified.

### 5.1 Frame encoding and decoding

#### 5.1.1 Conventions

Frame encoding and decoding assume a binary medium with two distinct levels, called HIGH and LOW.

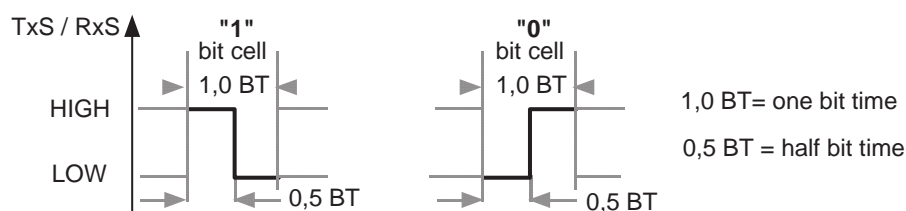
The transmitter imposes these levels on the line according to its transmitter signal TxS, and the receiver detects them as the received signal RxS. The relation between the TxS signal, the line levels, and the RxS signal is defined for each medium.

The following specifications consider the TxS and RxS signals as idealised line levels.

### 5.1.2 Bit encoding

Individual data bits, "1" and "0", shall be encoded in a bit cell as follows, as shown in Figure 28:

- A "1" shall be encoded by a HIGH level during the first half of a bit cell followed by a LOW level during the second half of a bit cell.
- A "0" shall be encoded by a LOW level during the first half of a bit cell followed by a HIGH level during the second half of a bit cell.



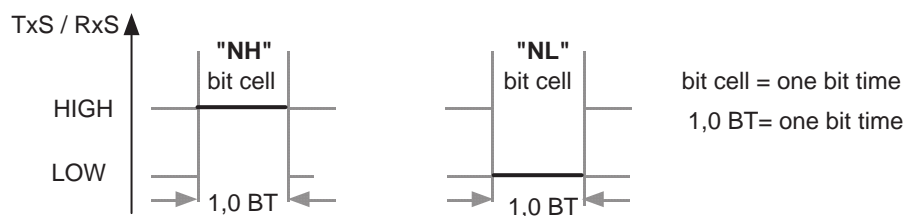
**Figure 28 – "0" and "1" data encoding**

NOTE This encoding is known as Manchester or Bi-phase-L encoding. There is no general agreement regarding the signal polarity.

### 5.1.3 Non-data symbols

Non\_Data symbols, NH and NL, shall be encoded in a bit cell as follows, as shown in Figure 29:

- A "NH" shall be encoded by a HIGH level during one bit cell.
- A "NL" shall be encoded by a LOW level during one bit cell.



**Figure 29 – Non\_Data symbols encoding**

### 5.1.4 Start Bit

The transmitter shall begin a frame with a Start Bit, which shall be sent as a "1" symbol.

NOTE 1 The timing of the start bit depends on the medium.

NOTE 2 The start bit may be received with a different shape or timing because of transitory effects and definition of the idle level.

### 5.1.5 Start Delimiter

Master Frames shall be headed by a Start Delimiter, called the Master Start Delimiter, consisting of the sequence:

{Start Bit, "NH", "NL", "0", "NH", "NL", "0", "0", "0"}, as shown in Figure 30.

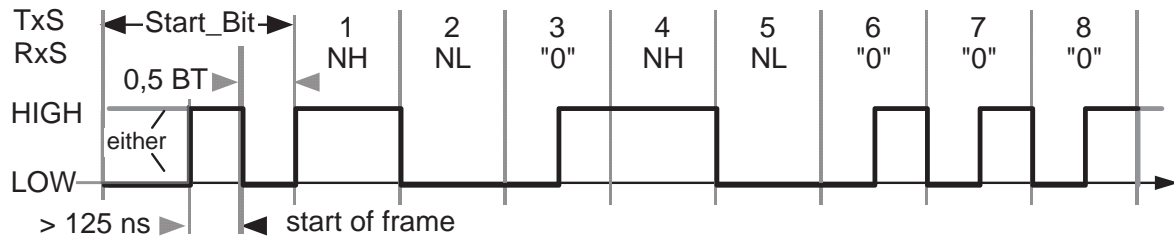


Figure 30 – Master Start Delimiter

Slave Frames shall be headed by a Start Delimiter, called the Slave Start Delimiter, consisting of the sequence:

{Start Bit, "1", "1", "1", "NL", "NH", "1", "NL", "NH"}, as shown in Figure 31.

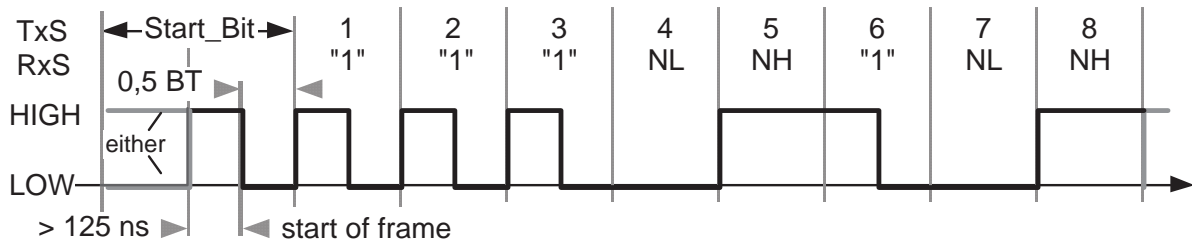


Figure 31 – Slave Start Delimiter

### 5.1.6 End Delimiter

The transmitter shall, after the last bit cell of a frame, send an End Delimiter, depending on the medium:

- with the ESD medium, append an "NL" symbol and cease transmission as specified in 4.4.8.4;
- with the EMD medium, append an "NL" symbol followed by an "NH" symbol and cease transmission as specified in 4.5.9.3;
- with the OGF medium, append an "NL" symbol and cease transmission.

NOTE The End Delimiter specification depends on the medium.

EXAMPLE An End Delimiter for the EMD medium is shown in Figure 32.

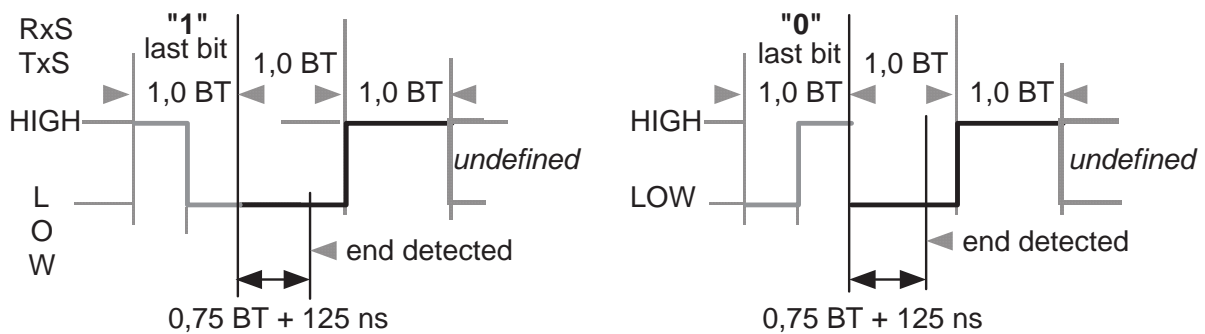


Figure 32 – Example of End Delimiter for EMD medium

The decoder shall consider that an End Delimiter is present when the RxS signal remains at the LOW level for longer than  $0,75 \text{ BT} + 0,125 \mu\text{s}$ .



A decoder shall ignore any pulses after the end of frame until it detects the next Start Delimiter.

### 5.1.7 Valid frame (definition)

A decoder shall recognise a valid frame as consisting of

- a Start Delimiter (5.1.5); followed by
- Frame Data consisting of "0" and "1" symbols (5.1.2); and followed by
- an End Delimiter (5.1.6).

EXAMPLE A typical valid frame with 16 bits of useful data is shown in Figure 33.

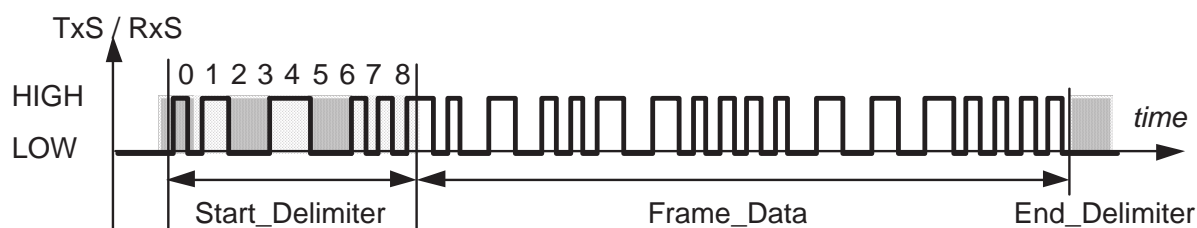


Figure 33 – Example of a valid frame (OGF medium)

The decoder may detect the correct polarity of signals by decoding the Start Delimiter, but it may not automatically invert the signal if the polarity is wrong.

To define timing:

- the start of the frame is defined as the HIGH to LOW transition of the Start Bit;
- the end of the frame is defined as the last HIGH to LOW or LOW to HIGH transition before the End Delimiter.

NOTE 1 The decoder may check in addition that the number of data bits is plausible.

NOTE 2 Since the decoder of the Trusted\_Line and of the Observed\_Line may check different criteria, they may interpret a frame differently.

### 5.1.8 Detection of line idle

The decoder shall have the ability to sense that the line is idle.

NOTE The decoder is expected to distinguish a limited amount of spurious noise (e.g. coming from reflections) from the absence of transmission.

### 5.1.9 Detection of collision

The decoder shall have the ability to distinguish between a line idle and a collision.

NOTE The decoder is not expected to distinguish a lasting spurious noise from a collision.

### 5.1.10 Receiver behaviour in case of error

If it detects a collision, the decoder shall ignore all frames on that line until the next Master Frame\_Delimiter is received.

### 5.1.11 Jabber halt

A device may provide means to detect that its transmitter is active during a time longer than a configurable value  $T_{jabber}$ , and means to disconnect the transmitter in such a case.

In this case, the time-out  $T_{jabber}$  shall be set to the duration of the longest frame which the device is capable of sending under normal operating conditions plus 10 %.

The jabber-halt shall enable transmission again after the next correct frame is received by the device.

## 5.2 Line redundancy (option)

NOTE This subclause specifies the handling of redundant lines in case this option is used.

### 5.2.1 Principle

The line redundancy principle assumes that a device transmits simultaneously the same data over both Line\_A and Line\_B, and that a device accepts the data from one line, called the Trusted\_Line, while it monitors the other line, called the Observed\_Line.

Each device chooses its Trusted\_Line and Observed\_Line based on the signals generated by its physical layer, or upon request of its Link Layer, independently from other devices.

The skew  $t_{skew}$  is defined as the timing difference between the signals over both lines, measured at the transmitter, at the receiver, or anywhere on the line as shown in Figure 34.

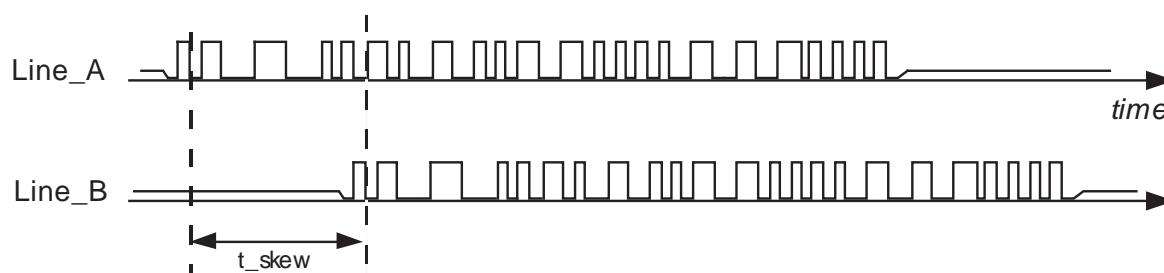


Figure 34 – Signal skew

### 5.2.2 Redundant transmission

#### 5.2.2.1 Operation

A device with double-line attachment shall transmit a frame over both redundant lines Line\_A and Line\_B, so that the skew of the two output signals does not exceed  $T_{skew\_t} = 0,100 \mu s$ .

#### 5.2.2.2 Propagation skew

The layout of the bus shall ensure that the skew between the signals received over Line\_A or Line\_B does not exceed  $T_{skew} = 8,0 \mu s$  at any place on the bus.

### 5.2.3 Redundant reception

A device shall receive data only from the Trusted\_Line and shall monitor continuously the Observed\_Line.

A receiver shall operate correctly and report no error if it receives valid frames on both lines with a skew less than  $T_{skew\_r} = 8,0 \mu s$ .

### 5.2.4 Switchover

A device shall exchange the Trusted\_Line and the Observed\_Line under the following conditions:

- a) if no valid frame has been received from the Trusted\_Line for  $T_{\text{switchover}} = 1,4 \text{ ms}$  since the end of the last valid Master Frame or since the last switchover;
- b) if the decoder of the Observed\_Line detects a valid frame and the decoder of the Trusted\_Line detects no valid frame within  $T_{\text{skew}_r}$ , but no collision is expected on that frame (see 6.3.2), and after the Observed\_Line has been idle for longer than 2,0 BT, and provided Redundant\_Line\_Disturbed is not set (5.2.5);
- c) if a valid frame has been received over the Trusted\_Line, but the frame length or the Check Sequence is incorrect, provided Redundant\_Line\_Disturbed is not set (option);
- d) if a device responds to a Device\_Status\_Request with a Device\_Status\_Response, provided Redundant\_Line\_Disturbed is not set, this condition being mandatory for Class 1 devices and a configuration option for the other Classes;
- e) if a device receives a request of its Link Layer, this condition being a configuration option not applicable to Class 1 devices.

NOTE 1 The time-out  $T_{\text{switchover}}$  (1,4 ms) is related to the maximum distance between master frames  $T_{\text{alive}}$  (1,3 ms).

NOTE 2 Once a switchover takes place, a switchover to the former line may only take place once the Redundant\_Line\_Disturbed bit has been reset by a Device\_Status\_Response or by the Link Layer of that device.

NOTE 3 If a continuous jammer sends on the Trusted\_Line, switchover may have to wait 1,4 ms until forced switchover.

### 5.2.5 Redundancy status report

The MAU shall report the redundancy status to the Link Supervision Interface through two status bits, called LAT and RLD:

- LAT (Line\_A Trusted) shall be set while Line\_A is the Trusted\_Line;
- RLD (Redundant\_Line\_Disturbed) shall be:
  - set if a switchover took place or if the decoder of the Trusted\_Line detects a valid frame and the decoder of the Observed\_Line detects no valid frame within a skew less than  $T_{\text{skew}_r}$ , but no collision is expected on that frame (see 6.3.2);
  - reset either after sending the Device\_Status\_Response or by the Link Layer (at the application's choice).

In a single-line attachment, a device shall consider the unused line as permanently disturbed, RLD is always set.

There shall be an error counter for each line, which shall be incremented by a Line\_Disturbed signal defined as follows:

The Line\_Disturbed signal shall be

- set:
  - if a collision happens on the Trusted\_Line, but no collision is expected (see 6.3.2);
  - if a valid frame is received on the Observed\_Line and no valid frame appears on the Trusted\_Line within  $T_{\text{skew}_r}$ , but no collision is expected (see 6.3.2), and
- reset:
  - by reception of a valid frame on the Trusted\_Line.

NOTE This definition of the error counter prevents the counter from being rapidly incremented if there is a permanent disturbance on a line, for example a line break or if the redundant line is not used.

### 5.3 Repeater

Signal regeneration takes place in repeaters (optical or galvanically isolated coupler). The following specifications aim at allowing at least up to four repeaters in series.

NOTE The time-out  $T_{jabber\_all}$  of a repeater is larger than  $T_{jabber}$  of a device by 5% to ensure that a device is able to shut down itself in case of failure before its failure affects the segment.

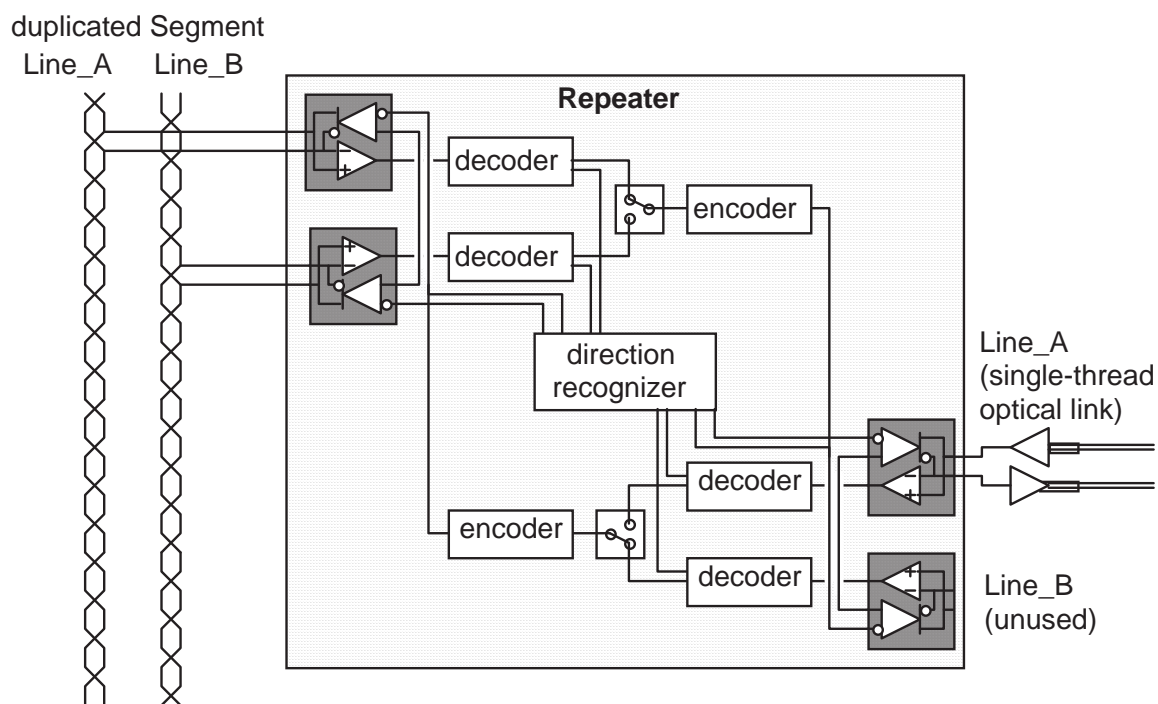


Figure 36 – Example of repeater connecting a double-line to a single line segment

## 6 Frames and telegrams

### 6.1 Frame format

#### 6.1.1 Master Frame format

A Master Frame shall, as shown in Figure 37:

- begin with the Master Start Delimiter (5.1.5);
- followed by 16 bits of Frame Data (7.2.1);
- followed by the 8-bit Check Sequence (6.1.3).

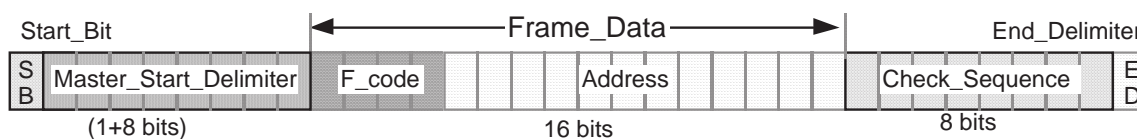
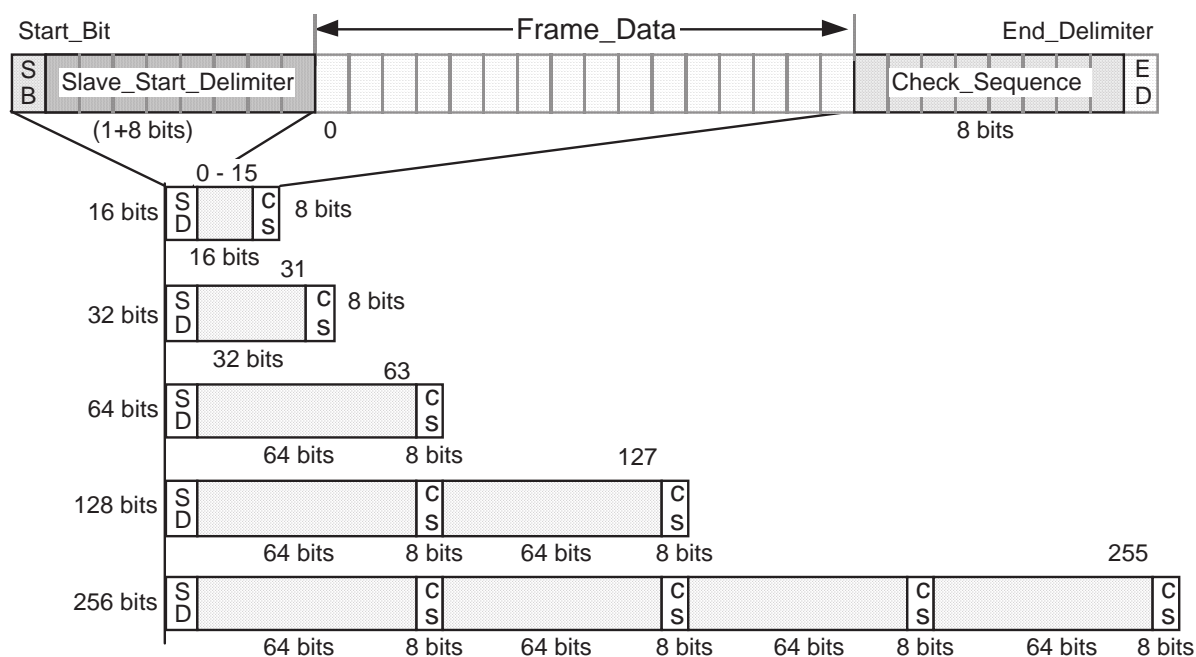


Figure 37 – Master Frame Format

#### 6.1.2 Slave Frame format

A Slave Frame shall, as shown in Figure 38:

- begin with the Slave Start Delimiter (5.1.5);
- followed by 16, 32, 64, 128 or 256 bits of Frame Data (7.3.1);
- include an 8-bit Check Sequence after each word of 64 data bits or be appended to the frame when the Frame Data is 16 or 32 data bits (6.1.3).



**Figure 38 – Slave Frames**

NOTE 1 Start Bit (SB) and End Delimiter (ED) are implicit in the following specifications and omitted in the Figures.

NOTE 2 Longer frames could be built on the same pattern, i.e. one Check Sequence after each word of 64 bits.

### 6.1.3 Check Sequence

Frames shall be protected by one or more 8-bit Check Sequences.

The contents of the data shall be treated as a codeword of 64 bits (16 and 32 bit for the smaller frames), excluding Start Delimiter and End Delimiter.

This codeword and the following Check Sequence shall be transmitted most significant bit first.

The Check Sequence shall be computed as a cyclic redundancy check (CRC) on the 16, 32 or 64 bits of data it protects.

The algorithm for the Check Sequence shall correspond to IEC 60870-5-1 format class FT2, which is repeated here.

The Check Sequence shall be computed according to the following generating polynomial:

$$G(x) = x^7 + x^6 + x^5 + x^2 + 1$$

The resulting 7-bit remainder shall be extended by an even parity bit.

All eight bits shall be transmitted inverted.

EXAMPLE

The 16-bit information  
multiplied by  $x^7$   
divided by  $x^7 + x^6 + x^5 + x^2 + 1$ :  
yields a remainder of:  
which is extended by an even parity bit:  
and transmitted inverted as the check  
octet:

0111 1110 1100 0011  
0111 1110 1100 0011 000 0000  
1110 0101  
001 0001  
001 00010  
CS = 1101 1101

6.2 Telegram timing

6.2.1 Conventions

A telegram is defined as a Master Frame and of the Slave Frame sent in response to it.

EXAMPLE The timing of a telegram anywhere on the bus is shown in Figure 39.

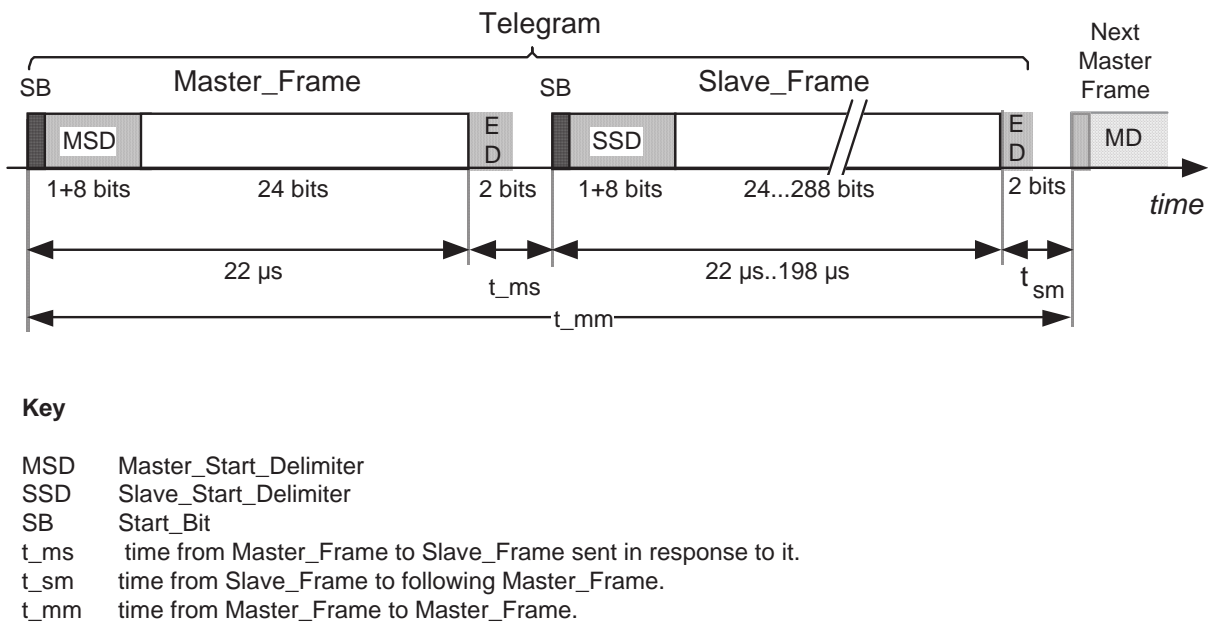


Figure 39 – Telegram timing

The space between frames is counted from the start of frame and end of frame points defined for each medium.

In a double-line segment, the interframe spacing is counted from the end of the first frame on any line, whichever comes later, to the start of the second frame, whichever comes earlier and as long as the RLD bit is not set.

6.2.2 Reply delay (definition)

The reply delay  $T_{reply}$  is, for a given bus, the maximum delay which may appear between the end of a Master Frame and the beginning of the Slave Frame sent in response to it, measured at the master.

The reply delay is the sum of propagation, decoding and access delays.

$T_{reply}$  is a configuration parameter which tells the master how long it shall wait before sending the next Master Frame if it receives no Slave Frame or if it expects a collision (see 6.3.2).

EXAMPLE The reply delay in a bus with 2 repeaters is shown in Figure 40 .

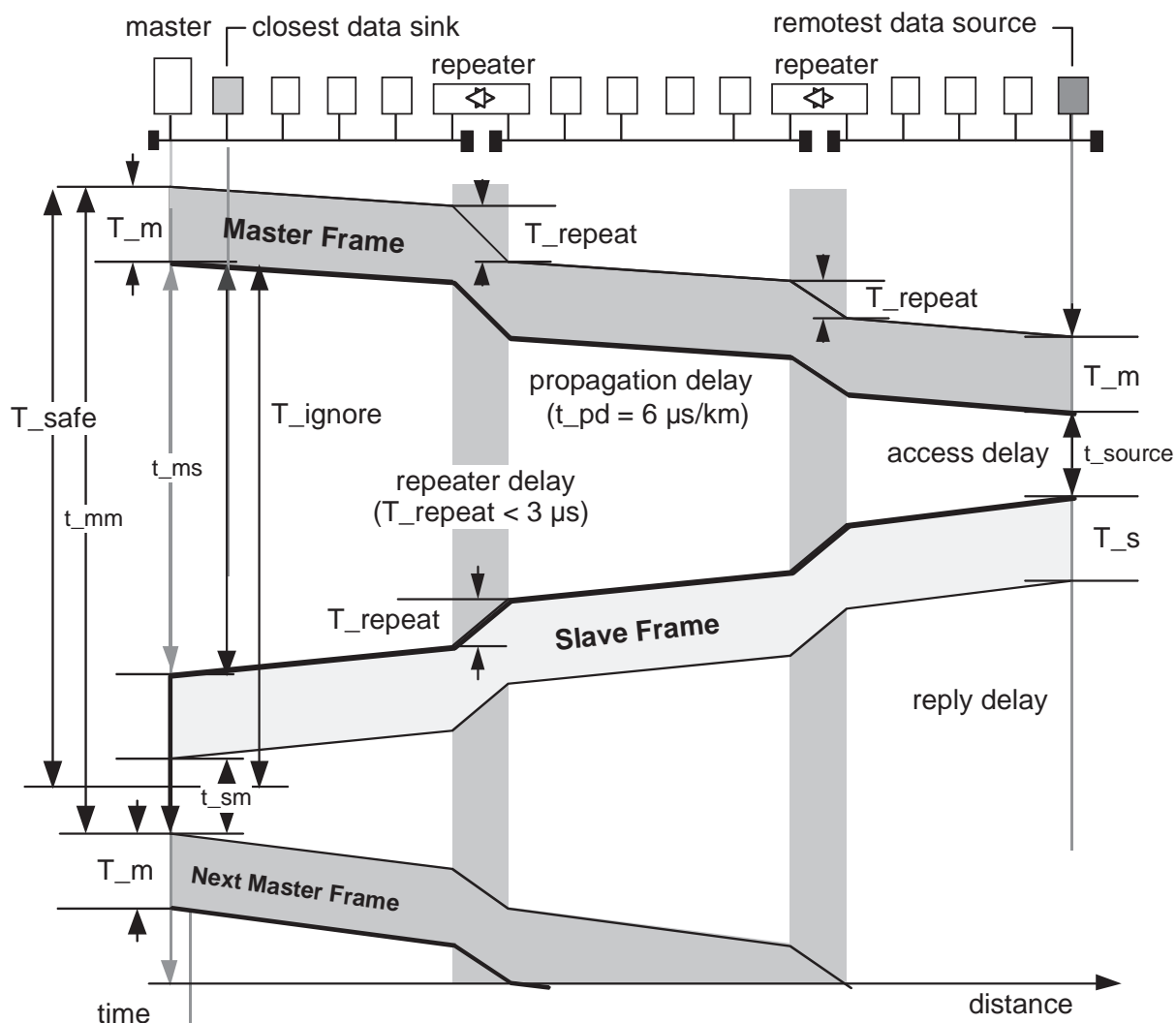


Figure 40 – Example of Reply delay

$T_{reply}$  for a given bus is calculated as the sum of twice the propagation delays (line and repeaters), plus the delay at the slave in the worst-case:

$$T_{reply} \text{ } [\mu s] = 2 \times (6,0 \times L + T_{repeat\_max} \times N_{rep}) + T_{source\_max}$$

where

6,0 μs/km is the worst-case delay per km of a line (0,5 × speed of light);

L is the electrical length of the line in km (which is larger than the physical distance);

$T_{repeat\_max}$  is the maximum delay that a repeater may introduce (3,0 μs in one direction);

$N_{rep}$  is the number of repeaters;

$T_{source\_max}$  (6,0 μs) is the maximum delay at the source (6.2.3).

EXAMPLES



$T_{reply} = 6,24 \mu s$  for a 20,0 m bus without repeater in between.

$T_{reply} = 44,0 \mu s$  for a 2,0 km bus with two repeaters.

The default and minimum value of  $T_{reply}$  shall be  $T_{reply\_def} = 42,7 \mu s$ .

NOTE In the "Extended Reply Delay" configuration option,  $T_{reply}$  exceeds the value  $T_{reply\_def}$ .

### 6.2.3 Frame spacing at the source

After the end of a Master Frame, the addressed slave source device shall reply with a Slave Frame as shown in Figure 41:

- at the earliest after  $T_{source\_min} = 2,0 \mu s$ ; and
- at the latest before  $T_{source\_max} = 6,0 \mu s$ .



Figure 41 – Frame spacing at the source side

NOTE The recommended value of  $T_{source}$  is  $4,0 \mu s$ . Devices with  $T_{source} > 3,0 \mu s$  always comply with 4.3.2.

### 6.2.4 Frame spacing at the destination

#### 6.2.4.1 Default frame spacing

A destination device shall ignore all Slave Frames which start later than  $T_{ignore}$  after the end of the last Master Frame it received, as shown in Figure 42.

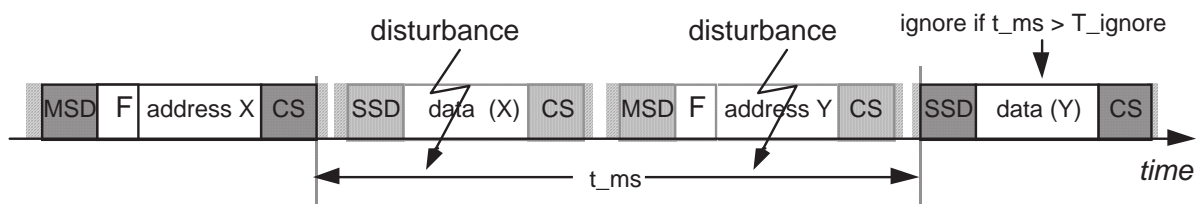


Figure 42 – Frame spacing at the destination(s)

The time-out  $T_{ignore}$  shall have a default value of  $T_{reply\_def} = 42,7 \mu s$ .

NOTE Process Data Slave Frames do not contain an address, but are identified by the preceding Master Frame. The loss of a Slave Frame (data(X)) and of the following Master Frame (address Y) could lead a device to accept the second Slave Frame (data(Y)) as belonging to the first Master Frame (address X). Therefore, a device suspects a double loss if it measures a delay  $t_{ms}$  longer than the time to transmit the smallest Slave Frame and the following Master Frame, which is  $49 \mu s$  ( $22 \mu s + 2 \mu s + 22 \mu s + 3 \mu s$ ).  $T_{ignore}$  is therefore set by default to a value  $42,7 \mu s$ , which is smaller than  $49 \mu s$ . On the other hand, a device should be able to accept a Slave Frame which is delayed by  $T_{reply}$ , to allow reception of worst-case delayed Slave Frames. When  $T_{reply}$  is smaller than  $42,7 \mu s$  this condition is always fulfilled, but it is not for Extended Reply Delay mode, which requires special precautions (6.2.4.2).

#### 6.2.4.2 Extended Reply Delay frame spacing

In long busses where the value of  $T_{reply}$  exceeds the default value of  $T_{reply\_def}$ ,  $T_{ignore}$  may be increased to a value larger than  $T_{reply\_def}$ , provided the following precautions are taken:

- devices with a time-out  $T_{ignore}$  set to a value higher than  $T_{reply\_def}$  shall indicate this by setting the ERD (Extended Reply Delay) bit in their Device\_Status;
- the configuration variable  $T_{safe}$  shall be set equal to the highest value of  $T_{ignore}$  used by any device attached to the bus;
- the configuration variable  $T_{safe}$  shall be set equal to  $T_{reply\_def}$  if all devices in a given bus are using the value  $T_{ignore} = T_{reply\_def}$ .

NOTE In Extended Reply Delay applications, the master is responsible to space the Master Frames so that the minimum time to transmit a Slave Frame and the following Master Frame is greater than  $T_{safe}$ .

In effect, if any device uses a value of  $T_{ignore}$  which is larger than  $T_{safe}$ , there exists a potential integrity breach on that device. To detect this situation, the bus administrator should check if any device is operating in Extended Reply Delay mode by reading its Device\_Status.

#### 6.2.5 Frame spacing at the master

A master shall send the next Master Frame not later than  $T_{alive} = 1,3$  ms after the start of the previous Master Frame, as shown in Figure 43.

A master may send the next master frame:

- at the earliest  $T_{safe}$  after the start of its previous master frame in all cases (6.2.4.2);
- at the earliest a time  $T_{reply} + T_m$  after the start of its previous master frame if it expects a collision or a silence in response to its sent master frame;
- at the earliest  $T_{sm\_min} = 3,0$   $\mu$ s if it expects neither collision nor silence in response to its previous master frame.

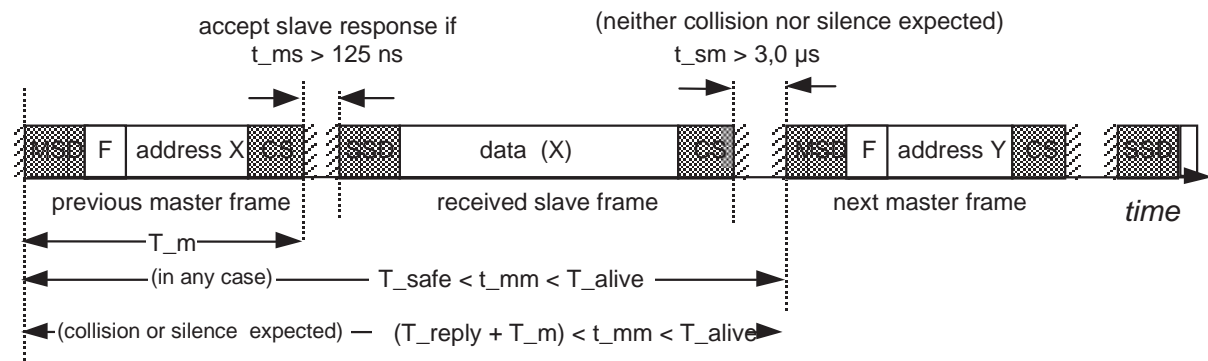


Figure 43 – Frame spacing at the master side

NOTE 1 The  $T_{alive}$  delay is used to supervise the master's activity for redundant masters.

NOTE 2 The requirement  $t_{mm} > T_{safe}$  is always met when all devices on the bus use the default value  $T_{ignore} = 42,7$   $\mu$ s, since the minimum time in which a telegram can be transmitted is equal to 49  $\mu$ s ( $= 22,0$   $\mu$ s + 2,0  $\mu$ s + 22,0  $\mu$ s + 3,0  $\mu$ s), which is longer than 42,7  $\mu$ s. If any device operates in Extended Reply Delay mode,  $T_{safe}$  will be set to the highest value of  $T_{ignore}$ .

### 6.3 Detection of correct frames, collision and silence by the master

#### 6.3.1 Correct frame (definition)

A correct Slave Frame is defined as a valid frame (5.1.7) which is received within the limits set by 4.2.4, headed by a Slave Start Delimiter, which has a size corresponding to that announced in the Master Frame and with correct CRCs inserted.

NOTE Only correct frames may be forwarded to the application.

#### 6.3.2 Detection of collision by the master

The master shall detect a collision if the decoder of the trusted line reports a collision or if the received slave frame is not correct.

The master shall detect no collision if it receives a correct slave frame, even if a second valid frame comes within  $T_{reply}$  after the first one due to long propagation delays. The master shall then consider the first received correct Slave Frame as a single answer.

A collision is said expected on the frames of type General\_Event\_Response, Group\_Event\_Response and Individual\_Event\_Response.

NOTE 1 A collision occurs when several transmitters are simultaneously active. This occurs normally only during event arbitration. The collision is monitored only by the master.

NOTE 2 In a long bus, it is possible that the master receives a correct frame although several slaves answer: a close transmitter can override a remote one, or a second frame can arrive after the end of the first frame because the reply delay  $T_{reply}$  could exceed the minimum Slave Frame duration (= 22,0  $\mu$ s).

#### 6.3.3 Detection of silence by the master

The master shall detect a silence if the decoder reports neither a valid frame nor a collision within  $T_{reply}$  after the end of the previous Master Frame.

A silence is said expected on all frames except Process Data\_Responses.

NOTE A disturbance can cause the master to interpret silence as a collision. This can lead the event search on the false path. The decoder helps to distinguish between disturbance and collision.

## 7 Link Layer Control

### 7.1 Addressing

#### 7.1.1 Device Address

A device shall be identified on a bus by its Device Address.

The Device Address shall have a size of 12 bits.

Device Address 0 shall not be used.

NOTE 1 Devices supporting the Message Data capability should have a Device Address smaller than 256.

NOTE 2 The broadcast device address is indicated by the mode field, not by a dedicated address.

#### 7.1.2 Logical\_Address

A Process Data source shall be identified by its Logical\_Address.

The Logical\_Address shall have a size of 12 bits.

Logical\_Address 0 shall not be used.

7.1.3 Group\_Address

For the purpose of Event\_Arbitration, devices may be addressed by their Group\_Address, as specified in 8.3.1.

7.2 Master Frames contents

7.2.1 Master Frame format

The Master Frame shall carry a 16-bit data word.

The first transmitted bit of that word shall be its most significant bit, identified as bit 0 as shown in Figure 44.

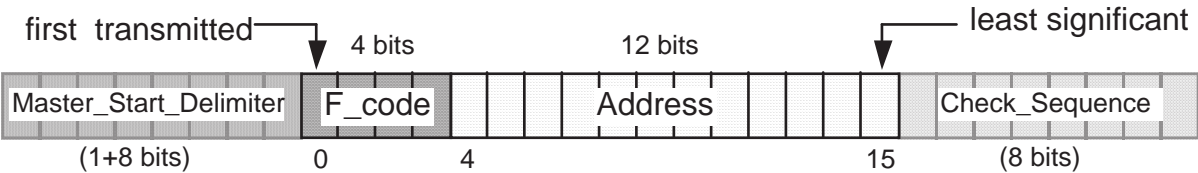


Figure 44 – Master Frame contents

The four most significant bits of the 16-bit word shall be the F\_code.

The least significant 12 bits shall represent an address or parameters as specified by the F\_code.

7.2.2 F\_code encoding

The 4-bit F\_code qualifies the following 12-bit and indicates the size of the expected Slave Frame.

The F\_code shall be encoded according to Table 8.

**Table 8 – Master Frame types and F\_code**

Master Frame			Slave Frame			
F_code	Address	Request	Source	Size (bits)	Response	Destination
0	logical	Process Data	single device subscribed as source	16	Process Data (application-dependent)	all devices subscribed as sink
1				32		
2				64		
3				128		
4				256		
5		Reserved		-		
6		Reserved		-		
7		Reserved		-		
8	device	Mastership_Transfer	proposed master	16	Mastership_Transfer	master
9	all devices	General_Event (parameters)	one or more devices	16	Event_Identifier	master
10	device	Reserved	-	-		
11	device	Reserved	-	-		
12	device	Message Data	single device	256	Message Data	selected device(s)
13	device group	Group_Event	one or more devices	16	Event_Identifier	master
14	device	Single_Event	single device	16	Event_Identifier	master
15	device	Device_Status	single device	16	Device_Status	master or monitor device

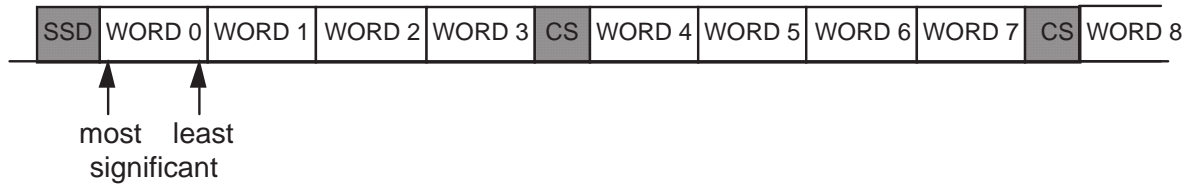
### 7.3 Slave Frame contents

#### 7.3.1 Slave Frame format

A Slave Frame shall transmit 1, 2, 4, 8 or 16 words of 16 bits each.

Bits within a word shall be transmitted most significant bit first (treating the word as an unsigned, 16 bit integer, the first transmitted bit having offset 0).

Words shall be numbered starting with 0, as shown in Figure 45.



**Figure 45 – Word ordering in a Slave Frame**

Frames shall be extended by "0s" if the full frame size is not used for link data.

NOTE The contents of a Slave Frame are not specified except for the Message Data header and for the Supervisory Data frames.

### 7.3.2 Size error

A destination shall ignore a Slave Frame if its length does not match the length indicated by the F\_code in the Master Frame, even if its Check Sequence is otherwise correct.

## 7.4 Telegram types

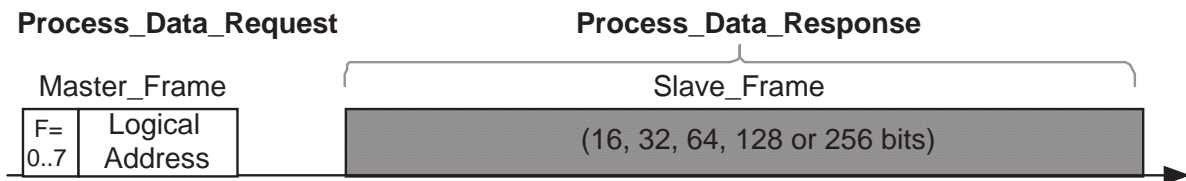
### 7.4.1 Process Data telegram

#### 7.4.1.1 Process Data format

A master shall request the transmission of Process Data by sending a Process Data\_Request, which is a Master Frame with one of the F\_codes: 0..4.

NOTE F\_codes 5..7 are reserved for future extensions of Process Data.

The slave which sources the Process Data corresponding to the identifier in the Process Data\_Request shall respond with a Process Data\_Response, as shown in Figure 46.



**Figure 46 – Process Data telegram**

#### 7.4.1.2 Process Data identifier

The Process Data shall be identified by both the 12-bit Logical\_Address and the F\_code in the Master Frame, which indicates the size of the Process Data (7.2.2).

For each Logical\_Address, the size of the corresponding Process Data shall be configured in the source and all sink devices before regular operation begins.

#### 7.4.1.3 Buffers

A device shall hold a buffer for each Logical\_Address it is subscribed to, as source or as sink.

#### 7.4.1.4 Source devices

A device sourcing Process Data, the identifier of which appears in the Process Data\_Request, shall read the contents of its source buffer and send it in one indivisible operation in the Slave Frame.

A device shall not respond if the F\_code of the Process Data\_Request indicates a frame length different from the configured length for that Logical\_Address.

A source buffer shall provide means to be overwritten with "0" to invalidate its data.

#### 7.4.1.5 Sink devices

A device sinking Process Data, the identifier of which appears in the Process Data\_Request, shall store that frame in the corresponding buffer, overwriting the previous contents of that buffer in one indivisible operation with the contents of the frame received over the bus.

A device sinking a Process Data frame, the identifier of which appears in the Process Data\_Request, shall not accept a Slave Frame if the F\_code indicates a frame length different from the configured length for that Logical\_Address.

A sink buffer shall provide means to indicate how long ago it was updated.

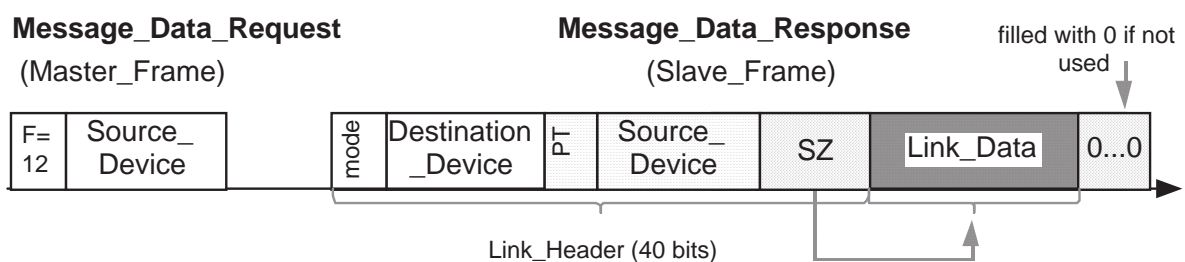
### 7.4.2 Message Data

#### 7.4.2.1 Message Data format

The master shall request the transmission of Message Data by sending a Message Data\_Request, which is a Master Frame with F\_code = 12.

The device addressed in the Master Frame shall respond with a Message Data\_Response, with the following format, starting with the first transmitted bit, as shown in Figure 47:

- a) the first 4 bits define the communication mode:
  - Mode '0001'B (+1) specifies the single-cast addressing mode,
  - Mode '1111'B (-1) specifies the broadcast addressing mode,
  - all other modes are reserved for future use;
- b) the next 12 bits define the address of the Destination\_Device.  
In case a broadcast mode is used, this field may be used by the application;
- c) the next 4 bits define the Protocol\_Type. The Protocol\_Type for the Real-Time Protocols is defined in 10.3.4;
- d) the next 12 bits indicate the address of the Source\_Device;
- e) the next 8 bits (SZ) express the number of significant octets in the following Link Data;
- f) the following bits carry the Link Data;
- g) the remaining octets of the frame are filled with "0" if the Link Data are less than 216 bits in size.



**Figure 47 – Message Data telegram**

NOTE 1 The Link Data contents are defined in Clause 6 of IEC61375-2-1 "Real-Time Protocols".

NOTE 2 The single-cast Mode and the broadcast Mode are the 2's complement of each other. For instance, if a future Mode expresses single-cast by 3 ('0011'B), the corresponding broadcast Mode will be -3 ('1110'B).

7.4.2.2 Queues

Each device capable of Message Data communication shall hold a pair of queues, one for sending (Send\_Queue) and one for receiving (Receive\_Queue) Message Data.

NOTE The queues are not prioritised.

7.4.2.3 Message Data protocol

When receiving a Message Data\_Request, a device specified by the source address in the Message Data\_Request shall:

- send a Message Data\_Response from its Send\_Queue and remove it from the Send\_Queue, or
- not respond at all if its Send\_Queue is void.

When receiving a Message Data\_Response, a device which recognises its own Device Address as Destination\_Address or the broadcast mode shall:

- append the Message Data to its Receive\_Queue, if this is possible, or
- ignore the Message Data, if this is not possible.

NOTE Repetition of lost frames is responsibility of the Transport\_Layer.

7.4.3 Supervisory Data telegrams

7.4.3.1 Supervisory Data format

A master shall request Supervisory Data from a device or shall send Supervisory Data to a device with a Supervisory\_Data\_Request, which is a Master Frame using one of the following F\_codes (the appended 12 bits are specified in the corresponding subclauses):

F_code =	8	Mastership_Transfer_Request	(9.3.2);
F_code =	9	General_Event_Request	(8.3.4.1);
F_code =	13	Group_Event_Request	(8.3.4.2);
F_code =	14	Single_Event_Request	(8.3.4.3);
F_code =	15	Device_Status_Request	(8.4.1.1).

The Supervisory\_Data\_Response shall carry 16 bits of Frame Data, as shown in Figure 48. These bits are specified in the Subclause defining the response frames corresponding to the F\_code:

F_code =	8	Mastership_Transfer_Response	(9.3.2);
F_code =	9	Event_Identifier_Response	(8.3.4.4);
F_code =	13	Event_Identifier_Response	(8.3.4.4);
F_code =	14	Event_Identifier_Response	(8.3.4.4);
F_code =	15	Device_Status_Response	(8.4.1.2).

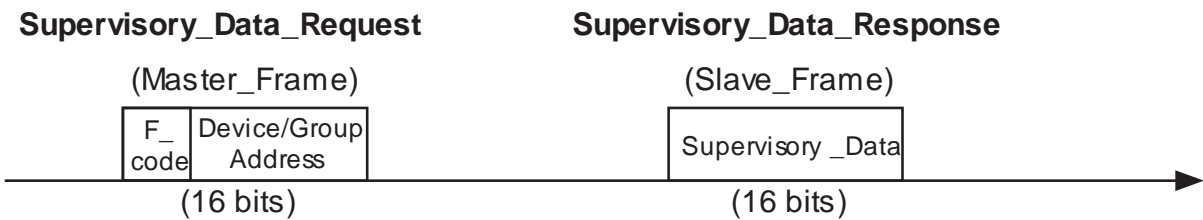


Figure 48 – Supervisory Data telegram



### 7.4.3.2 Supervisory Data protocol

Supervisory Data telegrams shall not be repeated immediately if no Supervisory\_Data\_Response comes after a Supervisory\_Data\_Request.

## 8 Medium allocation

### 8.1 Organisation

#### 8.1.1 Turn

The medium access shall be controlled by one of the bus administrators, the master, which remains in control of the bus for the duration of a turn.

At the end of a turn, the master shall try to pass mastership to another bus administrator or shall obtain it again if it finds no available bus administrator.

NOTE The beginning of a turn may be observed by the attached devices, which can monitor which bus administrator is currently master, for synchronisation purposes.

#### 8.1.2 Basic Period

The master shall divide its turn into fixed time slots, called "Basic Periods", all of the same duration  $T_{bp}$ .

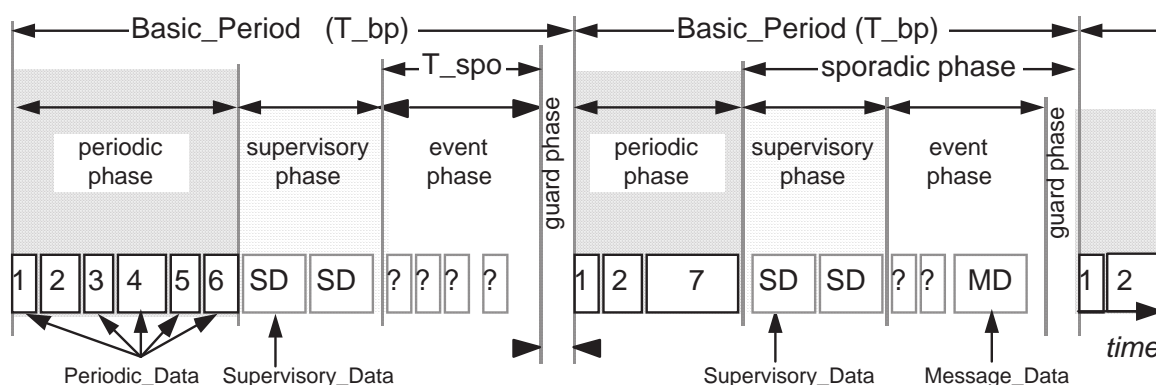
The Basic Period to be used for a turn shall take a value:

$$1,0 \text{ ms} \leq T_{bp} \leq 2,5 \text{ ms}.$$

NOTE 1 For synchronism purposes, any fraction of the period of the controlled process, given dividing this period by a power of two and included in the range between 1 ms and 2 ms can be used, for example: 1,042 ms ( $16 \times 60 \text{ Hz}$ ), 1,250 ms ( $16 \times 50 \text{ Hz}$ ) and 1,875 ms ( $32 \times 16,66 \text{ Hz}$ ).

A Basic Period shall be divided into four phases, as shown in Figure 49:

- a Periodic Phase;
- a Supervisory Phase;
- an Event Phase;
- a Guard Phase.



**Figure 49 – Basic Periods**

NOTE 2 Event Phase, Supervisory Phase and Guard Phase form the Sporadic Phase.

The value of  $T_{spo}$  may vary from Basic Period to Basic Period.

The default value of  $T_{\text{spo}}$  shall be 350,0  $\mu\text{s}$ .

If it becomes necessary to use a lower value of  $T_{\text{spo}}$  for a limited number of Basic Periods,  $T_{\text{spo}}$  shall be larger than the time to transmit the shortest telegram,  $T_{\text{mm\_min}} = 49,0 \mu\text{s} + T_{\text{reply}}$ .

NOTE 3 The maximum value of  $T_{\text{spo}}$  is equal to the duration of the Basic Period.

#### **8.1.2.1 Periodic Phase**

During the Periodic Phase, the master shall perform the Periodic Polling specified in 8.2

#### **8.1.2.2 Event Phase**

During the Event Phase, the master shall perform the Event Polling specified in 8.3.

#### **8.1.2.3 Supervisory Phase**

The master shall use the Supervisory Phase for the Devices\_Scan (8.4) and for Mastership Transfer (9).

NOTE Supervisory Data may be transmitted as Periodic\_Data, as Sporadic\_Data or by initiative of the master.

#### **8.1.2.4 Guard Phase**

The master shall reserve a Guard Phase as a buffer after the sporadic transmissions to allow an exact start of the next Periodic Phase.

The master shall not send a Master Frame if the transmission of the longest expected response would take longer than the time remaining until the next Basic Period is scheduled.

#### **8.1.3 Padding**

Each Basic Period shall be constructed in such a way that the master has an opportunity to issue a Master Frame (of any type) at least once every  $T_{\text{alive}}$  (4.2.5) = 1,3 ms to allow each device to monitor the line (9.2.2).

### **8.2 Periodic Polling**

#### **8.2.1 Periodic List**

The Periodic List shall define the Master Frames sent out periodically and the time left for the Sporadic Phase for each Basic Period of a turn.

During Periodic Polling, the master shall send a predefined sequence of Master Frames according to its Periodic List.

Modifications made to the Periodic List shall not come into effect during the same turn.

NOTE The Periodic List contains the Periodic\_Data, i.e. mainly Process Data\_Requests, but also Supervisory\_Data\_Requests or Message Data\_Requests for special purposes.

#### **8.2.2 Individual Period**

The master shall poll each Periodic\_Data with its Individual Period  $T_{\text{ip}}$ .

An Individual Period shall be equal to the Basic Period multiplied by a power of 2, but shall not exceed 1024  $T_{\text{bp}}$ , or in general:

$T_{ip}(i) = T_{bp} \times 2^n$ , with  $n = (0..10)$ .

The longest Individual Period in the Periodic List, called the Macro\_Period, shall not exceed 1024,0 ms.

NOTE 1 The Individual Period is defined by the application.

NOTE 2 If the shortest Individual Period exceeds the Basic Period, the Periodic Phase of some Basic Periods could be void.

### 8.2.3 Periodic Phase construction

The Periodic List is configured by the application. The following subclauses indicate how the Periodic List is to be constructed.

#### 8.2.3.1 Conventions for constructing the Periodic List

The following conventions shall apply to the format of the Periodic List as specified in the Link Supervision Interface:

- a) Periodic\_Data are ranked by their Individual Period  $T_{ip}(i)$ ;
- b) a Cycle groups Periodic\_Data with the same Individual Period and receives the same name as its Individual Period in multiple of the Basic Period (e.g. Cycle\_2 when polled every 2 ms and the Basic Period is 1 ms);
- c) a Macro\_Cycle consists of all cycles which fill a Macro\_Period, therefore it takes:
  - 1024 Basic Periods if  $T_{bp} = 1$  ms or
  - 512 Basic Periods if  $T_{bp} = 2$  ms;
- d) Basic Periods are numbered BP(j), where
  - $j = (0 \dots \text{macro}-1)$ ,
  - BP(0) = first period of a Macro\_Cycle,
  - BP (macro-1) = last period of a Macro\_Cycle;
- e) a turn is defined as the number of Macro\_Cycles after which a mastership transfer shall take place. A turn may be stopped after each Macro\_Cycle by a Link Supervision Interface command;
- f) a Cycle may be split over several Basic Periods, into Subcycles. Cycle\_n has n Subcycles, for instance Cycle\_16 has 16 Subcycles, of which some or all may be void. A Subcycle is identified by its index, for instance Cycle\_4.1 being the second Subcycle of Cycle\_4;
- g) the last Sporadic Phase of every Macro\_Cycle is reserved for mastership transfer, even if not used for mastership transfer in this Macro\_Cycle.

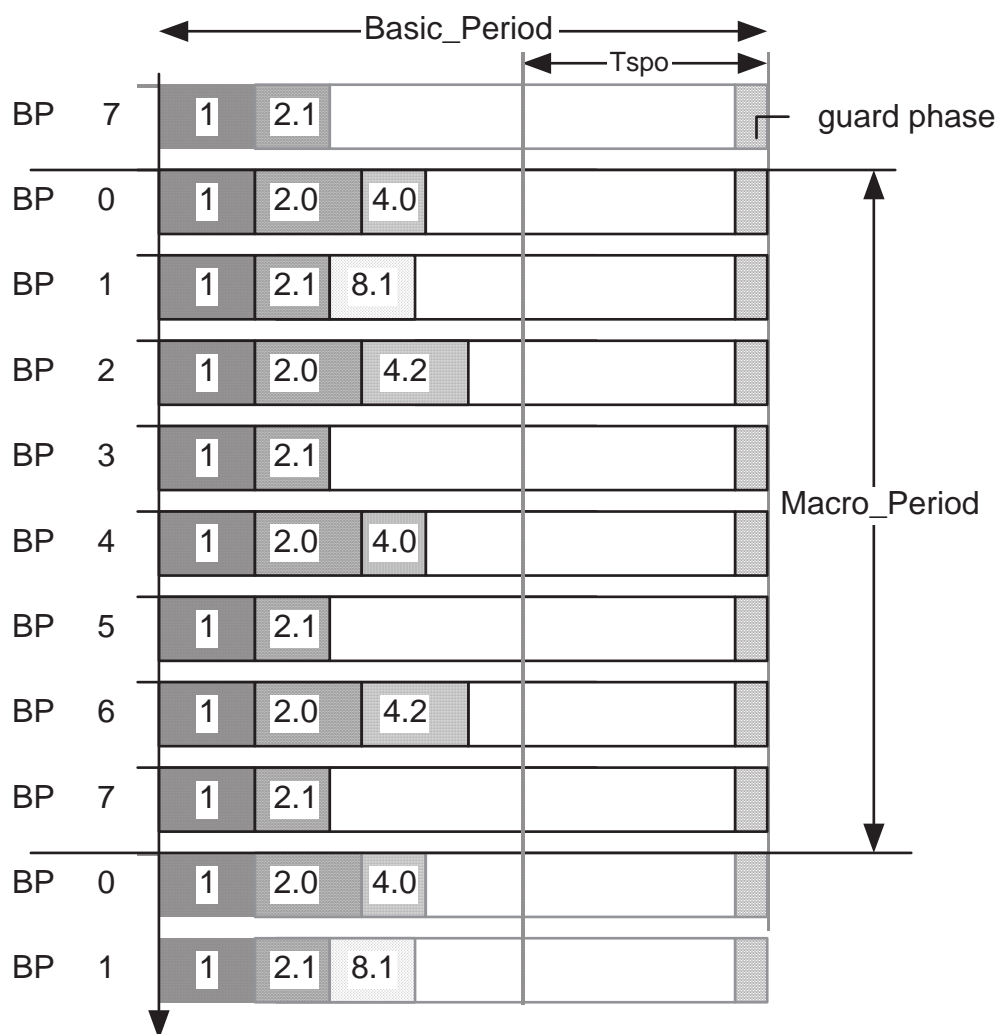
NOTE There are no constraints about the split of Subcycles over the different periods, but for performance reasons, it is recommended to spread the cycles evenly over the periods.

#### 8.2.3.2 Construction of a Macro\_Cycle

The Periodic List shall contain two sub-lists for each Macro\_Cycle:

- a) the Cycle\_List, containing the Master Frames belonging to each of the 11 Cycles (Cycle\_1 to Cycle\_1024) and
- b) the Split\_List, containing the number of frames of each cycle to send in each Subcycle.

EXAMPLE A simplified construction of a Macro\_Cycle consisting of 8 Cycles is shown in Figure 50.



**Figure 50 – Example of construction of the Macro\_Cycle**

NOTE The Periodic List of the bus administrator of this example contains:

- the four Cycle\_Lists, one for each cycle 1, 2, 4 and 8;
- the three Split\_List indicating the number of frames for
  - Subcycles 2.0 and 2.1 (Cycle 1 has no split, it is present in every period),
  - Subcycles 4.0, 4.1 (void here), 4.2 and 4.3 (void here),
  - Subcycles 8.0, 8.1, 8.2, 8.3, 8.4, 8.5, 8.6 and 8.7 (all voids here except 8.1).

### 8.3 Event Polling

#### 8.3.1 Group\_Address

NOTE For the purpose of Event\_Arbitration, the master selects groups of devices which share the same lowest significant address bits.

Devices on the bus which differ in the most significant bits of their Device Address form a Group, which is identified by its Group\_Address.

The Group\_Address is characterised by

$G = \{M, C\};$

- M being the number of most significant digits in which the device addresses differ;
- 2M being the number of possible devices in the group ( $M = 1..12$ );

- C being the binary number expressed by the common least significant bits of the address.

EXAMPLE The devices with binary address:

1111 0000 0101,  
1000 0000 0101 and  
0101 0000 0101

belong to the same group {M = 4, C = 5}.

### 8.3.2 Event\_Round

An Event\_Round collects all event requests of a given priority pending at the time the Event\_Round begins.

#### 8.3.2.1 Start

The master shall start an Event\_Round by sending a General\_Event\_Request, which defines:

- a) that this is a new Event\_Round and not a continuation {new\_round};
- b) that an answer is required {answer\_now};
- c) one of two event priority levels for this Event\_Round {high\_priority or low\_priority}.

This General\_Event\_Request addresses Group {M = 12}, i.e. all possible 4096 devices.

#### 8.3.2.2 Event clearing and enabling

A device shall participate in an Event\_Round if it has an event of the corresponding priority pending at the time it receives a General\_Event\_Request {new\_round}.

A participating device shall respond with its Event\_Identifier\_Response to an Event\_Request with the following F\_codes:

- a) F\_code 9: General\_Event\_Request (if "answer\_now" and priority matches);
- b) F\_code 13: Group\_Event\_Request (if the Group\_Address matches that of this device);
- c) F\_code 14: Single\_Event\_Request (if the Device Address matches that of this device).

A device shall clear the corresponding event when it receives an Event\_Read\_Request which matches the F\_code and address of the Event\_Identifier\_Response.

NOTE 1 A slave is not informed that the master has read its Event\_Identifier\_Response.

NOTE 2 Several devices can report the same event. In this case, all devices with that event clear it when the event is read since no one knows that it was the one which signalled that event.

#### 8.3.2.3 Recognised situations

An Event\_Request is defined as one of General\_Event\_Request, Group\_Event\_Request, Single\_Event\_Request.

The master shall distinguish the following situations occurring in response to an Event\_Request, within the limits of T\_reply:

- a) correct: a correct Slave Frame with 16 bits was received (6.3.1);
- b) silence: no frame has been received (6.3.3);
- c) collision: all other cases (5.1.9).

#### 8.3.2.4 Event search algorithm

In case a collision comes in response to a `General_Event_Request`, the master shall search all pending events by sending `Group_Event_Request` or `Single_Event_Request`, with the aim of restricting the number of devices participating in the poll.

NOTE The algorithm for searching the pending events is not specified. A general method is given in 8.3.3 as a recommended practice, but other algorithms are permitted. In particular, algorithms may take advantage of the fact that the identity of participating devices is known and that events tend to occur in bursts for Message Data transfers.

#### 8.3.2.5 Event\_Arbitration termination

The master shall terminate `Event_Arbitration`:

- a) when it receives an `Event_Identifier_Response`;
- b) when it detects either a collision or a silence in response to an `Single_Event_Request`.

In configurations where the event-signalling devices have a limited address range (e.g. 1..255), the master may consider that the search terminates when no response comes to a `Group_Event_Request` which addresses that device group.

#### 8.3.2.6 Event\_Round interruption and continuation

The high priority `Event_Round` shall be suspended only by the Periodic Phase.

In case both a high and a low priority are configured, a high priority `Event_Round` shall be started after each time a low priority event has been read, suspending the low priority `Event_Round`.

The low-priority `Event_Round` shall be resumed with a `General_Event_Request` (NOT `new_round`, `answer_now`).

#### 8.3.2.7 Event\_Round termination

When the search tree has been run completely, and after all events have been read, the master shall issue a `General_Event_Request` {NOT `new_round`, `answer_now`} to check for devices left out in the `Event_Round`.

If no answer comes to that `General_Event_Request`, the master shall close the `Event_Round`.

Otherwise, it shall resume the `Event_Round` from the beginning.

If the same event is reported more than twice during `Event_Round` termination, the master shall close the `Event_Round` and report an error.

NOTE Devices may be left out in a round due to a wrongly interpreted situation, in particular:

- undetected collision (correct frame received although several devices transmitted);
- undetected silence (disturbance caused a collision to appear when there is none);
- error during transmission of the `Event_Identifier_Response`;
- two or more devices requesting the same event.

#### 8.3.2.8 Event\_Read

The master shall re-send the received `Event_Identifier_Response` (Slave Frame) as an `Event_Read_Request` (as a Master Frame).

The master shall not repeat Event\_Read\_Request even if it does not receive a correct Event\_Read\_Response in response to it, for example in case of transmission error.

NOTE The master may send an Event\_Read\_Request just after receiving an Event\_Identifier\_Response, or it may bundle the Event\_Read\_Requests of several received Event\_Identifier\_Responses after the end of the Event\_Round, as the application decides.

### 8.3.3 Recommended event search algorithm

The following is a recommended algorithm.

#### 8.3.3.1 First Event\_Arbitration

If a collision occurs in response to a General\_Event\_Request, the master executes an Event\_Arbitration to single out one event.

To this effect, the master sends a Group\_Event\_Request, addressing Group {N-1, 0}.

NOTE The first Group\_Event\_Request addresses the 2048 devices with an even address.

#### 8.3.3.2 Down Event\_Arbitration

Assuming that the master sent a Group\_Event\_Request to Group {N, C}:

- if a single response comes, the master returns that response;
- if collision occurs, the master issues a Group\_Event\_Request to a group half the size of the former group, and whose common address is the same:

$$G\_new = G\_old \{M = M-1, 0 + C\}$$

NOTE 1 This Group\_Event\_Request addresses the one half of the preceding group, whose common most significant address bit is "0".

- if silence follows a Group\_Event\_Request, the master issues a Group\_Event\_Request to a group potentially half the size of the former Group, and whose common address is:

$$G\_new = G\_old \{M = M-1, (2^{(12-M)} + C)\}$$

NOTE 2 This Group\_Event\_Request assumes that the event comes from the other group, and starts with the half-group whose common most significant common address bit is "0".

The master sends a Single\_Event\_Request where a collision occurs on the second most significant Device Address bit (Group\_Event\_Request with bit 4=0).

#### 8.3.3.3 Up Event\_Arbitration

The master continues the next Event\_Arbitration at one level lower in the tree from the level where the most recent collision occurred.

To this effect, the master sends the Group\_Event\_Request with the least significant bit of the Group\_Address set to the opposite value.

$$G\_new = G\_old \{M = M/2, 2^{(12-M)} + C\}$$

If this Group\_Event\_Request is responded by a silence or an Event\_Identifier\_Response, the master sends a General\_Event\_Request and restarts arbitration from the top.

If the Group\_Event\_Request is responded by a collision, the master increases the size of the Group\_Address by one and performs an Event\_Arbitration on that branch:

$$G\_new = G\_old \{M = M+1, (2^{(12-M)} + C)\}$$

### 8.3.4 Supervisory Data frames for Event\_Arbitration

#### 8.3.4.1 General\_Event\_Request

The General\_Event\_Request shall consist of four fields of 4 bits each, as shown in Figure 51:

The bits 0 to 3 are the F\_code = 9;

The next 4 bits are the Event\_Mode, encoded as:

- EM = '0000'B: answer\_now, NOT new\_round
- EM = '0001'B: answer\_now, new\_round
- EM = '0010'B: NOT answer\_now, NOT new\_round
- EM = '0011'B: NOT answer\_now, new\_round
- EM = '0100'B to '1111'B: reserved.

The next 4 bits are the Event\_Type, encoded as:

- ET = '0000'B: high priority Event\_Round (high\_priority)
- ET = '0001'B: low priority Event\_Round (low\_priority)
- ET = '0010'B to '1111'B: reserved.

The last 4 bits are reserved.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	1	0	0	QT	NR	0	0	0	HP	0	0	0	0
F_code = 9				Event_Mode				Event_Type				reserved			

**Figure 51 – General\_Event\_Request frame format**

NOTE General\_Event\_Request is the only Master Frame without an address, since it is by default broadcast to all devices.

#### 8.3.4.2 Group\_Event\_Request

The master shall poll a group of devices belonging to a group {M, C} by a Group\_Event\_Request consisting of three fields, as shown in Figure 52:

- a) the bits 0 to 3 are the F\_code = 13;
- b) the next bits, starting with bit 4, consisting of a set of contiguous "1" closed by a "0" build the mask. The location of the "0" bit depends on the step in the search tree. The number of "1" is equal to M-1;
- c) the remaining low order bits to the right of the zero until bit 15 define the group address C.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	1	1	1	1	1	1	0	A	B	C	D	E	F
F_code = 13				mask (don't care)						Group_Address					

**Figure 52 – Group\_Event\_Request frame (M = 6, C = ABCDEF)**

EXAMPLE Assuming all devices have an event pending, if the address is:



1111 1111 1110, all devices respond (= General\_Event\_Request {NOT new\_round, answer\_now});  
1111 1111 1101, all devices with an odd address respond;  
1111 1001 1001, all devices with an address ending in 01 1001 respond;  
0111 0110 1000, two devices which differ in their most significant bit respond.

In the above Figure 52, the Group is {M = 6, C = ABCDEF} and it addresses 64 devices.

8.3.4.3 Single\_Event\_Request

The master shall poll a single device for its event by an Single\_Event\_Request, which is a Master Frame with the following structure, as shown in Figure 53:

- the bits 0 to 3 are the F\_code = 14;
- the next 12 bits define an individual device address.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	0	A	A	A	A	A	A	A	A	A	A	A	A
F_code = 14				Device Address											

Figure 53 – Single\_Event\_Request frame

8.3.4.4 Event\_Identifier\_Response

A slave with a pending event of the priority established at the beginning of the Event\_Round shall respond with an Event\_Identifier\_Response to

- a) a General\_Event\_Request {answer\_now};
- b) a Group\_Event\_Request with the group address matching its own; or
- c) a Single\_Event\_Request whose address matches the 12-bit Device Address;

The Event\_Identifier\_Response shall be a 16-bit Slave Frame, the content of which are interpreted as a 4-bit F\_code followed by a 12-bit address, as shown in Figure 54.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
F_code				logical or physical address											

Figure 54 – Event\_Identifier\_Response frame

A slave shall not use the following F\_codes in its Event\_Identifier\_Response:

- F\_code = 8 (Mastership\_Transfer);
- F\_code = 9 (General\_Event\_Request);
- F\_code = 13 (Group\_Event\_Request);
- F\_code = 14 (Single\_Event\_Request).

8.3.4.5 Event\_Read\_Request

The master shall re-send a correctly received Event\_Identifier\_Response as an Event\_Read\_Request .

The Event\_Read\_Request shall have the same format as the Event\_Identifier\_Response, except that it has a Master Start Delimiter instead of the Slave Start Delimiter.

### 8.3.4.6 Event\_Read\_Response

The format of the Event\_Read\_Response is that of any Slave Frame.

NOTE The Event\_Read\_Response will most likely be a Message Data frame.

## 8.4 Devices\_Scan

### 8.4.1 Device\_Status

All devices shall implement a source port for their Device\_Status, containing a 16-bit word which indicates their status, and which can be polled.

#### 8.4.1.1 Device\_Status\_Request

The master shall poll the Device\_Status of a device with a Device\_Status\_Request, which is a Supervisory\_Data\_Request with F\_code = 15, and which contains the 12-bit address of the device, as Figure 55 shows.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	1	A	A	A	A	A	A	A	A	A	A	A	A
F_code = 15				Device Address											

Figure 55 – Device\_Status\_Request

#### 8.4.1.2 Device\_Status\_Response

##### 8.4.1.2.1 General format

A slave shall respond to a Device\_Status\_Request with a Device\_Status\_Response, which is a Supervisory\_Data\_Response consisting of the following fields as shown in Figure 56.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SP	BA	GW	MD	T	T	T	T	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capabilities				class_specific				common_flags							

Figure 56 – Device\_Status\_Response

##### 8.4.1.2.2 Capabilities field

The first four bits shall define the basic capabilities of a device:

- a) SP bit 0 set to "0" indicates the Device\_Status and Process Data capability, bit 0 set to "1" indicates a special device, the next 3 bits are not specified;
- b) BA bit 1 set to "1" indicates the Bus Administrator capability;
- c) GW bit 2 set to "1" indicates the Gateway capability;
- d) MD bit 3 set to "1" indicates the Message Data capability.

##### 8.4.1.2.3 Specific field

This 4-bit specific field depends on the capabilities field. It distinguishes four specific fields:

- a) Class 1 device;
- b) device with Message Data capability;
- c) device with Bus Administrator capability;

d) device with Gateway capability.

#### 8.4.1.2.3.1 Class 1 device specific field

A device of Class 1 shall respond with the "capabilities" field set to "0000", as shown in Figure 57.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capabilities				class_specific				common_flags							

**Figure 57 – Device\_Status of Class 1 device**

The specific field shall be "0000".

Two bits of the Device\_Status word are fixed:

- SER = 0 (no reservation for service possible);
- FRC = 0 (no forcing possible).

#### 8.4.1.2.3.2 Message Data specific field

A device capable of Message Data communication according to the Real-Time Protocols of the TCN shall respond with Bit 3 set to "1", as shown in Figure 58.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	x	x	1	x	X	x	x	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capabilities				class_specific				common_flags							

**Figure 58 – Device\_Status of Class 2/3/4/5 device**

#### 8.4.1.2.3.3 Bus Administrator specific field

A device with the Bus Administrator capability shall respond with Bit 1 set to "1", and set the four bits in the "class\_specific" field, as shown in Figure 59:

- AX1: Bit 4 second least significant bit of the Actualisation\_Key of the Periodic List;
- AX0: Bit 5 least significant bit of the Actualisation\_Key of the Periodic List;
- ACT: Bit 6 is set if the device is actualised and in condition of becoming master;
- MAS: Bit 7 is set if the device is the current master.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	x	x	AX1	AX0	ACT	MAS	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capabilities				Class_specific				common_flags							

**Figure 59 – Device\_Status of a device with Bus Administrator capability**

#### 8.4.1.2.3.4 Gateway specific field

A device with the Gateway capability shall respond with its Bit 2 set to "1".

If the device has at the same time the Bus Administrator capability, the "specific" field shall be that of a Bus Administrator, as shown in Figure 60.

If the device is a gateway without Bus Administrator capability, the device shall set the "class\_specific" field as shown in Figure 60:

- STD: Bit 4 set to "1" indicates a static disturbance (remote bus down);
- DYD: Bit 5 set to "1" indicates a dynamic disturbance (e.g. train inauguration);
- RV6 Bit 6 is "0" (reserved); and
- RV7 Bit 7 is "0" (reserved).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	x	1	x	STD	DYD	RV6	RV7	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capabilities				class_specific				common_flags							

**Figure 60 – Device\_Status of a device with Gateway capability**

NOTE The STD and DYD bits can be read directly on a gateway which has the Bus Administrator capability.

#### 8.4.1.2.4 Common field

These eight bits are common to all devices:

LAT: Line\_A\_Trusted:

shall be asserted if the Master Frame of this telegram was received over Line\_A, negated if it was received over Line\_B;

RLD: Redundant\_Line\_Disturbed:

shall be set if the Observed\_Line is disturbed (5.2.5);

SSD: Some\_System\_Disturbance:

shall be set when the sink time supervision of any Port triggered and reset when all configured Ports operated normally;

SDD: Some\_Device\_Disturbance:

shall be set by a device malfunction (for instance: ROM or RAM checksum error) or a fault outside of the device (for instance: damaged sensors, ...) and reset when the fault is removed;

ERD: Extended\_Reply\_Delay:

shall be asserted if  $T_{ignore} > T_{reply\_def}$ , negated otherwise (6.2.4.2);

FRC: forced device:

shall be asserted if any Port has been forced to an imposed value and negated if all Ports attached to the MVB are in the unforced state;

DNR: Device\_Not\_Ready:

shall be asserted if the device is not operational (for instance application not running) but able to operate normally on the bus, and negated if the device is operational;

SER: System Reserved:

shall be set if the device has been reserved for exclusive use and reset when this exclusive use is lifted or timed-out.

#### 8.4.2 Device\_Status protocol

When receiving a Master Frame with F\_code = 15 (Device\_Status\_Request) and with its own address, a device shall reply with its Device\_Status\_Response.

NOTE 1 The destination of the Device\_Status frames is normally the master, but it can also be a monitoring device which operates by invitation.

NOTE 2 Reading the Device\_Status may cause clearing of the RLD bit in Class 1 devices and some other devices, see 5.2.5.

### 8.4.3 Devices\_Scan protocol

NOTE A configured master holds a list of the devices configured by Network\_Management, the Known\_Devices\_List. The list of found devices, the Devices\_List may differ from the Known\_Devices\_List because of device failure or configuration error.

To supervise the bus, the master shall regularly poll the Device\_Status of all devices and build the Devices\_List, which contains the address and Device\_Status of each found device.

The master shall list the devices with Bus Administrator capability it finds in the Bus Administrators\_List.

The master shall scan the known devices in its Known\_Devices\_List, and also unknown devices, to check for insertion of new devices.

The scan rate for known and unknown devices is a configuration parameter.

It is recommended that the Devices\_Scan be performed at the rate of 64 devices every 512,0 ms.

A device which ceases to respond for three consecutive polls shall be removed from the Devices\_List.

## 9 Mastership transfer

NOTE Mastership Transfer describes the protocols which select a master from one of several bus administrators, and ensures mastership transfer at the end of a turn or upon the occurrence of a failure.

### 9.1 Mastership transfer operation

#### 9.1.1 Bus administrator configuration

##### 9.1.1.1 Bus\_Configuration

The information which a bus administrator needs for regular operation (T\_reply, Periodic List, etc.), shall be contained in the Bus\_Configuration data structure.

The Bus\_Configuration shall be identified by an Actualisation\_Index, which identifies its version and which shall be incremented by one for each new version loaded.

The Bus\_Configuration may be in addition identified by a 15-bit Actualisation\_Key, which identifies its version of the configuration and which (if used) shall be incremented by one for each new version loaded into the bus administrator. The two lowest significant bits of the Actualisation\_Key may be used as Actualisation\_Index.

A bus administrator in possession of a Bus\_Configuration (10.4.3) is said to be "configured". It shall set the ACT bit in its Device\_Status to "1".

An "unconfigured" bus administrator only executes the Event\_Round and the Devices\_Scan. It shall set the ACT bit in its Device\_Status to "0".

A master shall assume that another bus administrator is "actualised" if its Actualisation\_Index is one higher or equal to its own.

A bus administrator shall revert to the "unconfigured" state if the Actualisation\_Index has been modified four times in sequence, but the bus administrator has not become master.

NOTE The generation of the Actualisation\_Key is at the charge of Network Management.

#### 9.1.1.2 Bus Administrators\_List

Each bus administrator shall hold a circular list of all other bus administrators, called the Bus Administrators\_List, ordered as the token circulates, and preferably structured in ascending order of the device addresses.

NOTE The Bus Administrators\_List may be part of the Bus\_Configuration, or may be generated on-line by the Devices\_Scan.

### 9.2 Mastership transfer specifications

NOTE The bus administrator states are described in the SDL syntax in Figure 61.

#### 9.2.1 States

##### 9.2.1.1 State "STANDBY\_MASTER"

In this state, the bus administrator does not exercise mastership. It shall go to the state "REGULAR\_MASTER":

- if it receives no Master Frames during a time-out  $T_{\text{standby}}$ , or
- if it receives a mastership offer from the current master to which it can respond (i.e. the bus administrator is "actualised").

##### 9.2.1.2 State "REGULAR\_MASTER"

In this state, the bus administrator shall exercise mastership (it executes the Periodic Polling, Event Polling and Devices\_Scan).

If it detects a Master Frame which it did not send itself, the bus administrator shall assume a master collision, and return to the state "STANDBY\_MASTER".

After its turn elapses or after a "resign" command, the bus administrator shall go to the state "FIND\_NEXT".

##### 9.2.1.3 State "FIND\_NEXT"

In this state, the bus administrator shall fetch the address of the proposed master in its Bus Administrators\_List and poll its Device\_Status.

The proposed master is a valid bus administrator if its Device\_Status states that:

- a) it has the Bus Administrator capability ( $BA = 1$ );
- b) it is configured ( $ACT = 1$ );
- c) it has an Actualisation\_Index equal to or higher by 1 than that of the current master, (Modulo 4).

The bus administrator shall send a Mastership\_Transfer\_Request to itself and then to the valid bus administrator and go to the state "INTERIM\_MASTER".

Otherwise, the bus administrator shall return to the state "REGULAR\_MASTER" and it shall try the next bus administrator in its Bus Administrators\_List at the next turn.

#### **9.2.1.4 State "INTERIM\_MASTER"**

In this state, the bus administrator shall expect the Mastership\_Transfer\_Response from the next bus administrator.

When it receives an (accept or reject) response or when the time-out elapses, the bus administrator shall go the state "STANDBY\_MASTER". If the response of the other bus administrator is "reject", the bus administrator shall report an error.

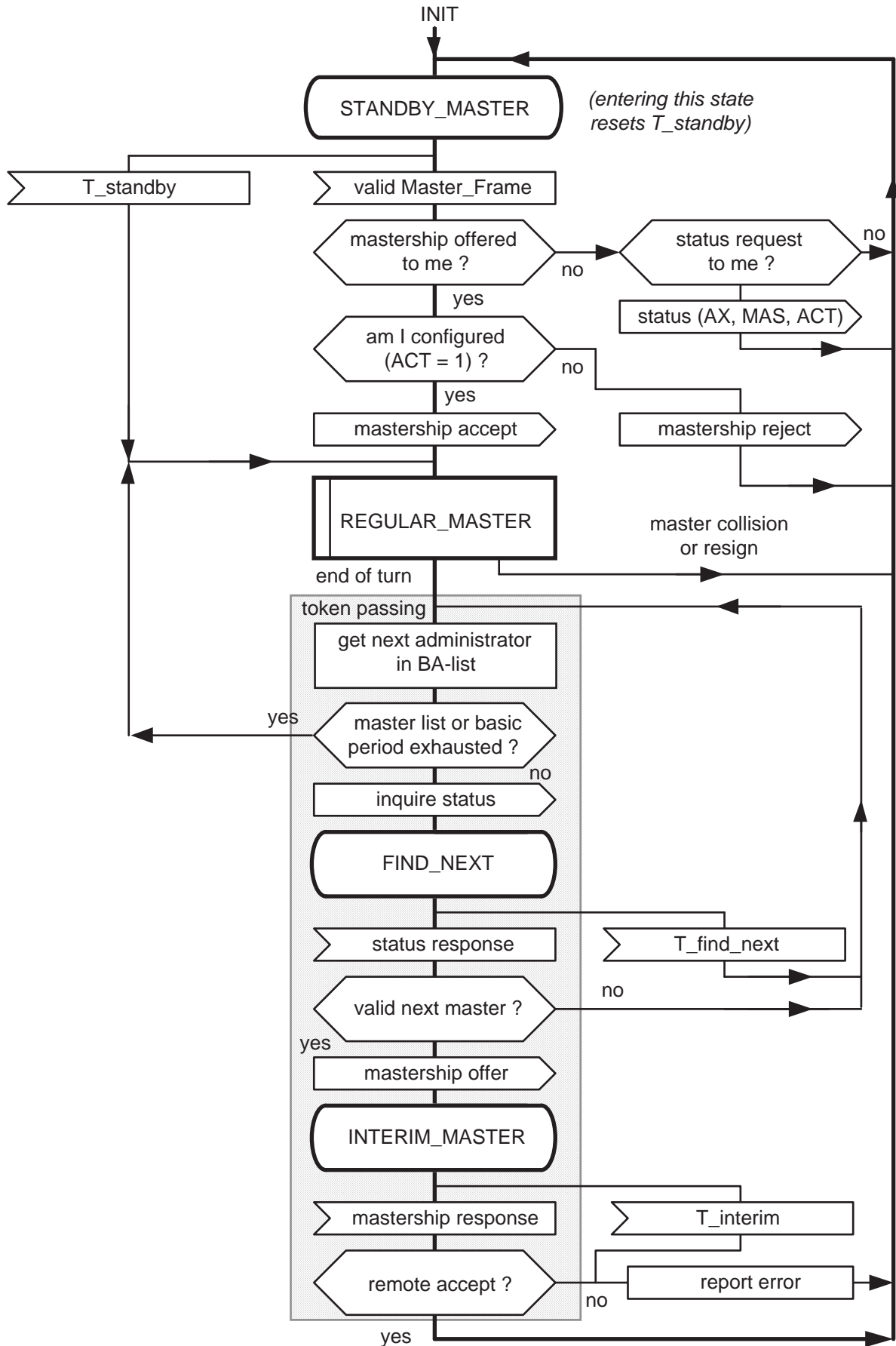


Figure 61 – Mastership Transfer states



9.2.2 Time-outs for mastership transfer

9.2.2.1 Turn duration

A turn shall last a multiple of 1024 ms and shall not exceed  $256 \times 1024$  ms.

9.2.2.2 Time-outs of "STANDBY\_MASTER"

T\_standby shall be equal to

- $(T_{\text{alive}} \times 2 \times (1 + \text{rank\_in\_ba\_list}))$  for a configured bus administrator; or
- $(T_{\text{alive}} \times 2 \times (\text{ba\_adr} + 15))$  for an unconfigured bus administrator; or
- infinite for a disabled bus administrator

where

T\_alive is the maximum interval between Master Frames;

rank\_in\_ba\_list is the order of that device in the Bus Administrators\_List;

ba\_adr is the Device Address of the bus administrator.

NOTE Allocating the lower Device Addresses to the bus administrators speeds up recovery.

9.2.2.3 Time-outs of "FIND\_NEXT"

T\_find\_next shall be equal to T\_reply

9.2.2.4 Time-outs of "INTERIM\_MASTER"

T\_interim shall be equal to T\_reply.

9.3 Supervisory data frames for mastership transfer

9.3.1 Device\_Status telegram

A master shall check if the proposed master is active and properly configured by reading the Device\_Status of the proposed master through a Device\_Status\_Request, as shown in Figure 62.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
F_code = 15				Device Address of proposed master											

Figure 62 – Device\_Status\_Request (sent by current master)

The proposed master shall respond with a Device\_Status\_Response, as shown in Figure 63.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x	1	x	x	AX1	AX0	ACT	MAS	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capabilities				class_specific				common_flags							

Figure 63 – Device\_Status\_Response (sent by proposed master)

### 9.3.2 Mastership transfer telegram

The current master shall send a Mastership\_Transfer\_Request containing, as shown in Figure 64:

- the F\_code = 8;
- the address of the proposed master or its own address (self-transfer).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
F_code = 8				Device Address of proposed master											

**Figure 64 – Mastership\_Transfer\_Request (sent by current master)**

The master or proposed master shall respond with a Mastership\_Transfer\_Response, which is a Supervisory\_Data\_Response consisting, as shown in Figure 65, of

- Bit 0 = ACP,  
if ACP = 1, indicates that the bus administrator accepts mastership;  
if ACP = 0, indicates that the bus administrator rejects mastership;
- Bit 1 to Bit 15 contain the Actualisation\_Key of the proposed master (or 0 if it is not used).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ACP	Actualisation_Key														

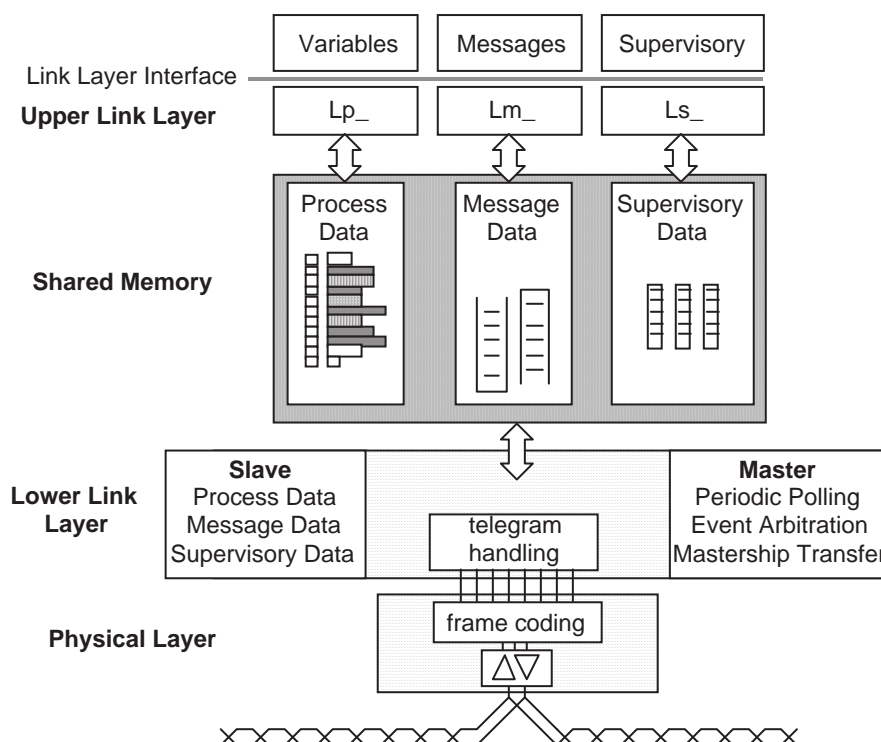
**Figure 65 – Mastership\_Transfer\_Response (sent by proposed next master)**

## 10 Link Layer Interface

### 10.1 Link Layer layering

The Link Layer Interface of the MVB shall provide three services as shown in Figure 66:

- the Link Process Data Interface (LPI), which provides a set of Ports in which Process Data are stored. This interface is specified in Clause 6 of IEC61375-2-1 and only the parameters specific to the MVB are specified in this standard;
- the Link Message Data Interface (LMI), which provides a set of queues for Message Data, used by the upper layers (network, transport, session). This interface is specified in Clause 6 of IEC61375-2-1 and only the parameters specific to the MVB are specified in this standard;
- the Link Supervision Interface (LSI), which provides the services for the configuration of the link layer and for the supervision of the bus. This interface is specific to the MVB and is specified in 10.4.



**Figure 66 – Link Layer Layering**

## 10.2 Link Process Data interface

The Process Data interface is specified in Clause 6 of IEC61375-2-1. The following defines parameters particular to the MVB.

The length of a Dataset shall be limited to 256 bits.

The Port\_Address of a dataset shall be 12-bits long. It shall be the same as the address used to transmit the dataset over the bus.

The number of Datasets per Traffic\_Store shall be limited to 4095.

The dataset with address 4095 shall be reserved for synchronisation and time distribution.

Where exact synchronisation is required, the end of the Master Frame shall serve as a reference point.

**NOTE** A Dataset is a group of variables transmitted in the same Process Data frame.

## 10.3 Link Message Data interface

### 10.3.1 General

The Link Message Data Interface provides services for sending Message Data frames and services for retrieving received Message Data.

The Link Message Data Interface provides the basic service on which the message services are built.

The Link Message Data Interface (LMI) is specified in Clause 6 of IEC 61375-2-1. The following specifies which parameters are particular of the MVB.

This interface is not implemented in devices lacking the Message Data capability.

### 10.3.2 Priority

The Link Message Data Interface builds on an output queue, which may be attached to either of the two priority levels (ET = 0 and ET = 1) of the sporadic medium access, and on an input queue, which is not prioritised.

The allocation of the priorities is application dependent.

### 10.3.3 Packet size

The "SZ" field in a void packet shall be set to zero.

The maximum length the "SZ" field shall indicate is 27 (32 octets – 5 octets of link header, including 1 octet for the "SZ" field).

### 10.3.4 Protocol\_Type

The value of the Protocol\_Type for Real-Time Protocols shall be '1000'B. Other values for the Protocol\_Type are reserved for future use.

### 10.3.5 Message Transport Protocol

An MVB device shall announce a packet size of 27 octets when opening a connection.

When responding to a connection opening, an MVB device shall specify a packet size equal to 27 octets.

## 10.4 Link Supervision Interface

### 10.4.1 General

The Link Supervision Interface is specific to the MVB.

It provides general services for configuration and inspection of the Link Layer and event reporting.

The Link Supervision Interface is grouped into 6 services:

- a) MVB\_Status: expresses the state of the link layer;
- b) MVB\_Control: controls the operation of the device;
- c) MVB\_Devices: provides the Device\_Status of the other devices on the same bus;
- d) MVB\_Administrator: sets up the bus administrator;
- e) MVB\_Report: event reporting.

The access to these objects may be through interface procedures or may be hard-wired. Whichever the method, a device shall be able to provide the functionality described.

The procedures of this interface are prefixed by ls\_v\_ (link supervision, vehicle bus).

The format of the parameters of the following interface procedures is not indicated. However, the Train Network\_Management document (Clause 8 of IEC 61375-2-1) proposes a format which is recommended as a parameter format.

### 10.4.2 Link Supervision Interface procedures

The services are defined by data structures which can be read and written with a procedure of the following form:

Definition	MVB link layer access procedure
Syntax	<pre> LS_RESULT          ls_v_&lt;service_name&gt;                     (&lt;parameters&gt;) </pre>

The possible results are listed in Table 9.

**Table 9 – LS\_RESULT encoding**

Constant	Code	Meaning
LS_OK	0	success
LS_BUSY	1	try again later
LS_CALL_SEQUENCE	2	wrong command sequence

Only the parameters of the procedures are defined here.

#### 10.4.3 MVB\_Status

The MVB\_Status object describes the static and dynamic parameters of the link layer. This structure is intended to be transmitted without modifications.

Every device shall have an MVB\_Status object with the attributes listed in Table 10.

**Table 10 – MVB\_Status object**

Attribute	Type	Meaning
device_address	WORD16	address of the device on the bus. Most significant four bits = '0000'B.
mvb_hardware_name	STRING32	identifies the MVB controller, the hardware and its version.
mvb_software_name	STRING32	identifies the MVB software version, preferably in the format: xxxx-Vz.z-dd.mm.yy.
device_status	Device_Status	copy of the 16-bit Device_Status.
reserved	WORD8 (=0)	used for alignment.
t_ignore	UNSIGNED8	configured time-out value for the delay between a Master Frame and a Slave Frame on that device in multiples of 1 µs. if t_ignore = 0, the value is the default value T_reply_def = 42,7 µs.
lineA_errors	UNSIGNED32	32-bit counter incremented each time the Line_Disturbed signal of Line_A is set; this counter does not wrap-around when reaching its highest value.
lineB_errors	UNSIGNED32	32-bit counter incremented each time the Line_Disturbed signal of Line_B is set; this counter does not wrap-around when reaching its highest value.

#### 10.4.4 MVB\_Control

The MVB\_Control object controls the static and dynamic parameters of the link layer. This structure is intended to be transmitted without modifications.

Every device shall have an MVB\_Control object including the parameters listed in Table 11.

**Table 11 – MVB\_Control object**

Attribute		Type	Meaning
device_address		WORD16	address of device on the bus, most significant four bits = 0 (if software-modifiable).
t_ignore		UNSIGNED16	value of the time-out T_ignore to be used, in multiples of 1 $\mu$ s default (t_ignore = 0): T_reply_def = 42,7 $\mu$ s recommended values: 22,0 $\mu$ s, 64,0 $\mu$ s and 86,0 $\mu$ s maximum value: 255,0 $\mu$ s.
command		BITSET8	command to be performed by the MVB Link Layer:
	aon	(0)	(Bus Administrator only) enables the bus administrator
	aof	(1)	(Bus Administrator only) disables the bus administrator
	spl	(2)	(Bus Administrator only) switch to new Periodic List (= 1). Causes the bus administrator to switch to the new Periodic List next time the bus administrator becomes master
	tms	(3)	(Bus Administrator only) transfer mastership to the next bus administrator in the Bus Administrators_List at the end of the macro-cycle following the request, retain mastership if other bus administrator does not accept it.
	sla	(4)	trust always Line_A, even if this line is disturbed (sla = slb = 0: pas d'action)
	slb	(5)	trust always Line_B, even if this line is disturbed (sla = slb = 1: normal redundant operation)
	cla	(6)	reset frame error counter of Line_A (cla = clb = 0: no action)
	clb	(7)	reset frame error counter of Line_B (cla = clb = 1: reset both counters)

NOTE On devices which are not bus administrators, redundant lines are switched when the bus administrator reads the Device\_Status.

#### 10.4.5 MVB\_Devices

The MVB\_Devices object contains in particular the list of the devices and their Device\_Status ordered by increasing address, indicating for each:

- the Device Address;
- the Device\_Status read during the latest Devices\_Scan.

Every bus administrator shall have an MVB\_Devices object with the attributes listed in Table 12.

**Table 12 – MVB\_Devices object**

Attribute		Type	Meaning
nr_devices		UNSIGNED16	number of devices in the Devices_List.
devices_list		ARRAY [nr_devices] OF	list of the found devices, comprising for each:
	device_address	WORD16	12-bit address of the device (four most significant bits = 0)
	device_status	Device_Status	copy of the 16-bit Device Status of the device.

#### 10.4.6 MVB\_Administrator

The MVB\_Administrator object contains in particular:

- the configuration parameters (Basic Period, T\_reply, Event strategy, etc.);
- the Known\_Devices\_List and the Devices\_Scan algorithm;
- the Periodic List, consisting of the Cycle\_Lists and the Split\_Lists;
- the Bus Administrators\_List.

Every bus administrator shall have an MVB\_Administrator object with the attributes listed in Table 13.

**Table 13 – MVB\_Administrator object**

Attribute	Type	Meaning
checksum0	UNSIGNED16	Checksum over this structure – application dependent. This word is located at offset 0 of this structure.
actualisation_key	UNSIGNED16	Version number of the configuration, as a 16-bit number. The most significant bit set to 1 indicates to transmit the least significant 15 bits in Mastership_Transfer_Response. The two least significant bits are copied to AX0 and AX1.
t_reply_max	UNSIGNED16	Maximum expected reply delay between a Master Frame and a Slave Frame, in multiples of 1µs default (t_reply = 0) = 42,7 µs, maximum = 255,0 µs. In case of Extended Reply Delay operation, this value shall be equal or greater to the largest value of the T_ignore time-out of any device in this configuration.
macro_cycles	UNSIGNED16	Number of basic periods after which the periodic traffic repeats itself (e.g. 512 for a 2,0 ms T_bp).
event_poll_strategy	ENUM16	'0000'H no event polling '4000'H high priority event polling only '8000'H low priority event polling only 'C000'H high and low priority event polling.
basic_period	UNSIGNED16	T_bp in microseconds (recommended values: 1000, 1250, 1042, 2000, 2043, 2500).
macrocycles_per_turn	UNSIGNED16	Number of Macro_Cycles after which the master shall offer mastership to the next bus administrator in its list.
devices_scan_strategy	ENUM16	0: scan all known devices addresses 1: scan all device addresses
reserved2	WORD16	unused
reserved3	WORD16	unused
reserved4	WORD16	unused
reserved5	WORD16	unused
known_devices_list_offset (offset_KDL)	UNSIGNED16	Offset (from checksum0) where the first address of the device list is located. The list size is equal to offset_RSL – offset_KDL.
reserved_list_offset (offset_RSL)	UNSIGNED16	unused (offset_RSL= offset_PL)
periodic_list_offset (offset_PDL)	UNSIGNED16	offset (from checksum0) where the first F_code and address of the first cycle is located. The list size is equal to offset_BAL – offset_PDL.

**Table 13** (continued)

Attributes		Type	Meaning
bus_administrators_list_offset (offset_BAL)		UNSIGNED16	offset (from checkword0) where the first address of the Bus Administrators_List is located.
devices_scan_list_offset (offset_DSL)		UNSIGNED16	offset (from checkword0), where the device scan allocation list is located (unused)
end_list_offset (offset_END)		UNSIGNED16	dummy offset indicating the size of the devices_scan_list and pointing to the next location after the last device of that list.
known_devices_list		ARRAY [nr_devices] OF	list of the known devices, (nr_devices = offset_KDL – offset_RSL), comprising for each:
	device_address	WORD16	12-bit address of the device (four most significant bits =0)
cycle_lists_offsets		ARRAY [11] OF	array of 11 elements, comprising for each one of the possible 11 cycles:
	cycle_n_offset	UNSIGNED16	offset (from beginning of Periodic List) where the first address of the cycle is located
split_list_offsets		ARRAY [5] OF	array of 5 elements, comprising for each:
	split_n_m_offset	UNSIGNED16	offset (from beginning of Periodic List) where the first address of the table split_n_m is located
cycle_lists		ARRAY[11] OF	array of 11 lists, each comprising
	cycle_n	ARRAY [cycle_size] OF	array of [cycle_size] values, (cycle_size is the offset between this list and the next) each comprising:
	f_code	ENUM4	F_code (any value) of the Master Frame to be sent
	address	UNSIGNED12	address in the Master Frame
split_lists		ARRAY [5] OF	array of five lists (n= 4, n=16, n=64, n= 256, n= 1024) and m = n/2, each comprising:
	split_n_m	ARRAY [n] OF	array of [n] values, each comprising:
	lower_count (n)	UNSIGNED8	number of frames of the lower cycle (n) to be put into each Basic Period in the most significant octet (this list is repeated twice; it is only half as long as the upper_count list).
	upper_count (m)	UNSIGNED8	number of frames of the upper cycle (m) to be put into each Basic Period
bus_administrators_list		ARRAY [nr_admins] OF	list of the known bus administrators, (nr_admins = devices_scan_list_offset – bus_administrators_list_offset), comprising for each:
	f_code	UNSIGNED4	F_code = 8 (ready for mastership transfer)
	device_address	UNSIGNED12	12-bit address of the device.

**10.4.7 MVB\_Report**

Definition	Reports are given to the application by calling a subscribed function of the type:
Syntax	<pre>typedef void      (* ls_v_report)                   (ls_report)</pre>

The value of the reports shall be as given as encoded in Table 14.



**Table 14 – LS\_V\_REPORT encoding**

Constant	Code	Meaning
LS_V_FRAME_COUNT	1	number of received frames reached 65526.
LS_V_HW_ERROR	2	some disturbance occurred
LS_V_CHANGE	3	(Bus Administrator only) change in devices list
LS_V_REPEATED_EVENT	4	(Bus Administrator only) a device keeps on signalling the same event
LS_V_SYNCHRO	5	synchronisation received
LS_V_MASTER_CHANGE	6	this device transferred mastership.

## 11 Real-Time Protocols

In the TCN architecture, WTB and MVB busses share the same Real-Time Protocols, which offer two communication services:

- a) Process Variables, a distributed, real-time database, periodically refreshed through broadcasting;
- b) messages, transmitted on demand either as:
  - unicast messages (point-to-point) or/and
  - multicast messages.

These services are defined in detail in Clause 6 of IEC 61375-2-1.

## 12 Gateway Function

Void.

## 13 Network Management

### 13.1 Contents of this clause

Train Network Management specifies a number of services to assist commissioning, testing, operation and maintenance of a Train Communication Network, such as:

- a) station identification and control;
- b) management of the Train Bus and Vehicle Bus link layers;
- c) distribution of routing and topography;
- d) remote reading and forcing of variables;
- e) downloading and uploading.

WTB and MVB common services are specified in Clause 8 of IEC 61375-2-1.

This clause defines MVB specific services.

### 13.2 MVB Managed objects

#### 13.2.1 MVB link objects

##### 13.2.1.1 MVB\_Status object

Each Station attached to an MVB shall implement a read-only MVB\_Status object.

The MVB\_Status object shall identify the hardware and software version, define dynamic and static parameters, disclose errors and statistics about the MVB Link\_Layer and contain the Device\_Status specified in IEC 61375-3-1 ("Multifunction Vehicle Bus").

NOTE The Agent accesses this object through the MVB link layer supervisory interface, Is\_v\_xxx.

#### **13.2.1.2 MVB\_Control object**

Each Station with an MVB attachment supporting the Bus\_Administrator capability shall implement a write-only MVB\_Control object.

Over this object, a Manager can command a Bus\_Administrator:

- a) to release control so that another Bus\_Administrator can take over, or
- b) to use a polling list which has been loaded.

#### **13.2.1.3 MVB\_Configuration object**

Each Station with an MVB attachment supporting the Bus\_Administrator capability shall implement the write-only Bus\_Configuration object, containing all parameters required for the Bus\_Administrator to operate.

The Bus\_Configuration shall comprise in particular the following sub-objects, which are treated as a whole and are covered by a common version control:

- a) the setup parameters of the Bus\_Administrator;
- b) the Known\_Devices\_List, which indicates which devices should normally be present;
- c) the Periodic\_List, which defines the polling strategy for Process Data;
- d) the Bus\_Administrators\_List, which defines the mastership transfer algorithm;
- e) the Devices\_List, which defines how known and unknown devices are scanned.

NOTE 1 Modifying the Periodic\_List on-line can lead to a non-deterministic operation or even cause a breakdown in traffic if configuration fails (e.g. too many variables included).

NOTE 2 The mastership transfer protocol ensures that once one Bus\_Administrator starts using the new configuration list, all others will do likewise.

#### **13.2.1.4 MVB\_Devices object**

Each Station with an MVB attachment supporting the Bus\_Administrator capability shall implement a read-only MVB\_Devices object, containing the list of the present devices together with their Device\_Status.

NOTE 1 The Bus\_Administrator builds this list by scanning the devices periodically.

NOTE 2 This object may also be implemented by a Station which is not a Bus\_Administrator, by observing the bus traffic.

### **13.3 MVB Services and management messages**

#### **13.3.1 MVB link services**

##### **13.3.1.1 Read\_MVB\_Status**

##### **13.3.1.1.1 Description**

The service reads the MVB\_Status, which is the status of an MVB link layer to which a Station is connected.

**13.3.1.1.2 Call\_Read\_Mvb\_Status**

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 10							
bus_id								reserved1							

Call\_Read\_Mvb\_Status ::= RECORD

```
{
  bus_id          UNSIGNED8 (0..15),  -- link layer identifier
  reserved1       WORD8 (=0)          -- reserved
}
```

**13.3.1.1.3 Reply\_Read\_Mvb\_Status**

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 10							
bus_id								reserved1							
reserved2															
device_address															
mvb_hardware_name: STRING32															
								CHARACTER8 or '00'H							
mvb_software_name: STRING32															
								CHARACTER8 or '00'H							
device_status															
reserved3								t_ignore							
lineA_errors															
lineB_errors															

Reply\_Read\_Mvb\_Status ::= RECORD

```
{
  bus_id          UNSIGNED8 (0..15),  -- identifier of the link
  reserved1       WORD8 (=0),         -- reserved
  reserved2       WORD16 (=0),        -- reserved
  device_address  WORD16,             -- address of the device on the
                                      bus
  mvb_hardware_name  STRING32,        -- descriptor of the MVB
                                      controller, including the
                                      hardware version
  mvb_software_name  STRING32,        -- descriptor of the MVB
                                      software version, in the
                                      format:
                                      xxxx-Vz.z-dd.mm.yy
  device_status    Device_Status,     -- copy of the 16-bit MVB
                                      Device_Status
  reserved3       WORD8 (=0),         -- reserved
  t_ignore        UNSIGNED8,          -- configured time-out value
                                      for the delay between a
                                      Master_Frame and a
                                      Slave_Frame on that device
                                      in multiples of 1,0 µs,
                                      Default value (t_ignore = 0)
                                      = 42,7 µs.
  lineA_errors     UNSIGNED32,        -- 32-bit counter incremented
                                      each time an erroneous frame
```

lineB\_errors                      UNSIGNED32

of any type is received over Line\_A; this counter does not wrap-around when reaching its highest value.  
 -- 32-bit counter incremented each time an erroneous frame of any type is received over Line\_B; this counter does not wrap-around when reaching its highest value.

}

The Device\_Status is defined by the following:

Device\_Status ::= RECORD

```

{
  capabilities                      BITSET4                      -- basic capabilities of an
                                                                    MVB device
                                                                    (all four bits 0 = class 1)

  {
    sp                      (0),                      -- special - Class 1 device
    ba                      (1),                      -- MVB Bus_Administrator
    gw                      (2),                      -- gateway or Train Bus node
    md                      (3)                      -- messages capability
  },
  class_specific                      ONE_OF                      -- Class1 (C1), Bus
                                                                    Administrator (BA) or
                                                                    gateway (GW)

  {
    [C1]                      WORD4,                      -- for class 1, = 0
    [BA]                      RECORD                      -- for bus administrator
    {
      act_index                      UNSIGNED2                      -- lowest two bits of
                                                                    actualisation index
      act                      BOOLEAN1                      -- 1 = actualised
      mas                      BOOLEAN1                      -- 1 = current master
    }
    [GW]                      RECORD                      -- for a gateway which is not
                                                                    bus administrator at the
                                                                    same time.

    {
      std                      BOOLEAN1                      -- 1 = static disturbance
      dyd                      BOOLEAN1                      -- 1 = dynamic disturbance
      rsv1                      BOOLEAN1                      -- reserved, =0
      rsv2                      BOOLEAN1                      -- reserved, =0
    }
  }
  common_flags                      BITSET8
  {
    lat                      (0)                      -- unused = 0
    old                      (1)                      -- observed line disturbed
    ssd                      (2)                      -- Some System Disturbance
                                                                    (fault in the controlled
                                                                    process, e.g. power loss).

    sdd                      (3)                      -- Some Disturbance of the
                                                                    Device (device malfunction,
                                                                    e.g. error in checksum)

    erd                      (4)                      -- extended reply delay enabled
    frc                      (5)                      -- forced device (port forced
                                                                    in any Traffic_Store of this
                                                                    device)

    dnr                      (6)                      -- Device Not Ready (e.g. not
                                                                    initialised)
  }
}

```

```

    ser          (7)          -- Reserved for Service (this
                                Station is currently
                                reserved by a Manager).
  }
}
```

NOTE The MVB\_Status is defined in 8.4.3, which is the reference for the definition of the terms in case of discrepancy.

13.3.1.2 Write\_MVB\_Control

13.3.1.2.1 Description

This service sets up the Bus\_Administrator parameters, and in particular:

- a) switches a Bus\_Administrator on and off,
- b) lets it switch to the new Periodic\_List;
- c) causes it to pass mastership.

In devices lacking the Bus\_Administrator capability, this service is used for selective reception only.

13.3.1.2.2 Call\_Write\_Mvb\_Control

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 11							
bus_id								reserved1							
device_address															
reserved2								t_ignore							
reserved3								aon	aof	spl	tms	sla	slb	cla	clb

```

Call_Write_Mvb_Control ::= RECORD
{
  bus_id          UNSIGNED8 (0..15),  -- identifier of the link
  reserved1       WORD8 (=0),         -- reserved
  device_address  WORD16,             -- Device_Address to be set.
                                         or 0 if not used.
  reserved2       WORD8 (=0)          -- reserved, =0
  t_ignore        UNSIGNED8,          -- value of the time-out
                                         T_ignore in multiples of 1,0
                                         µs;
                                         default (t_ignore = 0):
                                         42,7 µs;
                                         recommended values:
                                         22,0 µs, 64,0 µs, 86,0 µs,
                                         maximum value:255,0 µs.
  reserved3       WORD8 (=0),         -- reserved
  command         BITSET8,            -- command to be performed
  {
    aon           (0)                -- (Bus_Administrator only)
                                         enables Bus_Administrator
    aof           (1)                -- (Bus_Administrator only)
                                         disables Bus_Administrator
                                         aon = aof = 1: no action
    spl           (2)                -- (Bus_Administrator only)
                                         switch to new Periodic_List
                                         when Bus_Administrator
                                         becomes master
  }
}
```

```

tms          (3)          -- (Bus_Administrator only)
                           transfer mastership to the
                           next Bus_Administrator in
                           the Bus_Administrators_List
                           at the end of the macro-
                           cycle following the request,
                           retain mastership if other
                           Bus_Administrator do not
                           accept it.

sla          (4)          -- (sla = slb = 0: no action)
                           trust Line_A, even if
                           disturbed;

slb          (5)          -- (sla = slb = 1: normal
                           redundancy)
                           trust Line_B, even if
                           disturbed

cla          (6)          -- (cla = clb = 0: no action)
                           reset frame error counter of
                           Line_A.

clb          (7)          -- (cla = clb = 1: reset both
                           counters)
                           reset frame error counter of
                           Line_B;

    }
}

```

NOTE The MVB\_Control is defined in 10.4.4, which is the reference in case of discrepancy.

### 13.3.1.2.3 Reply\_Write\_Mvb\_Control

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 11							

```

Reply_Write_Mvb_Control ::= RECORD
{
    -- no parameters
}

```

### 13.3.1.3 Read\_MVB\_Devices

#### 13.3.1.3.1 Description

Retrieves the list of devices which the Bus\_Administrator found on the bus, together with their Device\_Status.

#### 13.3.1.3.2 Call\_Read\_Mvb\_Devices

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 12							
bus_id								reserved1							

```

Call_Read_Mvb_Devices ::= RECORD
{
    bus_id          UNSIGNED8 (0..15),  -- identifier of the link
    reserved1       WORD8 (=0)         -- reserved
}

```

**13.3.1.3.3 Reply\_Read\_Mvb\_Devices**

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 12							
bus_id								reserved1, =0							
device_address															
nr_devices															
devices_list: ARRAY [nr_devices] OF															
device_address (n)															
device_status (n)															

Reply\_Read\_Mvb\_Devices ::= RECORD

```

{
  bus_id            UNSIGNED8 (0..15),  -- identifier of the link
  reserved1         WORD8 (=0),         -- reserved
  device_address    WORD16,              -- Device_Address device on
                                         this bus
  nr_devices        UNSIGNED16,          -- number of devices in this
                                         list
  devices_list      ARRAY [nr_devices] OF
  {
    device_address  WORD16,              -- address of the device.
    device_status   Device_Status        -- copy of the Device Status of
                                         the device
  }
}
```

**13.3.1.4 Write\_MVB\_Administrator****13.3.1.4.1 Description**

To allow fast access to data structures, the Bus\_Administrator configuration list operates with offsets and indices. This applies especially to the four lists:

- the Known\_Devices\_List;
- the Periodic\_List;
- the Bus\_Administrators\_List;
- the Devices\_List.

For instance, the Periodic\_List is pointed to by an offset located at word 24. The size of the Periodic\_List is implicitly given by the difference between this offset (periodic\_list\_offset) and the next offset (bus\_administrators\_list\_offset) located at word 26.

The Periodic\_List is itself divided into

- the 11 Cycle Lists, which list for each Cycle the master frames (F\_code and address) in the order in which they should be sent out. All F\_codes are permitted. Each Cycle List is named according to its individual period, which is a  $2^n$  multiple of the basic period, with  $n = (1..10)$  as Cycle\_1, Cycle\_2, Cycle\_4,... up to Cycle\_1024;
- the five Split Lists, which define the actual split of the cycles into subcycles over the different periods and which contain the number of master frames to be sent in each basic period. A subcycle may not have more than 32 members. In practice, it is not possible to have more than 15 identifiers in a subcycle.

For efficiency, the Split List of two cycles are included into the same data structure.

EXAMPLE 1 Split\_8\_16 contains in the first octet, the Split List of Cycle\_8 and, in the second, the Split List of Cycle\_16. Since the Split List of Cycle\_8 encompasses only 8 periods, while the Split List of Cycle\_16 encompasses 16 periods, the Split List of Cycle\_8 is repeated twice.

The offset of the Cycle List and of the Split List itself is not relative to the beginning of the message, but to the first word of the Periodic\_List, called “periodic\_list\_offset”.

The size of each list is given by the difference between its offset and the offset of the next list.

EXAMPLE 2 The size of Cycle\_1 is given by the difference: Cycle\_2\_Offset – Cycle\_1\_Offset, the size of Cycle\_32 is given by Cycle\_64\_Offset – Cycle\_32\_Offset.

An example of a MVB\_Administrator list appears in Table 15.

NOTE In case of discrepancy between the following specification and the MVB specification in 4.6, the latter holds.



**13.3.1.4.2 Call\_Write\_Mvb\_Administrator**

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 13							
bus_id								reserved1							
checksum0															
configuration_version															
t_reply_max															
macro_cycles															
event_poll_strategy															
basic_period															
macrocycles_per_turn															
device_scan_strategy															
reserved2															
reserved3															
reserved4															
reserved5															
known_devices_list_offset															
reserved_list_offset															
periodic_list_offset															
bus_administrators_list_offset															
device_scan_list_offset															
end_list_offset															
known_devices_list [nr_devices]															
device_address (n)															
(offset_P0) cycle_lists_offsets ARRAY [11] OF															
cycle_n_offset (counted from offset_P0)															
split_lists_offsets ARRAY [5] OF															
split_m_n_offset (counted from offset_P0)															
cycle_lists [11]															
cycle_n ARRAY [cycle_size = cycle_n+1_offset – cycle_n_offset] OF															
f_code (n)				address (n)											
split_2_4 ARRAY [4] OF															
cycle_2								cycle_4							
split_8_16 ARRAY [32] OF															
cycle_8								cycle_16							
split_32_64 ARRAY [64] OF															
cycle_32								cycle_64							
split_128_256 ARRAY [256] OF															
cycle_128								cycle_256							
split_512_1024 ARRAY [1024] OF															
cycle_512								cycle_1024							
bus_administrators_list [nr_admins]															
void				device_address (n)											

```

UpperLowerCount ::= RECORD
    {
        lower_count      UNSIGNED8
                                -- number of frames of the
                                -- shorter cycle to be put into
                                -- each basic period
                                -- (this list appears twice, it
                                -- is half as long as the
                                -- upper_count list).

        upper_count      UNSIGNED8,
                                -- number of frames of the
                                -- longer cycle to be put into
                                -- each basic period
    }

Call_Write_Mvb_Administrator ::= RECORD
    {
        bus_id            UNSIGNED8 (0..15),
                                -- identifier of the link
        reserved1          WORD8,
                                -- reserved, =0
        checkword0         UNSIGNED16,
                                -- checksum over this structure
                                -- (user dependent)
                                -- checkword0 is located at
                                -- offset 0.

        actualisation_key  UNSIGNED16,
                                -- version number of the
                                -- configuration, as a 16-bit
                                -- number. The most significant
                                -- bit set to 1 indicates to
                                -- transmit the least
                                -- significant 15 bits in
                                -- Mastership_Transfer_Response
                                -- .
                                -- The two least significant
                                -- bits are copied to AX0 and
                                -- AX1.

        t_reply_max        UNSIGNED16,
                                -- maximum value of the
                                -- expected reply delay for any
                                -- device on the bus, in
                                -- multiples of 1,0 µs;
                                -- default (t_reply_max = 0) =
                                -- 42,7 µs,
                                -- maximum = 256 µs).

        macro_cycles       UNSIGNED16,
                                -- duration of a macrocycle
                                -- after which the master may
                                -- release control if
                                -- requested, in number of
                                -- basic periods.

        event_poll_strategy ENUM16
            {
                NOEVENTS      ('0000'H),
                                -- no event polling
                HIPRONLY       ('4000'H),
                                -- high priority events only
                LOPRONLY       ('8000'H),
                                -- low priority events only
                HILOPRIO       ('C000'H),
                                -- high and low priority
                                -- events.
            },

        basic_period        UNSIGNED16,
                                -- basic period in
                                -- microseconds;
                                -- allowed values: 0000 (no
                                -- poll), 1000 µs.
    }

```

```

macrocycles_per_turn  UNSIGNED16,      -- number of macrocycles during
                                         which the master retains
                                         control (a multiple of 1024
                                         ms).

device_scan_strategy  ENUM16,
{
    SCANKNOWNONLY  (0), --          scan all known devices
    SCANALL        (1) --          scan all device addresses
},
reserved2            WORD16 (=0),
reserved3            WORD16 (=0),
reserved4            WORD16 (=0),
reserved5            WORD16 (=0),
known_device_list_offset  UNSIGNED16,  -- offset (from checkword0)
                                         where the first address of
                                         the device list is located;
                                         (The list size is the
                                         difference between this
                                         offset and the
                                         reserved_list_offset.)

reserved_list_offset  UNSIGNED16,      -- equal to
                                         "periodic_list_offset" (list
                                         size = 0)

periodic_list_offset  UNSIGNED16,      -- offset (from checkword0)
                                         where the first F_code +
                                         address of the first cycle
                                         is located;
                                         (list size is the difference
                                         between this offset and
                                         bus_administrators_list_offs
                                         et.)

bus_administrators_list_offset  UNSIGNED16,  -- offset (from
                                         checkword0) where the first
                                         address of the
                                         Bus_Administrators_List is
                                         located.

device_scan_list_offset  UNSIGNED16,      -- offset (from checkword0),
                                         where the device scan
                                         allocation list is located
                                         (unused)

end_list_offset        UNSIGNED16,      -- dummy offset indicating size
                                         of ScanList and pointing to
                                         the next location after the
                                         last one.

known_devices_list     ARRAY [nr_devices] OF -- list of the known
                                         devices
{
    device_address      WORD16          -- (nr_devices =
                                         known_devices_list_offset -
                                         reserved_list_offset)
},
cycle_lists_offsets    RECORD          -- Device_Address of the device
{
    cycle_1_offset      UNSIGNED16      -- the periodic list begins
                                         here
    cycle_2_offset      UNSIGNED16      -- comprising for each one of
                                         the
    cycle_4_offset      UNSIGNED16      -- possible 11 cycles: offset
                                         (counted from
                                         cycle_lists_indices)
    cycle_8_offset      UNSIGNED16      -- where the first address of
                                         the cycle
    cycle_16_offset     UNSIGNED16      -- is located
}

```

```

    cycle_32_offset    UNSIGNED16
    cycle_64_offset    UNSIGNED16
    cycle_128_offset   UNSIGNED16
    cycle_256_offset   UNSIGNED16
    cycle_512_offset   UNSIGNED16
    cycle_1024_offset  UNSIGNED16
  }
split_list_offsets    RECORD
  {
    split_2_4_offset   UNSIGNED16
    split_8_16_offset  UNSIGNED16
    split_32_64_offset UNSIGNED16
    split_128_256_offset UNSIGNED16
    split_512_1024_offset UNSIGNED16
  }
cycle_lists           RECORD
  {
    -- eleven lists, one for each
    -- cycle
    -- cycle_size = offset between
    -- this list and the next

    cycle_1            ARRAY[cycle_size] OF FcodeAdr
    cycle_2            ARRAY[cycle_size] OF FcodeAdr
    cycle_4            ARRAY[cycle_size] OF FcodeAdr
    cycle_8            ARRAY[cycle_size] OF FcodeAdr
    cycle_16           ARRAY[cycle_size] OF FcodeAdr
    cycle_32           ARRAY[cycle_size] OF FcodeAdr
    cycle_64           ARRAY[cycle_size] OF FcodeAdr
    cycle_128          ARRAY[cycle_size] OF FcodeAdr
    cycle_256          ARRAY[cycle_size] OF FcodeAdr
    cycle_512          ARRAY[cycle_size] OF FcodeAdr
    cycle_1024         ARRAY[cycle_size] OF FcodeAdr
  }
split_lists           RECORD
  {
    split_2_4          ARRAY [ 4 ] OF UpperLowerCount    -- see this Type
                                                            definition.
    split_8_16         ARRAY [16] OF UpperLowerCount
    split_32_64        ARRAY [64] OF UpperLowerCount
    split_128_256      ARRAY [256] OF UpperLowerCount
    split_512_1024     ARRAY [1024] OF UpperLowerCount
  }
bus_administrators_list ARRAY [nr_admins] OF -- list of the known
  {
    -- Bus_Administrators, with:
    -- (nr_admins =
    --   device_scan_list_offset -
    --   bus_administrators_list_offs
    --   et)

    f_code             UNSIGNED4
    device_address      UNSIGNED12
  }
}

```

**Table 15 – Example of mvb\_administrator\_list**

Offset	Value	Name	Contents
000	D4EC		checksum (unused)
002	0001		actualisation_key (the two LSBs form the Actualisation_Index)
004	07D0		T_reply_max = 2000,0 µs
006	0000		macro_cycle (unused, means 1024 ms)
008	C000		event poll strategy = high and low priority events
00A	03E8		basic_period = 1 ms
00C	1000		turn duration (4096 × 1 ms)
00E	0001		device scan strategy: 1= all devices
010	0000		reserved
012	0000		reserved
014	0000		reserved
016	0000		reserved
018	0024		offset to Known_Devices_List ('024'H = 36)
01A	0036		unused (Known_Devices_List size = '036'H – '024'H = 9 words)
01C	0036		offset to Periodic_List (size = 'B36'H – '036'H = 1664)
01E	0B36		offset to Bus_Administrators_List
020	0040		offset to Unused_List (void)
022	0040		offset to End
024	0020	Known_Devices_List	device '002'H (first of a list of 9 devices)
026	0022		device '004'H bus 0
028	0024		device '024'H bus 0
02A	0028		device '028'H bus 0
02C	002C		device '02C'H bus 0
02E	0034		device '034'H bus 0
030	0036		device '036'H bus 0
032	00C8		device '0C8'H bus 0
034	00DE		device '0DE'H (last device)
036	0020	Periodic_List	Cycle_1_offset (location = '36'H + '20'H = '56'H, size = 0)
038	0020		Cycle_2_offset (location = '056'H, size = 0)
03A	0020		Cycle_4_offset (location = '056'H, size = 0)
03C	0020		Cycle_8_offset (location = '056'H, size = 0)
03E	0020		Cycle_16_offset (location = '056'H, size = '024'H)
040	0044		Cycle_32_offset (location = '7A'H, size = '0A'H)
042	004E		Cycle_64_offset
044	0052		Cycle_128_offset
046	0056		Cycle_256_offset
048	0056		Cycle_512_offset
04A	0058	Split_List_indices:	Cycle_1024_offset
04C	0058		Split_2_4_offset (location = '36'H + '58'H = '8E'H)
04E	0060		Split_8_16_offset (location = '36'H + '60'H = '96'H)
050	0080		Split_32_64_offset (location = '36'H + '80'H = 'B6'H)
052	0100		Split_128_256_offset (location = '36'H + '100'H = '136'H)
054	0300		Split_512_1024_offset

**Table 15** (continued)

Offset	Value	Name	Contents
056	00C8	Cycle_16	1st F_code + address ('56'H = '36'H + '20'H)
058	00C9	(Cycle_1, Cycle_2,	2nd F_code + address
05A	03E8	Cycle_4, Cycle_8,	3rd F_code + address
05C	03EA	Cycle_1024 are void)	4th F_code + address
05E	0450		5th F_code + address
060	0452		6th F_code + address
062	0454		7th F_code + address
064	0456		8th F_code + address
066	0458		9th F_code + address
068	045A		10th F_code + address
06A	045C		11th F_code + address
06C	045E		12th F_code + address
06E	1464		13th F_code + address
070	1466		14th F_code + address
072	1468		15th F_code + address
074	146A		16th F_code + address
076	3460		17th F_code + address
078	3462		18th F_code + address (and last)
07A	015F	Cycle_32	1st F_code + address ('56'H = '36'H + '44'H)
07C	01F1		2nd F_code + address
07E	0321		3rd F_code + address
080	415E		4th F_code + address
082	4320		5th F_code + address (and last)
084	019D	Cycle_64	1st F_code + address ('84'H = '36'H + '4E'H)
086	11C2		2nd F_code + address (and last)
088	40F9	Cycle_128	1st F_code + address of Cycle_128
08A	418F		2nd F_code + address of Cycle_128 (and last)
08C	20FE	Cycle_512	1st F_code + address of Cycle_512 (and only)
08E	0000	Split_2_4	unused (void cycles)
090	0000	(4 words)	unused (void cycles)
092	0000		unused (void cycles)
094	0000		unused (void cycles)
096	0600	Split_8_16	1st period: 6 of Cycle_16, 0 of Cycle_8 (void)
098	0600	(16 words)	2nd period: 6 of Cycle_16, 0 of Cycle_8 (void)
09A	0400		3rd period: 4 of Cycle_16, 0 of Cycle_8 (void)
09C	0200		4th period: 6 of Cycle_16, 0 of Cycle_8 (void)
...	...		....
0B4	0000		16th period: 0 of Cycle_16, 0 of Cycle_8 (void)
0B6	0000	Split_32_64	1st period: 0
0B8	0000	(64 words)	2nd period: 0
0BA	0001		3rd period: 1 of Cycle_32
0BC	0002		4th period: 2 of Cycle_32
0BE	0201		5th period: 2 of Cycle_64, 1 of Cycle_32
0C0	0001		6th period: 1 of Cycle_32
0C2	0000		7th period: 0 of Cycle_64, 0 of Cycle_32
...	...		...

**Table 15** (continued)

Offset	Value	Name	Contents
134	0000	Split_128_256 (256 words)	64th period: 0
136	0000		1st period: 0
...	...		2nd period
...	...	Split_512_1024 (1024 words)	...
334	0000		256th period:
336	0000		1st period
338	0000	Bus_Admistrators_List	2nd period
...	...		...
340	0001		6th period: 1 of Cycle_512
...	...	(Unused List, End)	...
B34	0000		1024th period
B36	0002		first device in Bus_Admistrators_List
B38	0004	(Unused List, End)	2nd device in Bus_Admistrators_List
(B3A)			dummy

**13.3.1.4.3 Reply\_Write\_Mvb\_Administrator**

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 13							

```
Reply_Write_Mvb_Administrator ::= RECORD
  {}
```

```
-- no parameters
```

## Bibliography

IEC 60332-1-2, *Test for vertical flame propagation for a single insulated wire or cable – Procedure for 1kW pre-mixed flame*

IEC 60870 (all parts), *Telecontrol equipment and systems*

IEC 60874-10-1, *Connectors for optical fibres and cables – Part 10-1: Detail specification for fibre optic connector type BFOC/2,5 terminated to multimode fibre type A1 (withdrawn)*

IEC 60874-10-2, *Connectors for optical fibre and cables – Part 10-2: Detail specification for fibre optic connector BFOC/2,5 terminated to single-mode fibre type B1 (withdrawn)*

IEC 60874-10-3, *Connectors for optical fibre and cables – Part 10-3: Detail specification for fibre optic connector BFOC/2,5 terminated to single and multimode fibre (withdrawn)*

IEC 61158-2, *Digital data communications for measurement and control – Fieldbus standard for use in industrial control systems – Part 2: Physical layer specification and service definition*

ISO/IEC 7498-1, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO/IEC 7498-4, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework*

ISO/IEC 8073, *Information technology – Open Systems Interconnection – Protocol for providing the connection-mode transport service*

ISO/IEC 8473 (all parts), *Information technology – Protocol for providing the connectionless-mode network service*

ISO/IEC 8802-2, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 2: Logical link control*

ISO/IEC 8802-3, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*

ISO/IEC 9072 (all parts), *Information processing systems – Text communication – Remote Operations*

ISO/IEC 9506-1, *Industrial automation systems – Manufacturing Message Specification – Part 1: Service definition*

ISO/IEC 9506-2, *Industrial automation systems – Manufacturing Message Specification – Part 2: Protocol specification*

ISO/IEC 10040:1998, *Information technology – Open Systems Interconnection – Systems management overview*

ISO/IEC 10164 (all parts), *Information technology – Open Systems Interconnection – Systems Management*



ISO/IEC 10165 (all parts), *Information technology – Open Systems Interconnection – Management Information Services – Structure of management information*

ISO/IEC 13239:2002, *Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures*

UIC 558, *Remote control and data cable – Standard technical features for the equipping of RIC coaches*

UIC 568, *Loudspeaker and telephone systems in RIC coaches – Standard technical characteristics*

UIC 648, *Connections for electric cables and air pipes on headstocks of locomotives and driving trailers*

---

## SOMMAIRE

AVANT-PROPOS .....	134
INTRODUCTION .....	136
1 Domaine d'application .....	138
2 Références normatives .....	138
3 Termes, définitions, abréviations et conventions .....	139
3.1 Termes et définitions .....	139
3.2 Abréviations .....	156
3.3 Conventions .....	158
3.3.1 Base des valeurs numériques .....	158
3.3.2 Conventions d'appellation .....	159
3.3.3 Conventions pour les valeurs de temps .....	159
3.3.4 Conventions pour les interfaces de procédure .....	159
3.3.5 Spécification des données transmises .....	162
3.3.6 Conventions de diagrammes d'état .....	164
4 Couche Physique .....	165
4.1 Topologie .....	165
4.1.1 Segments .....	165
4.1.2 Coupleurs .....	165
4.1.3 Segments à ligne double .....	167
4.2 Classes de dispositifs .....	167
4.2.1 Capacités .....	167
4.2.2 Dispositifs de classe 0 .....	167
4.2.3 Dispositifs de classe 1 .....	167
4.2.4 Dispositifs de classe 2 .....	167
4.2.5 Dispositifs de classe 3 .....	168
4.2.6 Dispositifs de classe 4 .....	168
4.2.7 Dispositifs de classe 5 .....	168
4.2.8 Raccordement des dispositifs .....	168
4.3 Spécifications communes à tous les supports .....	168
4.3.1 Vitesse de signalisation .....	168
4.3.2 Délais de propagation .....	168
4.3.3 Interface émetteur-récepteur .....	169
4.3.4 Support redondant (option) .....	170
4.4 Support Électrique Courte Distance (choix) .....	171
4.4.1 Topologie ESD .....	171
4.4.2 ESD - Règles de configuration .....	172
4.4.3 ESD – spécifications des sections .....	173
4.4.4 ESD – blindage .....	174
4.4.5 ESD - interface dépendante du support .....	174
4.4.6 ESD – Spécifications Line_Unit .....	177
4.4.7 ESD – forme d'onde du signal .....	178
4.4.8 ESD – émetteur .....	179
4.4.9 ESD – récepteur .....	180
4.5 Support Electrique Moyenne Distance (choix) .....	181
4.5.1 Topologie EMD .....	181
4.5.2 EMD – règles de configuration .....	181

4.5.3	EMD – terminaison .....	182
4.5.4	Section de câble .....	182
4.5.5	EMD – blindage .....	183
4.5.6	EMD - Interface dépendant du support .....	184
4.5.7	EMD – Spécifications de Line_Unit.....	189
4.5.8	EMD – forme d’onde du signal .....	189
4.5.9	EMD – spécifications de l’émetteur .....	190
4.5.10	EMD – spécifications du récepteur .....	194
4.6	Support à fibres optiques (choix).....	196
4.6.1	OGF – topologie .....	197
4.6.2	OGF – câble et fibre optique .....	197
4.6.3	OGF - interface dépendante du support.....	198
4.6.4	OGF – signal d’essai (indications) .....	198
4.6.5	OGF – spécifications de l’émetteur.....	199
4.6.6	OGF – spécifications du récepteur .....	200
4.6.7	OGF – coupleur en étoile actif.....	201
4.6.8	OGF - configuration à ligne double (option) .....	201
5	Signalisation dépendant du support.....	202
5.1	Codage et décodage des trames.....	202
5.1.1	Conventions.....	202
5.1.2	Codage des bits.....	202
5.1.3	Symboles de non-données .....	203
5.1.4	Bit de Début.....	203
5.1.5	Délimiteur de Début de Trame .....	203
5.1.6	Délimiteur de Fin de Trame.....	204
5.1.7	Trame valide (définition) .....	205
5.1.8	Détection de ligne au repos.....	206
5.1.9	Détection de collision.....	206
5.1.10	Comportement du récepteur en cas d'erreur.....	206
5.1.11	Détection de bavardage .....	206
5.2	Redondance de ligne (option) .....	206
5.2.1	Principe .....	206
5.2.2	Transmission redondante.....	207
5.2.3	Réception redondante.....	207
5.2.4	Commutation .....	207
5.2.5	Rapport d’état de redondance .....	208
5.3	Répéteur .....	208
5.3.1	Répéteur placé entre segments à ligne simple .....	208
5.3.2	Répéteur pour support redondant (option) .....	209
6	Trames et télégrammes .....	210
6.1	Format des trames .....	210
6.1.1	Format de Trame-Maître .....	210
6.1.2	Format de Trame-Esclave.....	211
6.1.3	Code Détecteur d’Erreur .....	211
6.2	Cadence des télégrammes .....	212
6.2.1	Conventions.....	212
6.2.2	Délai de réponse (définition) .....	213
6.2.3	Espacement des trames à la source.....	215
6.2.4	Espacement des trames côté destinataire .....	216

6.2.5	Espacement des trames au niveau du maître .....	217
6.3	Détection de trames correctes, collision et silence par le maître .....	217
6.3.1	Trame correcte (définition).....	217
6.3.2	Détection de collision par le maître .....	218
6.3.3	Détection de silence par le maître .....	218
7	Contrôle de la Couche de Liaison.....	218
7.1	Adressage .....	218
7.1.1	Adresse de Dispositif .....	218
7.1.2	Logical_Address .....	218
7.1.3	Group_Address.....	219
7.2	Contenu des Trames-maîtres.....	219
7.2.1	Format de Trame-Maître .....	219
7.2.2	Codage du F_code.....	219
7.3	Contenu des Trames-esclaves.....	220
7.3.1	Format de Trame-Esclave.....	220
7.3.2	Erreur de taille .....	221
7.4	Types de télégrammes .....	221
7.4.1	Télégramme de Données de Processus .....	221
7.4.2	Données de Messagerie.....	222
7.4.3	Télégrammes de Données de supervision .....	223
8	Attribution du support.....	224
8.1	Organisation .....	224
8.1.1	Tour .....	224
8.1.2	Période de Base .....	224
8.1.3	Remplissage.....	226
8.2	Scrutin Périodique .....	226
8.2.1	Liste Périodique .....	226
8.2.2	Période Individuelle.....	226
8.2.3	Construction de la Phase Périodique.....	227
8.3	Scrutin d'Evénements .....	228
8.3.1	Group_Address.....	228
8.3.2	Event_Round .....	229
8.3.3	Algorithme de recherche d'événements recommandé .....	231
8.3.4	Trames de Données de Supervision pour Event_Arbitration.....	232
8.4	Devices_Scan.....	234
8.4.1	Device_Status .....	234
8.4.2	Protocole du Device_Status .....	237
8.4.3	Protocole du Devices_Scan .....	237
9	Transfert de maîtrise .....	237
9.1	Opération de transfert de maîtrise.....	238
9.1.1	Configuration de l'administrateur de bus.....	238
9.2	Spécifications relatives au transfert de maîtrise .....	238
9.2.1	Etats.....	238
9.2.2	Temporisations pour le transfert de maîtrise.....	241
9.3	Trames de données de supervision pour le transfert de maîtrise .....	242
9.3.1	Télégramme de Device_Status .....	242
10	Interface de Couche de Liaison .....	243
10.1	Organisation de la Couche de Liaison .....	243
10.2	Interface de Liaison de Données de Processus .....	244

10.3	Interface de Liaison de Données de Messagerie .....	245
10.3.1	Généralités .....	245
10.3.2	Priorité .....	245
10.3.3	Taille des paquets .....	245
10.3.4	Protocol_Type .....	245
10.3.5	Protocole de Transport de Messages .....	245
10.4	Interface de Supervision de Liaison .....	246
10.4.1	Généralités .....	246
10.4.2	Procédures de l'Interface de Supervision de Liaison .....	246
10.4.3	MVB_Status .....	246
10.4.4	MVB_Control .....	247
10.4.5	MVB_Devices .....	248
10.4.6	MVB_Administrator .....	248
10.4.7	MVB_Report .....	250
11	Protocoles en Temps Réel .....	251
12	Fonction Passerelle .....	251
13	Gestion de Réseau .....	251
13.1	Teneur du présent article .....	251
13.2	Objets gérés par le MVB .....	252
13.2.1	Objets de liaison MVB .....	252
13.3	Services MVB et messages de gestion .....	253
13.3.1	Services de liaison du MVB .....	253
	Bibliographie .....	267
	Figure 1 – Dispositif de référence et structure du document .....	137
	Figure 2 – Exemple de transition d'état .....	164
	Figure 3 – Configuration du MVB .....	166
	Figure 4 – Interface émetteur-récepteur .....	170
	Figure 5 – Exemple de segment ESD .....	171
	Figure 6 – Exemple de terminaison .....	173
	Figure 7 – Section ESD de fond de panier (ligne double) .....	175
	Figure 8 – Agencement du connecteur ESD .....	176
	Figure 9 – Agencement du connecteur de terminaison ESD .....	177
	Figure 10 – Exemple de début de trame (ESD) .....	178
	Figure 11 – Fin d'une trame ESD (les deux cas) .....	180
	Figure 12 – Support EMD .....	181
	Figure 13 – Blindage (segment à ligne simple) .....	184
	Figure 14 – Raccordement du dispositif à ligne simple .....	185
	Figure 15 – Raccordement du dispositif à ligne double à l'EMD .....	186
	Figure 16 – Disposition des connecteurs EMD .....	187
	Figure 17 – Rebouclage de la terminaison EMD .....	188
	Figure 18 – Exemple de début de trame EMD .....	190
	Figure 19 – Exemple de forme d'onde d'une impulsion au niveau de l'émetteur EMD .....	192
	Figure 20 – Exemple de fin de trame EMD .....	194
	Figure 21 – Signal d'essai du récepteur EMD .....	195
	Figure 22 – Liaison optique .....	197

Figure 23 – Connecteur optique (dimensions en millimètres) .....	198
Figure 24 – Exemple de début de trame OGF .....	199
Figure 25 – Gigue frontale.....	200
Figure 26 – Exemple de coupleur en étoile actif.....	201
Figure 27 – Exemple de coupleur en étoile dupliqué .....	202
Figure 28 – Codage des données "0" et "1" .....	203
Figure 29 – Codage des symboles Non_Data .....	203
Figure 30 – Délimiteur de Début de Trame-Maître .....	204
Figure 31 – Délimiteur de Début de Trame-Esclave .....	204
Figure 32 – Exemple de Délimiteur de Fin de Trame dans le cas du support EMD.....	205
Figure 33 – Exemple de trame valide (support OGF) .....	205
Figure 34 – Décalage du signal .....	207
Figure 35 – Exemple de répéteur pour un raccordement en ligne simple .....	209
Figure 36 – Exemple de répéteur connectant un segment à ligne double à un segment à ligne simple.....	210
Figure 37 – Format de Trame-Maître .....	211
Figure 38 – Trames-esclaves .....	211
Figure 39 – Cadence d'un télégramme .....	213
Figure 40 – Exemple de délai de réponse .....	214
Figure 41 – Espacement des trames du côté source.....	215
Figure 42 – Espacement des trames côté destinataire(s).....	216
Figure 43 – Espacement des trames du côté du maître.....	217
Figure 44 – Contenu d'une Trame-Maître .....	219
Figure 45 – Ordre des mots dans une Trame-Esclave.....	221
Figure 46 – Télégramme de Données de Processus .....	221
Figure 47 – Télégramme de Données de Messagerie .....	223
Figure 48 – Télégramme de Données de Supervision .....	224
Figure 49 – Périodes de Base .....	225
Figure 50 – Exemple de construction du Macro_Cycle .....	228
Figure 51 – Format de trame de General_Event_Request.....	232
Figure 52 – Trame de Group_Event_Request (M = 6, C = ABCDEF) .....	233
Figure 53 – Trame de Single_Event_Request.....	233
Figure 54 – Trame de la Event_Identifier_Response .....	233
Figure 55 – Device_Status_Request.....	234
Figure 56 – Device_Status_Response .....	234
Figure 57 – Device_Status d'un dispositif de Classe 1 .....	235
Figure 58 – Device_Status d'un dispositif de Classes 2/3/4/5.....	235
Figure 59 – Device_Status d'un dispositif ayant la capacité Administrateur de Bus.....	236
Figure 60 – Device_Status d'un dispositif ayant la capacité Passerelle.....	236
Figure 61 – Etats de transfert de maîtrise .....	241
Figure 62 – Device_Status_Request (envoyée par le maître actuel) .....	242
Figure 63 – Device_Status_Response (envoyée par le maître proposé) .....	242
Figure 64 – Mastership_Transfer_Request (envoyée par le maître actuel) .....	242
Figure 65 – Mastership_Transfer_Response (envoyée par le maître suivant proposé).....	242

Figure 66 – Organisation de la Couche de Liaison .....	244
Tableau 1 – Modèle pour la spécification d'une procédure d'interface .....	161
Tableau 2 – Exemple de structure de message.....	162
Tableau 3 – Exemple de forme de message textuel (correspondant au Tableau 2).....	163
Tableau 4 – Tableau de transitions d'état .....	165
Tableau 5 – Capacités des dispositifs MVB .....	167
Tableau 6 – Affectation des broches du connecteur ESD .....	177
Tableau 7 – Affectation des broches du connecteur EMD .....	188
Tableau 8 – Types de Trames Maître et F_code .....	220
Tableau 9 – Codage LS_RESULT.....	246
Tableau 10 – Objet MVB_Status.....	247
Tableau 11 – Objet MVB_Control .....	248
Tableau 12 – Objet MVB_Devices .....	248
Tableau 13 – Objet MVB_Administrator .....	249
Tableau 14 – Codage du LS_V_REPORT .....	251
Tableau 15 – Exemple de mvb_administrator_list .....	264

## COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

### MATÉRIEL ÉLECTRONIQUE FERROVIAIRE – RÉSEAU EMBARQUÉ DE TRAIN (TCN) –

#### Partie 3-1: Bus de Véhicule Multifonctions (MVB)

#### AVANT-PROPOS

- 1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de brevet. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale CEI 61375-3-1 a été établie par le comité d'études 9 de la CEI: Matériels et systèmes électriques ferroviaires.

Cette première édition annule et remplace les articles de la deuxième édition de la CEI 61375-1 publiée en 2007 applicables à la spécification du MVB, dont elle constitue une révision technique.

La présente norme a été élaborée en tenant compte de la troisième édition de la CEI 61375-1.



Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
9/1644/FDIS	9/1668/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Une liste de toutes les parties de la série CEI 61375, présentées sous le titre général *Matériel électronique ferroviaire – Réseau embarqué de train (TCN)*, peut être consultée sur le site web de la CEI.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous "<http://webstore.iec.ch>" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

## INTRODUCTION

La présente partie de la CEI 61375 spécifie un composant du Réseau Embarqué de Train, le Bus de Véhicule Multifonctions (MVB), qui est un bus série de transmission de données conçu en premier lieu, mais pas exclusivement, pour l'interconnexion d'équipements dont l'interopérabilité et l'interchangeabilité sont nécessaires.

La présente partie spécifie:

- a) le support physique en configurations de lignes simples et de lignes doubles;
- b) la signalisation et le traitement de la redondance;
- c) le format et la cadence des trames et des télégrammes;
- d) l'organisation du trafic sur le bus;
- e) l'allocation de la maîtrise de bus;
- f) la gestion du bus;
- g) l'interface de Couche de Liaison et l'interface de gestion de cette couche.

La présente partie est structurée suivant le modèle de couches OSI d'un dispositif de référence MVB, tel que le montre la Figure 1:

### Article 4 Couche Physique

- Support électrique pour courtes distances (RS-485, 20,0 m)
- Support électrique pour moyennes distances (avec couplage par transformateur, 200,0, m)
- Support optique OGF pour longues distances (fibres de verre, 2,0 km)

### Article 5 Signalisation dépendant du support

- Codage et décodage des trames
- Interface de l'Unité de Ligne,
- Traitement de la redondance physique

### Article 6 Trames et télégrammes

- Codage de la Trame-Maître et de la Trame-Esclave, cadence du télégramme,

### Article 7 Contrôle de la Couche de Liaison

- Adressage
- Format de la Trame-Maître et de la Trame-Esclave

### Article 8 Attribution du support

- Scrutin Périodique
- Scrutin d'Événements
- Balayage des Dispositifs

### Article 9 Transfert de maîtrise

- Transfert de maîtrise régulier et exceptionnel

### Article 10 Interface de Couche de Liaison

- Interface de liaison de données de processus (LPI),
- Interface de liaison de données de messagerie (LMI),
- Interface de Supervision de Liaison (LSI).

### Article 11 Protocoles en Temps Réel

### Article 12 Fonction Passerelle

### Article 13 Gestion de réseau

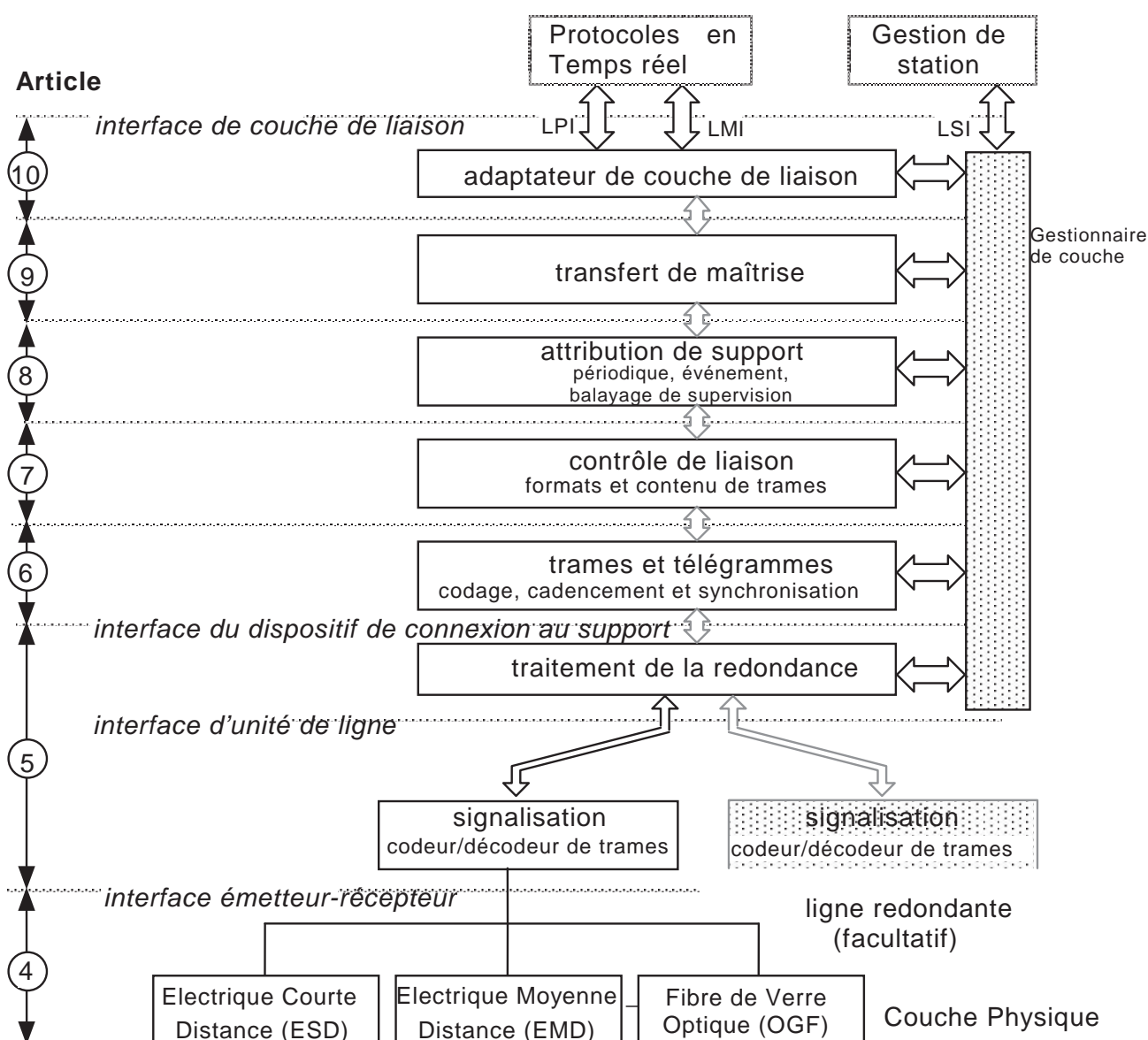


Figure 1 – Dispositif de référence et structure du document

# MATÉRIEL ÉLECTRONIQUE FERROVIAIRE – RÉSEAU EMBARQUÉ DE TRAIN (TCN) –

## Partie 3-1: Bus de Véhicule Multifonctions (MVB)

### 1 Domaine d'application

La présente partie de la CEI 61375 s'applique lorsque le MVB est exigé.

### 2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

CEI 60245-1, *Conducteurs et câbles isolés au caoutchouc – Tensions assignées au plus égales à 450/750 V – Partie 1: Exigences générales*

CEI 60304, *Couleurs de référence de l'enveloppe isolante pour câbles et fils pour basses fréquences*

CEI 60332-1-1, *Essais des câbles électriques et à fibres optiques soumis au feu – Partie 1-1: Essai de propagation verticale de la flamme sur conducteur ou câble isolé – Appareillage d'essai*

CEI 60571, *Équipements électroniques utilisés sur les véhicules ferroviaires*

CEI 60794-1-1, *Câbles à fibres optiques – Partie 1-1: Spécification générique – Généralités*

CEI 60807 (toutes les parties), *Connecteurs rectangulaires utilisés aux fréquences inférieures à 3 MHz*

CEI 60870-5-1, *Matériels et systèmes de téléconduite – Partie 5: Protocoles de transmission – Section Un: Formats de trames de transmission*

CEI 61375-2-1, *Matériel électronique ferroviaire – Réseau embarqué de train (TCN) – Partie 2-1: Bus de train filaire (WTB)*

ISO/CEI 8482, *Technologie de l'information – Télécommunications et échange d'informations entre systèmes – Interconnexions multipoints par paire torsadée*

ISO/CEI 8802-2, *Technologies de l'information – Télécommunications et échange d'informations entre systèmes – Réseaux locaux et métropolitains – Prescriptions spécifiques – Partie 2: Contrôle de liaison logique*

ISO/CEI 8824 (toutes les parties), *Technologie de l'information – Notation de syntaxe abstraite numéro un (ASN.1)*

ISO/CEI 8825 (toutes les parties), *Technologie de l'information – Règles de codage ASN.1*

ISO/CEI 9646 (toutes les parties), *Technologies de l'information – Interconnexion de systèmes ouverts - Cadre général et méthodologie des tests de conformité OSI*

ISO/CEI 13239, *Technologies de l'information – Télécommunications et échange d'information entre systèmes – Procédures de commande de liaison de données à haut niveau (HDLC)*

UIT-T Recommandation Z.100, *Langage de description et de spécification (SDL)*

### 3 Termes, définitions, abréviations et conventions

#### 3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

NOTE La première lettre de chaque mot-clé de la présente norme est écrite en majuscule et, lorsque celui-ci se compose de deux ou plusieurs mots, ces derniers sont reliés par un trait bas. Cette convention permet de retrouver les mots-clés dans les documents.

##### 3.1.1

##### **adresse**

identifie une entité de communication. Il en existe plusieurs types selon la couche considérée

##### 3.1.2

##### **agent**

processus d'application localisé dans une Station. L'agent accède aux objets gérés localement sous le contrôle du Gestionnaire

##### 3.1.3

##### **Données Apériodiques**

transmission de Données de Processus suite à une requête. Ce service n'est pas utilisé

##### 3.1.4

##### **Couche Application**

couche supérieure du modèle OSI en interface directe avec l'Application

##### 3.1.5

##### **Interface de Couche Application**

définition des services offerts par la Couche Application

##### 3.1.6

##### **Adaptateur de Messagerie d'Application**

code directement appelé par l'application réalisant les services de Messagerie

##### 3.1.7

##### **Interface de Messagerie d'Application**

définition des services de Messagerie

##### 3.1.8

##### **Processus d'Application**

entité de communication, réalisée par exemple par une tâche

##### 3.1.9

##### **Processeur d'Application**

processeur faisant tourner un Processus d'Application communiquant

### 3.1.10

#### **Interface de Supervision d'Application**

définition des services de Supervision disponibles en particulier pour l'Agent

### 3.1.11

#### **Adaptateur de Variables d'Application**

code appelé directement par l'application réalisant les services de Variables

### 3.1.12

#### **Interface de Variables d'Application**

définition des services de Variables

### 3.1.13

#### **arbitre**

dispositif ou protocole commun suivi par plusieurs dispositifs, qui sélectionne un ou plusieurs dispositifs en conflit pour la position du maître

### 3.1.14

#### **Canal Auxiliaire**

canal utilisé pour détecter des Nœuds supplémentaires

### 3.1.15

#### **Période de Base**

l'activité du bus est divisée en périodes. La plus courte période est la Période de Base, qui se compose d'une Phase Périodique (pour les Données Périodiques) et d'une Phase Apériodique (pour les Données de Messagerie et les Données de Supervision)

### 3.1.16

#### **big-endian (gros-boutiste)**

ordonnancement des données pour leur stockage et leur transmission dans lequel la part de poids fort d'une donnée représentée sur plusieurs octets est stockée à l'adresse la plus faible et transmise en premier

### 3.1.17

#### **bit-stuffing (bourrage de bits)**

méthode préconisée par l'ISO/CEI 13239 pour éviter le risque de confusion entre les Données de Trame et un Drapeau. Cette méthode consiste à insérer un "0" supplémentaire après chaque séquence de cinq symboles "1" consécutifs, et à supprimer ce "0" lors de la réception

### 3.1.18

#### **pont**

dispositif qui stocke et retransmet les trames d'un bus à l'autre en fonction de leurs adresses de Couche de Liaison

### 3.1.19

#### **diffusion**

transmission presque simultanée des mêmes informations vers plusieurs destinataires. La diffusion sur TCN n'est pas considérée comme fiable, c'est-à-dire que certains destinataires peuvent recevoir les informations et d'autres pas

### 3.1.20

#### **bus**

support de communication qui diffuse la même information quasiment au même instant à tous les équipements connectés, ce qui permet à tous les systèmes d'avoir une vue identique de son état, au moins dans le cas d'un arbitrage

**3.1.21****Administrateur de Bus**

équipement capable de devenir Maître du MVB

**3.1.22****Contrôleur de Bus**

processeur ou circuit intégré en charge de la Couche de Liaison de communication

**3.1.23****Commutateur (Contacteur) de Bus**

commutateur ou relais d'un Nœud de WTB qui connecte électriquement les sections de câble en provenance des deux directions

**3.1.24****Appelant (Caller)**

processus d'Application qui initie un échange de message

**3.1.25****code Détecteur d'Erreur**

méthode de détection d'erreurs reposant sur l'ajout aux données utiles transmises d'une somme de contrôle ou d'un contrôle de redondance cyclique (CRC) calculé sur les données utiles

**3.1.26****Validant/Variable de Contrôle**

variable de Processus de type booléen antivalent protégeant une autre Variable de Processus

**3.1.27****Check Offset**

offset (décalage) en bit d'une Variable de Contrôle dans un Dataset

**3.1.28****Rame Indéformable**

train composé d'un ensemble de véhicules, dont la composition n'est pas modifiée en exploitation normale, par exemple, métros, trains suburbains ou rames à grande vitesse

**3.1.29****composition**

numéro et caractéristiques des véhicules composant un train

**3.1.30****configuration**

définition de la topologie d'un bus, des équipements qui lui sont connectés, de leurs capacités et du trafic qu'ils génèrent. Par extension, opération consistant à charger dans les équipements les informations de configuration avant leur passage en mode opérationnel

**3.1.31****Confirmation de Connexion**

réponse du Consommateur à la Demande de Connexion émise par le Producteur

**3.1.32****Demande de Connexion**

premier paquet d'un message envoyé d'un Producteur vers un Consommateur

### **3.1.33**

#### **Rame**

véhicule unique ou groupe de véhicules qui ne sont pas séparés en exploitation normale. Une rame peut contenir aucun, un seul ou plusieurs réseaux de Rame.

### **3.1.34**

#### **réseau de Rame**

bus connectant des équipements dans une rame, le MVB par exemple, et qui se conforme ou s'adapte aux protocoles en Temps Réel TCN tel que décrit dans le présent document

### **3.1.35**

#### **cohérence**

Dataset composé de plusieurs éléments qui est considéré comme cohérent si tous les éléments sont lus ou écrits au cours d'une opération indivisible

### **3.1.36**

#### **Consommateur**

destinataire d'un message au niveau de la Couche Transport (voir: Producteur)

### **3.1.37**

#### **véhicule de continuité**

véhicule dépourvu de Nœud de Bus de Train en fonctionnement, mais assurant la continuité passive du Bus de Train entre ses véhicules adjacents

### **3.1.38**

#### **conversation**

échange de données au niveau de la Couche Application, consistant en un Message d'Appel et un Message de Réponse (ce dernier est inexistant dans le cas du protocole de distribution). Une conversation commence par la première trame Demande de Connexion et cesse lorsque le dernier acquittement pour le Message de Réponse a été reçu, ou lorsqu'il n'est plus attendu

### **3.1.39**

#### **datagramme**

trame contenant toutes les informations nécessaires à son acheminement vers son destinataire final, sans connaissance du contenu de la trame précédente. Les Datagrammes n'utilisent aucun mécanisme préliminaire d'établissement de connexion et ne sont pas acquittés au niveau de la Couche de Liaison

### **3.1.40**

#### **Dataset**

ensemble des Variables de Processus transmises dans une trame de Données de Processus

### **3.1.41**

#### **délimiteur**

séquence de signaux incluant des symboles de violation de code (ni " 1 " ni " 0 ") qui est utilisée pour délimiter le début (Délimiteur de Début de Trame) et la fin (Délimiteur de Fin de Trame), comme défini, par exemple, dans la CEI 61158-2

### **3.1.42**

#### **Dispositif Destinataire**

destinataire d'une trame au niveau de la Couche de Liaison (voir: Dispositif Emetteur)

### **3.1.43**

#### **dispositif**

module connecté à un ou plusieurs bus



**3.1.44****Adresse de Dispositif**

identifie un dispositif sur un bus; l'Adresse de Dispositif du MVB est codée sur 12 bits; l'Adresse de Dispositif du WTB est codée sur 8 bits, les 6 bits de plus faible niveau correspondant à l'Adresse de Nœud.

NOTE Un équipement connecté à plusieurs bus peut avoir une Adresse de Dispositif différente pour chaque bus. Des équipements particuliers comme les répéteurs participent uniquement à la Couche Physique et n'ont pas d'Adresse de Dispositif

**3.1.45****Etat du Dispositif**

mot de 16 bits exprimant les capacités et l'état d'une connexion de dispositif MVB

**3.1.46****Direction 1**

une direction d'un Nœud WTB

**3.1.47****Direction 2**

autre direction d'un Nœud WTB

**3.1.48****support Électrique Moyenne Distance**

un des supports du MVB

**3.1.49****support Électrique Courte Distance**

un des supports du MVB

**3.1.50****Délimiteur de Fin de Trame**

séquence qui termine une trame avant que le support ne retourne au repos

**3.1.51****Nœud d'Extrémité**

nœud qui réalise la terminaison des deux segments de bus auquel il est connecté mais qui n'établit pas la continuité entre eux

**3.1.52****Tour d'Événements**

séquence d'interrogation au cours de laquelle tous les événements en suspens au démarrage du tour sont traités

**3.1.53****boîte d'extension**

élément de connectique dans lequel le câble principal est interrompu et prolongé de manière passive par un câble d'extension pour la connexion d'un dispositif

**3.1.54****câble d'extension**

câble permettant l'insertion d'un Nœud dans un câble principal. Il est composé de deux paires torsadées par ligne, dont la surface de section est éventuellement plus petite que le câble principal lui-même

**3.1.55****dispositif de terrain**

équipement connectant des capteurs et actionneurs simples à un bus à l'extérieur d'un panier

### 3.1.56

#### **final**

destinataire d'un paquet (données ou acquittement) au niveau de la Couche Réseau. Lorsque deux équipements communiquent sur le même bus, l'équipement final se trouve dans le dispositif destinataire (voir: origine)

### 3.1.57

#### **Drapeau**

séquence de symboles "0" ou "1" qui sert à délimiter le début ou la fin d'une trame. Des drapeaux qui apparaîtraient dans les données transmises sont modifiés par un bourrage de bits, comme défini par exemple, dans l'ISO/CEI 13239

### 3.1.58

#### **trame**

séquence de symboles consécutifs envoyée en une fois par un émetteur, entre deux périodes d'inactivité de la ligne

### 3.1.59

#### **Code de Detection d'Erreurs**

somme de Contrôle (FCS – *Frame Check Sequence*) de 16 bits spécifiée dans l'ISO/CEI 13239

### 3.1.60

#### **Données de Trame**

données transmises entre le Délimiteur de Début de Trame et le Délimiteur de Fin de Trame (sur le MVB) ou entre le Préambule et le Délimiteur de Fin de Trame (sur le WTB)

### 3.1.61

#### **fritting (nettoyage des contacts)**

application d'une tension aux bornes d'un contact destiné à nettoyer son éventuelle oxydation

### 3.1.62

#### **Fonction**

processus d'Application qui échange des messages avec un autre Processus d'Application

### 3.1.63

#### **Répertoire (Table) de Fonctions**

répertoire qui associe un Indicatif de Fonction à un Indicatif de Station, et inversement

### 3.1.64

#### **Indicatif de Fonction**

mot de 8 bits identifiant une Fonction

### 3.1.65

#### **F code**

dans une Trame-Maître, indique le type de demande et la taille de la Trame Esclave attendue en réponse

### 3.1.66

#### **passerelle**

connexion de différents bus au niveau de la Couche Application requérant une analyse des données dépendant de l'application ainsi qu'une conversion de protocole

### 3.1.67

#### **Adresse de Groupe**

adresse d'un groupe de distribution auquel appartient un Nœud

**3.1.68****Répertoire de Groupes**

répertoire indiquant à un Nœud le groupe de distribution auquel il appartient

**3.1.69****distance de Hamming**

nombre minimal de bits d'une séquence de bits correcte donnée qui, si inversée, crée une séquence de bits erronée ne pouvant être distinguée d'une séquence correcte

**3.1.70****HDLC**

High-level Data Link Control (Commande de Liaison de Données à Haut Niveau), un ensemble de protocoles normalisés incluant l'ISO/CEI 13239 relatif à la transmission de données

**3.1.71****Données HDLC**

données transmises dans une trame HDLC

**3.1.72****Inauguration**

opération entreprise en cas de modification de la composition, qui donne à tous les Nœuds du WTB leur adresse par rapport au Maître, leur orientation et le descripteur de tous les Nœuds nommés présents sur le même bus

**3.1.73****Période Individuelle**

période séparant deux transmissions consécutives de la même Donnée de Processus par la même source. La Période Individuelle est un multiple d'une puissance de deux de la Période de Base

**3.1.74****instance**

- a) l'un parmi plusieurs objets qui partagent la même définition (instance d'objet)
- b) l'une parmi plusieurs exécutions (simultanées ou non) du même programme (instance de processus)

**3.1.75****intégrité**

capacité d'un système à reconnaître et à éliminer des données erronées en cas de dysfonctionnement d'un quelconque de ses sous-ensembles

**3.1.76****Nœud Intermédiaire**

nœud qui établit la continuité entre deux sections de bus qui lui sont connectées, mais qui n'assure pas leur terminaison

**3.1.77****câble de jonction**

câble connectant les câbles principaux de deux véhicules consécutifs, éventuellement de section supérieure à celle du câble principal, et qui est connecté à la main dans le cas du câble UIC. Il y a en général deux câbles de jonction entre les véhicules

**3.1.78****Ligne**

bus non redondant. Un bus redondant (dual-thread) met en œuvre deux lignes

### **3.1.79**

#### **Unité de Ligne**

ensemble des composants assurant la connexion électrique à la ligne

### **3.1.80**

#### **Adresse de Liaison**

adresse utilisée par la Couche de Liaison pour identifier le Bus et l'Adresse de Dispositif émetteur ou destinataire d'un paquet

### **3.1.81**

#### **Contrôle de Liaison**

champ de la trame HDLC qui identifie le type de trame

### **3.1.82**

#### **Données de Liaison**

données véhiculées par la Couche de Liaison mais non interprétées par elle

### **3.1.83**

#### **En-tête de Liaison**

partie d'une trame de Données de Messagerie interprétée par la Couche de Liaison

### **3.1.84**

#### **Couche de Liaison**

couche du modèle OSI permettant d'assurer des transferts de données en mode point à point ou en mode diffusion, entre différents équipements situés sur le même bus

### **3.1.85**

#### **Interface de Couche de Liaison**

interface entre la Couche de Liaison et les couches de communication supérieures

### **3.1.86**

#### **Gestion de Couche de Liaison**

interface contrôlant la Couche de Liaison pour sa gestion

### **3.1.87**

#### **little-endian (petit-boutiste)**

ordonnancement des données pour leur stockage et leur transmission dans lequel la part de poids faible d'une donnée représentée sur plusieurs octets est stockée à l'adresse la plus faible et transmise en premier

### **3.1.88**

#### **réseau local**

portion d'un réseau caractérisée par un mode d'accès et par un espace d'adressage uniques

### **3.1.89**

#### **contrôle de liaison logique**

protocoles et formats de trame associés qui servent à contrôler la Couche de Liaison

### **3.1.90**

#### **Adresse Logique**

adresse qui n'est pas liée à un équipement particulier (adresse de Données de Processus, par exemple)

### **3.1.91**

#### **Port Logique**

port d'un équipement utilisé pour le trafic de Données de Processus et adressé par l'Adresse Logique

**3.1.92****Macro Cycle**

nombre de Périodes de Base correspondant à une Macro Période

**3.1.93****Macro Période**

période Individuelle la plus longue mesurée en millisecondes. A l'issue de la Macro Période, le trafic périodique reprend selon le même enchaînement

**3.1.94****Canal Principal**

canal utilisé pour la transmission sur le bus principal

**3.1.95****Message de Gestion**

message échangé entre un Gestionnaire et un Agent dans le cadre de la Gestion de Réseau

**3.1.96****Gestionnaire**

fonction d'une Station dédiée à la Gestion de Réseau et qui envoie des Messages d'Appel de gestion en utilisant les Adresses Système

**3.1.97****triage**

attribution de noms ou d'adresses d'application aux Variables de Processus d'un dataset, qui, sur le WTB, dépend du type de Nœud et de sa Version

**3.1.98****Maître**

équipement qui envoie de manière spontanée des informations sur un bus à destination de plusieurs dispositifs esclaves. Il peut autoriser un Esclave à transmettre une Trame-Esclave uniquement dans un certain laps de temps

**3.1.99****Trame-Maître**

trame transmise par un Maître

**3.1.100****Délimiteur de Début de Trame-Maître**

Délimiteur de Début d'une Trame-Maître

**3.1.101****contrôle d'accès au support**

sous-couche de la Couche de Liaison qui contrôle l'accès au support (arbitrage, transfert de maîtrise, interrogation)

**3.1.102****interface dépendante du support**

interface mécanique et électrique entre le support de transmission et un dispositif de connexion au support

**3.1.103****support**

milieu de propagation du signal: câbles électriques, fibres optiques, etc.

#### **3.1.104**

##### **Dispositif de Connexion au Support**

dispositif dont la fonction est d'assurer la connexion d'un équipement au support

#### **3.1.105**

##### **message**

données transmises en un ou plusieurs paquets

#### **3.1.106**

##### **Messages (Messagerie)**

service de transmission du TCN

#### **3.1.107**

##### **Données de Messagerie**

données transmises de manière apériodique par la Couche de Liaison pour assurer la transmission de messages; le service Couche de Liaison correspondant

#### **3.1.108**

##### **messenger**

pile de communication assurant la transmission de messages de bout en bout, ainsi que l'interface avec l'application

#### **3.1.109**

##### **distribution (publipostage)**

transmission du même message à un groupe de Destinataires identifiés par leur Adresse de Groupe. Le terme « distribution » est utilisé même si le groupe comprend tous les Destinataires

#### **3.1.110**

##### **Bus de Véhicule Multifonctions**

##### **MVB (Multifunction Vehicle Bus)**

Bus de Véhicule à utiliser pour relier des stations programmables et de simples capteurs/actionneurs

#### **3.1.111**

##### **Train à Unités Multiples**

train composé d'un ensemble de rames indéformables, la composition de cet ensemble pouvant évoluer en exploitation normale

#### **3.1.112**

##### **réseau**

ensemble de différents systèmes possibles de communication qui échangent de l'information selon un moyen communément accepté

#### **3.1.113**

##### **Adresse Réseau**

adresse qui identifie une Fonction ou une Station sur le réseau. Il peut s'agir d'une Adresse Utilisateur ou d'une Adresse Système

#### **3.1.114**

##### **En-tête Réseau**

partie d'une trame de Données de Messagerie relevant de la Couche Réseau

#### **3.1.115**

##### **Couche Réseau**

couche du modèle OSI assurant le routage entre les différents bus

**3.1.116****Gestion de Réseau**

opérations nécessaires à la configuration, la supervision, le diagnostic et le maintenance à distance du réseau

**3.1.117****Nœud**

équipement sur le Bus de Train Filaire pouvant servir de passerelle entre le Bus de Train et le Bus de Véhicule

**3.1.118****Adresse de Nœud**

adresse d'un Nœud sur le Bus de Train (6 bits). Il s'agit des 6 bits de poids faible de l'Adresse de Dispositif codée sur 8 bits sur le WTB

**3.1.119****Descripteur de Nœud**

structure de données de 24 bits permettant de coder la Période de Nœud et la Clé de Nœud d'un Nœud

**3.1.120****Répertoire (Table) de Nœud**

répertoire de correspondance entre l'Adresse de Nœud et l'Adresse de Dispositif (correspondance biunivoque sur le WTB)

**3.1.121****Clé de Nœud**

indicatif de 16 bits fourni par l'application permettant d'identifier le type et la version d'un Nœud. Le Maître le distribue à tous les autres Nœuds après chaque changement de la composition et avant d'échanger des données

**3.1.122****Période de Nœud**

sur le WTB, Période Individuelle souhaitée pour un Nœud (identique à la Période Individuelle sauf en cas de surcharge du réseau)

**3.1.123****octet**

mot de 8 bits stocké dans la mémoire ou transmis sous forme d'unité <sup>1)</sup>

**3.1.124****Train à Composition Variable**

train composé d'un ensemble de véhicules, dont la configuration peut changer en exploitation normale, par exemple, les trains internationaux UIC

**3.1.125****Fibre de Verre Optique**

un des supports du MVB

**3.1.126****origine**

expéditeur d'un paquet (données ou acquittement) au niveau de la Couche Réseau. Lorsque deux équipements communiquent sur le même bus, l'Origine est située sur le dispositif émetteur (voir: final)

---

\* La CEI exige l'emploi, en anglais, du terme 'octet' et non du terme 'byte'.

### **3.1.127**

#### **paquet**

unité de message (information, acquittement ou commande) transmise dans exactement une trame de Données de Messagerie

### **3.1.128**

#### **période**

laps de temps après lequel le trafic sur le bus se répète selon le même motif

### **3.1.129**

#### **Données Périodiques**

Données de Processus transmises de manière périodique, à une fréquence correspondant à leur Période Individuelle

### **3.1.130**

#### **Liste Périodique**

liste des Nœuds, adresses ou équipements à interroger à chaque période de Macro Cycle

### **3.1.131**

#### **Phase Périodique**

phase durant laquelle le Maître gère le transfert des Données Périodiques selon le contenu de sa Liste Périodique

### **3.1.132**

#### **Adresse Physique**

Adresse de Dispositif sur le MVB et Adresse de Nœud sur le WTB qui identifient des équipements communiquant sur le même bus

### **3.1.133**

#### **Port Physique**

Port utilisé pour le transfert des Données de Messagerie ou les Données de Supervision, adressé par l'Adresse de Dispositif

### **3.1.134**

#### **écartement**

distance entre deux équipements adjacents sur le même bus électrique pour éviter les accumulations de charge sur ledit bus

### **3.1.135**

#### **interrogation**

envoi d'une Trame-Maître pour recevoir une Trame-Esclave

### **3.1.136**

#### **Port**

structure mémoire qui contient des données en émission ou en réception et dans laquelle une nouvelle valeur écrase la valeur précédente (registre par opposition à queue). Le bus et la ou les applications peuvent accéder simultanément à un Port

### **3.1.137**

#### **Table d'Index de Port**

table de conversion établissant la correspondance entre l'adresse mémoire d'un port et l'Adresse Logique des Données de Processus

### **3.1.138**

#### **Préambule**

séquence de signaux débutant une trame dont la fonction est de permettre la synchronisation du récepteur utilisé sur le WTB



**3.1.139****Couche Présentation**

couche du modèle OSI en charge de la représentation et de la conversion des données

**3.1.140****Données de Processus**

données identifiées par leur source diffusées de manière périodique par la couche de liaison en relation avec la transmission de Variables de Processus; le service Couche de Liaison correspondant

**3.1.141****Variable de Processus**

variable exprimant l'état d'un processus (vitesse, commande de frein, par exemple)

**3.1.142****Producteur**

émetteur d'un message au niveau de la Couche Transport (voir: Consommateur)

**3.1.143****Editeur**

source d'un Dataset en diffusion (voir: Souscripteur)

**3.1.144****PV Name**

indicatif d'une Variable de Processus

**3.1.145****PV Set**

ensemble de Variables de Processus appartenant au même Dataset

**3.1.146****queue**

mémoire stockant un ensemble ordonné de trames selon une stratégie premier arrivé, premier servi

**3.1.147****panier**

équipement contenant un ou plusieurs dispositifs, associé au même segment

**3.1.148****réassemblage**

action consistant à reconstituer un long message à partir de plusieurs paquets générés par une segmentation

**3.1.149****récepteur**

dispositif électronique qui peut recevoir des signaux en provenance du support physique

**3.1.150****Queue de Réception**

queue de réception de Données de Messagerie sur un dispositif

**3.1.151****fonctionnement normal**

activité établie du bus par opposition au mode Inauguration (sur le WTB) ou au mode Configuration (sur le MVB)

### **3.1.152**

#### **répéteur**

interconnexion entre segments de bus au niveau de la Couche Physique permettant une extension du bus au-delà des limites permises par des composants passifs. Les segments interconnectés fonctionnent à la même vitesse et selon le même protocole. Le retard introduit par un répéteur est de l'ordre d'un temps bit.

### **3.1.153**

#### **Répondeur (Replier)**

Processus d'Application à qui l'Appelant transmet un Message d'Appel et qui répond par un Message de Réponse

### **3.1.154**

#### **taux d'erreur résiduel**

probabilité d'atteinte à l'intégrité (bit erroné non reconnu) par bit transmis

### **3.1.155**

#### **routeur**

interconnexion entre deux bus au niveau de la Couche Réseau, le transfert de datagrammes d'un bus sur l'autre s'effectuant sur la base de leur Adresse Réseau

### **3.1.156**

#### **balayage**

interrogation des dispositifs selon une séquence donnée à des fins de supervision

### **3.1.157**

#### **section**

portion d'un segment connectée passivement à une autre section sans terminaison intermédiaire

### **3.1.158**

#### **segment**

portion de câble à laquelle les dispositifs sont connectés, terminée à ses deux extrémités par son impédance caractéristique. Les segments peuvent être composés de plusieurs sections (non terminées) connectées par des connecteurs

### **3.1.159**

#### **segmentation**

division d'un long message en plusieurs trames de longueur plus courte lors de sa transmission

### **3.1.160**

#### **Queue de Transmission**

queue d'émission de Données de Messagerie au sein d'un dispositif

### **3.1.161**

#### **service**

capacités et moyens d'un sous-système (une couche de communication, par exemple) offerts à un utilisateur

### **3.1.162**

#### **En-tête de Session**

partie d'une trame de Données de Messagerie relevant de la Couche Session

### **3.1.163**

#### **Couche Session**

couche du modèle OSI en charge de l'établissement et de la terminaison des communications

**3.1.164****Flanc A**

un flanc d'un véhicule par rapport à un Nœud du WTB

**3.1.165****Flanc B**

autre flanc d'un véhicule par rapport à un Nœud du WTB

**3.1.166****Esclave**

dispositif qui reçoit ou envoie des informations en provenance ou sur le bus en réponse à une requête (aussi appelée interrogation) émise par le Maître

**3.1.167****Trame-Esclave**

trame envoyée par un Esclave

**3.1.168****Délimiteur de Début de Trame-Esclave**

Délimiteur de Début d'une Trame-Esclave sur le MVB

**3.1.169****Dispositif Emetteur (source)**

émetteur d'une trame au niveau de la Couche de Liaison (voir: dispositif destinataire)

**3.1.170****transmission apériodique**

transmission réalisée sur demande, lorsqu'un événement extérieur au réseau le requiert (aussi appelée transmission apériodique, sur événement, sur demande)

**3.1.171****Données Apériodiques**

trames de données transmises sur demande pour transmettre des Données de Messagerie ou des Données de Supervision

**3.1.172****Phase Apériodique**

deuxième partie de la Période de Base, dédiée à la transmission de messages sur demande et de données de gestion de bus

**3.1.173****coupleur en étoile**

dispositif recevant la lumière en provenance d'une fibre optique et la retransmettant sur plusieurs autres fibres

**3.1.174****Délimiteur de Début de Trame**

délimiteur qui annonce le début d'une trame sur le MVB

**3.1.175****Station**

dispositif capable de communiquer des messages, par opposition aux dispositifs simples, et prenant en charge une Fonction Agent

**3.1.176**

**Répertoire (Table) de Station**

répertoire qui établit la correspondance entre Indicatif de Station et Adresse de Liaison et inversement

**3.1.177**

**indicatif de Station**

mot de 8 bits identifiant une Station

**3.1.178**

**Mot d'Etat de la Station**

descripteur de 16 bits définissant l'état et les capacités d'une Station

**3.1.179**

**Maître Fort**

Nœud Fort qui est Maître actif et n'y renonce pas tant qu'il n'a pas été rétrogradé en Nœud Faible

**3.1.180**

**Nœud Fort**

nœud sélectionné par l'application pour devenir Maître Fort. Il ne peut exister qu'un Maître Fort par segment de bus

**3.1.181**

**branche**

connexion en T à partir d'une ligne de bus électrique (au niveau de l'embranchement) permettant de connecter un dispositif à la ligne

**3.1.182**

**Souscripteur**

un des destinataires d'un Dataset en diffusion (voir: Éditeur)

**3.1.183**

**Données de Supervision**

données transmises sur un bus uniquement dans le cadre de la supervision de la Couche de Liaison (par exemple, pour l'arbitrage sur le MVB ou l'Inauguration sur le WTB)

**3.1.184**

**Adresse Système**

Adresse Réseau d'un Message de Gestion échangée entre un Gestionnaire et un Agent, consistant en une Adresse de Nœud et un Indicatif de Station

**3.1.185**

**embranchement**

endroit où un segment est connecté. Un embranchement est une fourche électrique à trois voies

**3.1.186**

**télégramme**

Trame-Maître et Trame-Esclave correspondante considérées comme un tout

**3.1.187**

**terminaison**

circuit qui termine une ligne de transmission électrique, de manière idéale sur son impédance caractéristique

**3.1.188****Interrupteur (Commutateur) de Terminaison**

interrupteur qui insère la Terminaison en bout de segment sur le WTB

**3.1.189****Topographie**

structure de données décrivant les Nœuds connectés au Bus de Train, incluant leurs adresse, orientation, position et Descripteur de Nœud

**3.1.190****topologie**

type d'interconnexion de câble et nombre de dispositifs qu'un réseau donné prend en charge

**3.1.191****Compteur de Topographie**

compteur dans un Nœud incrémenté lors de chaque nouvelle Inauguration

**3.1.192****Traffic Store (Mémoire Partagée)**

mémoire partagée accessible tant par le réseau que par l'utilisateur, et qui contient le Port de Données de Processus

**3.1.193****Réseau Embarqué de Train (TCN – Train Communication Network)**

réseau de communication de données permettant de connecter des équipements électroniques programmables embarqués sur des véhicules ferroviaires

**3.1.194****bus de Train**

bus connectant les véhicules d'un train, en particulier le WTB, et qui est conforme aux protocoles TCN

**3.1.195****Gestion de Réseau de Train (TNM – Train Network Management)**

services de Gestion de Réseau du TCN

**3.1.196****émetteur-récepteur**

association d'un émetteur et d'un récepteur

**3.1.197****émetteur**

dispositif électronique permettant de transmettre un signal sur le support physique

**3.1.198****Données de Transport**

données transportées par la Couche Transport, mais non interprétées par celle-ci

**3.1.199****En-tête de Transport**

partie d'une trame de Données de messagerie relevant de la Couche Transport

**3.1.200****Couche Transport**

couche du modèle OSI responsable du contrôle de flux de bout en bout et de la reprise sur erreur

### 3.1.201

#### **câble principal**

câble circulant le long des véhicules par opposition au câble d'extension ou au câble de jonction

### 3.1.202

#### **Adresse Utilisateur**

Adresse Réseau d'un Message Utilisateur échangé entre Fonctions, composée d'une Adresse de Nœud (ou Adresse de Groupe) et d'un Indicatif de Fonction

### 3.1.203

#### **Message Utilisateur**

message échangé entre des Fonctions utilisateur

### 3.1.204

#### **Variables**

service de transmission du TCN

### 3.1.205

#### **Var\_Offset**

offset (décalage) en bit d'une Variable de Processus dans un Dataset

### 3.1.206

#### **Descripteur de Véhicule**

informations descriptives d'un véhicule particulier dépendant de l'application (sa longueur et son poids, par exemple)

### 3.1.207

#### **Maître Faible**

Nœud Faible qui est Maître actif et prêt à abandonner son rôle à un Maître Fort s'il s'en présente un

### 3.1.208

#### **Nœud Faible**

Nœud qui peut prendre la maîtrise du bus spontanément, mais qui l'abandonne si un Nœud Fort se manifeste

### 3.1.209

#### **Bus de Train Filaire (WTB – Wire Train Bus)**

Bus de Train conçu pour des véhicules fréquemment accouplés et désaccouplés, comme par exemple les trains internationaux UIC

## 3.2 Abréviations

ALI	Application Layer Interface (Interface de Couche d'Application), la définition de la sémantique de tous les services réseau utilisés par l'application (un ensemble de primitives, exprimées en tant que procédures, constantes et types de données)
AMA	Application Messages Adapter (Adaptateur de Messagerie d'Application), code directement appelé par l'application qui met en œuvre le service de Messagerie
AMI	Application Messages Interface (Interface de Messagerie d'Application), définition des services de messagerie
ANSI	American National Standard Institute, organisme de normalisation des États-Unis
ASI	Application Supervision Interface (Interface de Supervision d'Application), définition des services de Gestion

ASN.1	Abstract Syntax Notation Number 1, concernant la présentation des données (ISO/CEI 8824)
AVA	Application Variables Adapter (Adaptateur de Variables d'Application), code directement appelé par l'application mettant en œuvre les services de Variables de Processus
AVI	Application Variables Interface (Interface de Variables d'Application), définition des services de Variables de Processus
BER	Basic Encoding Rules (Règles de Codage de Base), syntaxe de transfert pour les types de données ASN.1 (ISO/CEI 8825)
BR	Bit Rate (Débit Binaire), débit de passage des données sur le support, exprimé en bits par seconde (bit/s) ou en hertz (Hz), selon ce qui est approprié
BT	Bit Time (Temps bit), durée de la transmission d'un bit, exprimée en $\mu$ s
UIT	Union Internationale des Télécommunications, organisme de normalisation international pour les télécommunications dont le siège est à Genève
CRC	Contrôle par Redondance Cyclique, vérification de l'intégrité des données par division en polynômes
DIN	Deutsches Institut für Normung, organisme national de normalisation allemand
EIA	Electronics Industries Association, organisme de normalisation des Etats-Unis
EMD	Electrical Middle Distance (Support Electrique Moyenne Distance), un des supports du MVB
ESD	Electrical Short Distance (Support Electrique Courte Distance), un des supports du MVB
EP	Câble de frein électro-pneumatique tel qu'il est décrit dans la fiche 648 de l'UIC
ERRI	European Railway Research Institute, laboratoire dont le siège est à Utrecht, Pays-Bas
FCS	Frame Check Sequence (Séquence de Contrôle de Trame), code de détection d'erreur ajouté aux données transmises, tel que spécifié dans l'ISO/CEI 13239
GCT	Guidelines for Conformance Test (Directives pour l'Essai de Conformité), essais de conformité spécifiés à l'Annexe B de la présente norme
HDLC	High-level Data Link Control (Commande de Liaison de Données à Haut Niveau), protocole de Couche de Liaison dont le format de trame est défini dans l'ISO/CEI 13239
CEI	Commission Electrotechnique Internationale, Genève
IEEE	Institute of Electrical and Electronics Engineers, New York
ISO	Organisation Internationale de Normalisation, Genève
LLC	Logical Link Control (Contrôle de Liaison Logique), sous-couche de la Couche de Liaison gérant l'échange de données
LME	Layer Management Entity (Entité de Gestion de Couche), entité en charge de la supervision d'une couche pour la Gestion de Réseau
LMI	Layer Management Interface (Interface de Gestion de Couche), services fournis par la LME

MAC	Medium Access Control (Contrôle d'Accès au Support), sous-couche au sein de la Couche de Liaison décidant quel dispositif est autorisé à émettre vers le bus
MAU	Medium Attachment Unit (Dispositif de Connexion au Support), partie d'un Nœud qui assure l'interface électrique vers le bus et qui fournit/accepte des signaux logiques binaires
MIB	Management Information Base (Base d'Informations de Gestion), ensemble de tous les objets auxquels la Gestion de Réseau accède
MVB	Multifunction Vehicle Bus (Bus de Véhicule Multifonctions), Réseau de Rame.
NRZ	Non-Return to Zero, schéma d'encodage le plus simple dans lequel un bit est représenté par un niveau pour un "1" et l'autre niveau pour un "0", ou inversement avec une synchronisation séparée
OGF	Optical Glass Fibre (Fibre de Verre Optique), un des supports du MVB
ORE	Office de Recherches et d'Essais, laboratoire UIC dont le siège est à Utrecht, Pays-Bas
OSI	Interconnexion de Systèmes Ouverts, modèle de communication universel défini dans l'ISO/CEI 7498
PCTR	Protocol Conformance Test Report (Rapport d'Essai de Conformité au Protocole), défini dans l'ISO/CEI 9646
PICS	Protocol Implementation Conformance Statement (Déclaration de Conformité d'une Mise en œuvre de Protocole), définie dans l'ISO/CEI 9646
PTA	Process Data to Traffic Store Adapter (Adaptateur des Données de Processus au Traffic Store), composant qui accède à un des Traffic Stores (Mémoires Partagées)
RIC	Règlement pour l'emploi réciproque des voitures et des fourgons en trafic international, édité par l'UIC
RTP	Real Time Protocole (Protocoles en Temps Réel), protocoles de communication communs spécifiés à l'Article 11 de la présente norme
SDL	Specification and Description Language (Langage de description et de spécification), langage de spécification défini par l'UIT-T Z100 Annexe D pour les protocoles de communication
TCN	(Train Communication Network) Réseau Embarqué de Train, ensemble de réseaux de rame communicants et de Réseaux Centraux de Train
TNM	(Train Network Management) Gestion de Réseau de Train
UIC	Union Internationale des Chemins de fer, association internationale des exploitants de chemins de fer
WTB	(Wire Train Bus) Bus de Train Filiaire

### 3.3 Conventions

#### 3.3.1 Base des valeurs numériques

La présente norme utilise une représentation décimale pour toutes les valeurs numériques, sauf mention contraire.

Les valeurs analogiques et fractionnaires comportent une virgule.



EXEMPLE 1 La tension est de 20,0 V.

Les valeurs binaires et hexadécimales sont représentées en convention ASN.1 (ISO/CEI 8824).

EXEMPLE 2 Le nombre décimal '20' est codé sur 8 bits = '0001 0100'B = '14'H

### 3.3.2 Conventions d'appellation

La première lettre de chaque mot-clé utilisé dans les spécifications du TCN est une majuscule.

Lorsque le mot est composé, les différentes parties de l'appellation sont séparées par un espace.

Lorsqu'une structure de données est associée à un mot-clé, le type de ladite structure se compose des mêmes mots de base séparés par un trait bas.

Lorsque la valeur correspondant au mot-clé est transmise dans un message, le champ correspondant porte le même nom que le type de la structure, mais en minuscule.

Lorsque la valeur est transmise comme paramètre, le paramètre porte le même nom que le champ dans un message.

Dans les diagrammes SDL, la variable correspondante porte le même nom que le type de la structure, mais sans trait bas.

#### EXEMPLES

Topo Counter est un compteur de la couche de liaison;

Il est du type Topo\_Counter, qui est un UNSIGNED6.

Lorsque sa valeur est transmise dans un message, le champ correspondant s'appelle 'topo\_counter'.

Lorsque sa valeur est transmise par une interface de procédure, le paramètre s'appelle 'topo\_counter', son type en 'C' est Type\_Topo\_Counter.

Dans les diagrammes SDL, la variable représentant le compteur s'appelle TopoCounter.

### 3.3.3 Conventions pour les valeurs de temps

Les valeurs de temps qui commencent par une minuscule (par exemple, t mm) sont des intervalles de temps mesurables.

Les valeurs de temps qui commencent par une majuscule (par exemple, T reply) sont des paramètres ou des valeurs de temporisation.

### 3.3.4 Conventions pour les interfaces de procédure

Une interface de procédure est définie par un ensemble de primitives de service qui représentent une interaction abstraite, indépendante de la mise en œuvre, entre l'utilisateur du service et le fournisseur de service.

Dans la présente norme, ces primitives sont exprimées en tant que procédures dans la syntaxe ANSI C avec des types de paramètres.

Il convient de considérer qu'il s'agit uniquement d'une description sémantique qui n'implique pas une mise en œuvre ou un langage particuliers. Toute interface qui fournit la même sémantique est autorisée.

La conformité à la syntaxe de cette interface ne peut pas être revendiquée. Les mises en œuvre peuvent librement modifier les noms de procédure ou de paramètres, pour ajouter des paramètres ou scinder des procédures, tant que le service spécifié est fourni.

Les procédures d'interface sont définies dans la syntaxe ANSI C en utilisant la police Courier.

Les noms de procédures, les variables et les paramètres apparaissent tous en minuscule.

EXEMPLE 1 Im\_send\_request

Les constantes et les définitions de type apparaissent toutes en majuscule.

EXEMPLE 2 UNSIGNED32

Le nom d'une procédure ou d'un type se voit ajouter un préfixe:

pour le service de Variables:

- lp\_ ou LP\_ Couche de Liaison
- ap\_ ou AP\_ Couche Application

pour le service de Messagerie

- MD\_ messages en général
- Im\_ ou LM\_ Couche de Liaison
- nm\_ ou NM\_ Couche Réseau
- tm\_ ou TM\_ Couche Transport
- sm\_ ou SM\_ Couche Session
- am\_ ou AM\_ Couche Application

Le Tableau 1 donne un exemple d'un modèle utilisé pour les procédures et les types.

**Tableau 1 – Modèle pour la spécification d'une procédure d'interface**

Définition	<p>Le type de service ou de données est exprimé ici.</p> <p>Dans le cas d'une procédure d'indication, l'événement qui déclenche l'appel est indiqué ici, en commençant par "When" ("Quand")</p> <p>Le nom et les paramètres de la procédure de service sont définis ici.</p> <p>Dans le cas d'une procédure d'indication, le type de la procédure est spécifié.</p> <p>Les paramètres d'entrée, de sortie et de retour sont différenciés.</p>	
Syntaxe	<pre>MD_RESULT          lm_send_request                     ( /* example */                       unsigned,      destination,                       UNSIGNED8      link_control,                       MD_PACKET *    p_packet                       ENUM8 *        status                     )</pre>	
Entrée	Les paramètres d'entrée sont fournis à la procédure, qui n'est pas autorisée à les modifier	
	destination	Le type de données "unsigned" dépend du compilateur
	link_control	paramètre passé par référence, non modifié par la procédure. Le type de données est un mot en 8 bits
	p_packet	"*" indique un pointeur vers une structure de données p_packet, du type MD_PACKET, défini ailleurs dans la présente norme
Sortie	Il est prévu que l'appel modifie les paramètres de sortie	
	status	Le type ENUM8 est un type d'énumération en 8 bits.
Résultat	Le paramètre Result est un paramètre de sortie facultatif qui exprime le succès ou l'échec de l'appel, mais pas nécessairement du service.	
	MD_RESULT	<p>Le paramètre Result est un paramètre du type:</p> <p>AM_RESULT pour l'AMI,</p> <p>MD_RESULT pour le LMI,</p> <p>LP_RESULT pour le LPI,</p> <p>AP_RESULT pour l'AVI,</p> <p>Le modèle spécifie les codes d'erreur prévisibles pour chaque procédure individuellement.</p> <p>Le paramètre Result n'est pas décrit de manière explicite si les deux seules valeurs prévisibles sont:</p> <p>xx_OK = 0   achevé avec succès;</p> <p>xx_FAILURE   &lt;&gt; 0 problèmes rencontrés.</p> <p>Le résultat peut également être retourné comme paramètre de sortie dans la liste des paramètres, selon la mise en œuvre.</p>
Usage	Les règles répertoriées après le modèle de procédure indiquent comment il convient d'utiliser la procédure. Bien que les règles d'usage ne soient pas obligatoires, ne pas les suivre produit des résultats imprévisibles.	
<p>NOTE Les structures de données représentées dans ce Tableau sont des spécifications d'interface qu'il convient de ne pas confondre avec les formats des mêmes structures de données lorsque celles-ci sont transmises par l'intermédiaire d'un bus, voir 3.3.5.</p>		

### 3.3.5 Spécification des données transmises

Le format des données transmises, que ce soit des trames isolées ou des messages complets, est spécifié de deux manières:

- une forme graphique, qui n'est pas normative, mais qui montre intuitivement la structure du message;
- une forme textuelle basée sur ASN.1, dont les règles de codage sont spécifiées en 6.3 de la CEI 61375-2-1.

EXEMPLE 1 La forme graphique d'un message est présentée au Tableau 2, la forme textuelle correspondante est présentée au Tableau 3.

**Tableau 2 – Exemple de structure de message**

premier octet transmis								octet suivant transmis								
bit->	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	snu	gni	node_id						station_Id							
2	next_station_Id								rsv1	Tvd	topo_counter					
4	tnm_key								sif_code (= 3)							
6	paramètre 1															
8	paramètre 2								paramètre 3						par4	
	paramètre 5: ARRAY [n] OF (répétition du champ suivant (n) fois)															
	paramètre 5.1															
	paramètre 5.2															
	parameter5.3: STRING32															
	(CHARACTER8)								dernier caractère, sinon 0							
^octet																

Dans la forme graphique, une ligne est employée pour chaque mot de 16 bits, sauf dans l'Article 4 (WTB), où les lignes sont de huit bits.

Les tableaux de paramètres sont précédés d'un cadre de répétition par-dessus et sur le côté gauche.

Les répétitions peuvent être imbriquées (voir paramètre 5.3 dans le Tableau 4).

Si la taille d'un paramètre peut excéder trois mots, trois lignes sont accordées au paramètre, la ligne du milieu ayant une bordure grisée.

**Tableau 3 – Exemple de forme de message textuel (correspondant au Tableau 2)**

```

Call_Mgt_Message ::= RECORD
{
  snu                BOOLEAN1 (=1)      -- '1' signifie que le message
                                           emploie un adressage
                                           système
  gni                BOOLEAN1 (=0)      -- '0' signifie que l'adresse
                                           finale est un dispositif
                                           individuel
  node_id            UNSIGNED6          -- adresse du noeud
                                           d'extrémité à 6 bits
  station_id         UNSIGNED8          -- indicatif à 8 bits de la
                                           station
  next_station_id    UNSIGNED8          -- indicatif à 8 bits de la
                                           station suivante
  rsv1              BOOLEAN1 (=0)      -- ce bit est toujours '0'
  tvd               BOOLEAN1           -- ce bit indique si le
                                           compteur de topographie est
                                           valide
  topo_counter       UNSIGNED6          -- compteur de topographie de
                                           6 bits
  tnm_key            UNSIGNED8,         -- annonce un message d'appel
                                           de gestion de réseau.
  sif_code           UNSIGNED8,         -- il existe un code SIF
                                           différent pour chaque
                                           Message de Gestion
  parameter1         INTEGER16,        -- valeur de 16 bits. Si le
                                           paramètre comporte moins de
                                           16 bits, la valeur est
                                           justifiée à droite et
                                           étendue par son signe (par
                                           exemple, un seul octet est
                                           transmis comme deuxième
                                           octet).
  parameter2         INTEGER8,         -- cette valeur est transmise
                                           dans la partie de niveau 1
                                           plus élevé d'un mot.
  parameter3         UNSIGNED6         -- cette valeur est transmise
                                           dans l'octet de poids
                                           faible, mais les deux bits
                                           inférieurs de cet octet
                                           sont réservés pour le
                                           parameter4.
  par4              ANTIVALENT2        -- parameter4 comporte deux
                                           bits
  parameter5         ARRAY [n] OF     -- parameter5 représente des
                                           données structurées à
  {
    parameter5.1     INTEGER16         -- répéter n fois, contenant:
                                           premier paramètre du champ
                                           répété
    parameter5.2     UNIPOLAR4.16      -- deuxième paramètre du champ
                                           répété
    parameter5.3     STRING32          -- le troisième paramètre est
                                           une chaîne (tableau jusqu'à
                                           32 caractères de 8 bits),
                                           terminée par un '0', ou par
                                           deux '0' pour un alignement
                                           sur une limite d'un mot à
                                           16 bits,
                                           - une chaîne vide est
                                           composée de

```

32 caractères '0';  
- la taille réelle d'une  
chaîne est déduite du  
nombre de caractères  
significatifs qui précèdent  
le zéro.

```
} ,  
}
```

Les noms de champs et leur type commencent respectivement par une minuscule et une majuscule. Parfois, le même type est utilisé comme format de transmission, auquel cas seule la première lettre est une majuscule, et de type C, auquel cas, le type complet est en majuscule.

EXEMPLE 2 Am\_Result (format de transmission) et AM\_RESULT (type C d'une procédure d'interface).

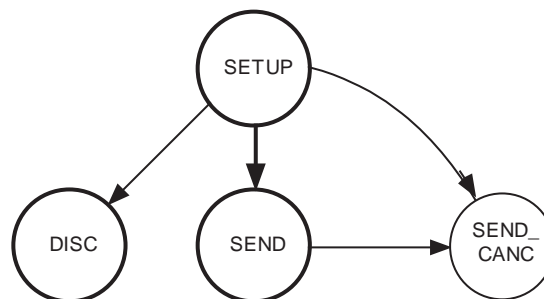
### 3.3.6 Conventions de diagrammes d'état

La machine d'état de protocole de transport est décrite dans l'ISO/CEI 8802-2 (Couche de Liaison Logique) sous la forme d'un tableau qui spécifie les transitions entre les états possibles dans lesquels une machine d'état peut se trouver.

Les transitions entre les états sont régies par les événements venant de la Couche Réseau (paquets d'entrée), de la Couche Session (commandes) ou des temporisations.

Une action dépendant de l'événement est exécutée avant de quitter l'état. Cette action définit l'état suivant.

La Figure 2 présente un exemple de diagramme de transition d'état.



#### Légende

Anglais	Français
SETUP	REGLAGE
DISC	DISQUE
SEND	ENVOI
SEND_CANC	ANN_ENVOI

**Figure 2 – Exemple de transition d'état**

A partir de "SETUP" (« REGLAGE »), la machine peut aller vers trois états différents, DISC (DISQUE), SEND (ENVOI) ou SEND\_CANC (ANN\_ENVOI).

La transition entre ces états est régie comme indiqué dans le Tableau 4.

**Tableau 4 – Tableau de transitions d'état**

État actuel	Événement	Action(s)	État suivant (si <> actuel)
<b>SETUP</b>	rcv_DR	close_send (DR_reason);	DISC
	rcv_CC AND (conn_ref = CC_conn_ref)	IF (eot) THEN close_send (AM_OK); ELSE credit:= CC_credit; send_not_yet:= credit; send_data_or_cancel; END;	DISC  SEND ou SEND_CANC
	TMO AND (rep_cnt = MAX_REP_CNT)	close_send (AM_CONN_TMO_ERR);	DISC

Conformément au Tableau 4, trois événements provoquent une transition de SETUP à l'état DISC:

- rcv\_DR (Disconnect\_Request reçu, un événement réseau). L'action correspondante consiste à fermer la connexion (close\_send) avant de passer à l'état DISC;
- un autre événement réseau (Connect\_Confirm reçu avec référence correcte), conduisant soit à l'état DISC, SEND ou SEND\_CANC, selon le résultat de la procédure send\_data ou d'annulation;
- une valeur de temporisation conditionnée par le prédicat (rep\_cnt = MAX\_REP\_CNT) qui provoque également la fermeture de la connexion.

## 4 Couche Physique

### 4.1 Topologie

#### 4.1.1 Segments

Un MVB doit être constitué d'un ou de plusieurs segments de bus sur les supports suivants:

- ESD: Support Electrique Courte Distance utilisant une paire de conducteurs avec transmission différentielle selon la norme RS-485, prenant en charge jusqu'à 32 dispositifs sur une distance de 20,0 m sans qu'une séparation galvanique ne soit nécessaire, et sur des distances supérieures en utilisant une isolation galvanique, comme indiqué en 4.4;
- EMD: Support Electrique Moyenne Distance utilisant un câble à paire torsadée blindé et des transformateurs pour l'isolation galvanique, prenant en charge jusqu'à 32 dispositifs sur une distance de 200,0 m, comme indiqué en 4.5;
- OGF: support à Fibres de Verre Optiques utilisant des fibres optiques "mises en bus" à travers un coupleur en étoile sur une distance jusqu'à 2,0 km, destiné à des environnements critiques comme les locomotives, comme indiqué en 4.6.

#### 4.1.2 Coupleurs

Les segments du MVB doivent être interconnectés par des coupleurs de l'un des types suivants:

- répéteur pour la connexion de différents supports;
- coupleurs en étoile pour la formation d'un bus à partir de fibres optiques.

**EXEMPLE** Une configuration MVB, indiquée à la Figure 3, est constituée d'un segment ESD prenant en charge un administrateur de bus, de plusieurs dispositifs à l'intérieur et à l'extérieur





### 4.1.3 Segments à ligne double

Il est admis, à titre d'option, de dupliquer un segment pour augmenter la disponibilité (segment à ligne double).

Un MVB peut être constitué d'une combinaison de segments à ligne double et de segments à ligne simple connectés par des coupleurs de bus.

## 4.2 Classes de dispositifs

### 4.2.1 Capacités

Il existe cinq classes de dispositifs rattachés au MVB, qui se différencient par leurs capacités.

Les dispositifs du MVB doivent présenter un sous-ensemble des six capacités suivantes, répertoriées dans le Tableau 5.

**Tableau 5 – Capacités des dispositifs MVB**

Capacité	Description	Classe
Device_Status	le dispositif est capable d'envoyer son Device_Status lorsqu'il est interrogé	1, 2, 3, 4, 5
Process_Data	le dispositif est capable d'envoyer des Données de Processus lorsqu'il est interrogé et de recevoir des Données de Processus provenant d'autres dispositifs	1, 2, 3, 4, 5
Message_Data	le dispositif est capable d'envoyer des Données de Messagerie lorsqu'il est interrogé et de recevoir des Données de Messagerie provenant d'autres dispositifs. Cette capacité implique que les dispositifs soient capables d'exécuter les Protocoles en Temps Réel et aient un Agent de Gestion de Réseau	2, 3, 4, 5
User_Programmable	le dispositif est capable d'être téléchargé avec des programmes d'Application Utilisateur. Cette capacité implique que le dispositif soit doté de la capacité Message_Data.	3, 4, 5
Bus_Administrator	le dispositif est capable d'agir en tant que maître. Cette capacité implique que le dispositif soit doté des capacités Message_Data, Process_Data et Device_Status ainsi que de la capacité à lire le Device_Status de tous les autres dispositifs.	4, 5
TCN_Gateway	le dispositif est capable d'accéder au moins à un autre bus (MVB ou autre). Cette capacité implique les capacités Device_Status, Process_Data et Message_Data ainsi que la présence d'un routeur si au moins deux des bus obéissent aux Protocoles en Temps Réel.	5

### 4.2.2 Dispositifs de classe 0

Les dispositifs de classe 0 ne sont pas censés être dotés d'une capacité.

NOTE La classe 0 inclut des dispositifs spéciaux, comme des répéteurs ou des coupleurs en étoile, qui ne participent pas à l'échange de données d'application ou participent d'une autre manière (par exemple, avec un autre protocole).

### 4.2.3 Dispositifs de classe 1

Les dispositifs de classe 1 doivent présenter les capacités: Device\_Status et Process\_Data.

NOTE Dans ces dispositifs, la Port\_Address des Données de Processus peut être liée à leur Device\_Address; la Port\_Address peut, par exemple, être identique à la Device\_Address.

### 4.2.4 Dispositifs de classe 2

Les dispositifs de classe 2 doivent présenter les capacités: Device\_Status, Process\_Data et Message\_Data.

NOTE Les dispositifs de classe 2 sont des dispositifs de terrain intelligents qui peuvent être configurés, mais non programmés, à travers le bus.

#### 4.2.5 Dispositifs de classe 3

Les dispositifs de classe 3 doivent présenter les capacités: Device\_Status, Process\_Data, Message\_Data et User\_Programmable.

#### 4.2.6 Dispositifs de classe 4

Les dispositifs de classe 4 doivent présenter les capacités: Device\_Status, Process\_Data, Message\_Data et Bus\_Administrator.

NOTE La capacité User\_Programmable est facultative.

#### 4.2.7 Dispositifs de classe 5

Les dispositifs de classe 5 doivent présenter les capacités: Device\_Status, Process\_Data, Message\_Data et TCN\_Gateway.

Les dispositifs de classe 5 peuvent présenter la capacité Bus\_Administrator.

NOTE Les passerelles comportant la capacité Bus\_Administrator peuvent synchroniser les bus.

#### 4.2.8 Raccordement des dispositifs

Trois types de dispositifs de raccordement sont considérés:

- a) ESD: Dispositifs équipés des connecteurs supportant des signaux ESD selon 4.4;
- b) EMD: Dispositifs équipés des connecteurs supportant des signaux EMD selon 4.5;
- c) OGF: Dispositifs équipés des connecteurs supportant des signaux ESD selon 4.6.

NOTE Un dispositif peut prendre en charge deux raccords ou plus.

### 4.3 Spécifications communes à tous les supports

#### 4.3.1 Vitesse de signalisation

La vitesse de signalisation doit être de 1,5 Mbit/s  $\pm$  0,01 %, avec le codage Manchester (BR = 1,5 MHz ou 1,5 Mbit/s, BT = 666,7 ns).

#### 4.3.2 Délais de propagation

Le délai de réponse entre deux dispositifs quelconques, qui inclut les délais de propagation et les délais de répéteur, et supposant que le retard le plus défavorable de T\_source\_max (voir 6.2.3) ne doit pas dépasser la valeur définie pour T\_reply\_def = 42,7  $\mu$ s (voir 6.2.2), sauf si le bus fonctionne en Mode Extended Reply (Réponse Etendue).

Si le délai de propagation le plus défavorable dépasse T\_reply\_def, le bus peut fonctionner en mode Extended Reply Delay (Retard de Réponse Etendue), sous réserve que le délai aller et retour le plus défavorable de T\_reply\_max soit inclus comme un paramètre de configuration de tous les dispositifs connectés.

La différence entre les délais de propagation maximal et minimal sur deux télégrammes consécutifs, considérant la totalité des délais de propagation et de répéteur, ne doit en aucun cas excéder 4,0 BT.

La somme des intervalles entre les trames d'une trame-maître à une autre doit être supérieure à 9,0 BT.

NOTE 1 La valeur par défaut de  $T_{reply\_def}$  de 42,7  $\mu s$  limite à quatre le nombre de répéteurs et à 2,0 km la distance de transmission. Il est possible d'utiliser une valeur supérieure à  $T_{reply\_def}$  avec certaines mesures de prévention qui sont expliquées dans le mode Extended Reply Delay (6.2.4.2).

NOTE 2 Les fluctuations du délai de propagation peuvent réduire l'intervalle entre les trames et peuvent générer un chevauchement de trames. Les répéteurs peuvent augmenter l'intervalle entre les trames, mais un dépassement peut survenir si l'exigence ci-dessus est dépassée.

NOTE 3 L'imposition d'une somme minimale aux intervalles entre les trames permet d'éviter à une paire rapide maître/esclave de dépasser les auditeurs les plus lents.

### 4.3.3 Interface émetteur-récepteur

L'interface émetteur-récepteur spécifie, pour tous les types de supports, l'interface entre le Bus\_Controller et les émetteurs-récepteurs à l'intérieur d'un dispositif.

Cette interface peut rester interne au dispositif, mais il est recommandé de la rendre accessible pour des essais. Cette interface ne fait pas l'objet des essais de conformité.

L'interface émetteur-récepteur est définie comme une interface électrique fonctionnant avec des signaux binaires.

L'interface émetteur-récepteur suppose que le support prend deux niveaux différents, un niveau HIGH (HAUT) et un niveau LOW (BAS). Les niveaux HIGH et LOW sont définis pour chaque support.

Si elle est exposée, l'interface émetteur-récepteur doit être constituée des signaux suivants:

a) TxS: Transmitter\_Signal

Ce signal commande le niveau du support. Le signal est un "0" lorsque le support est au niveau LOW, ou un "1" lorsque le support est au niveau HIGH;

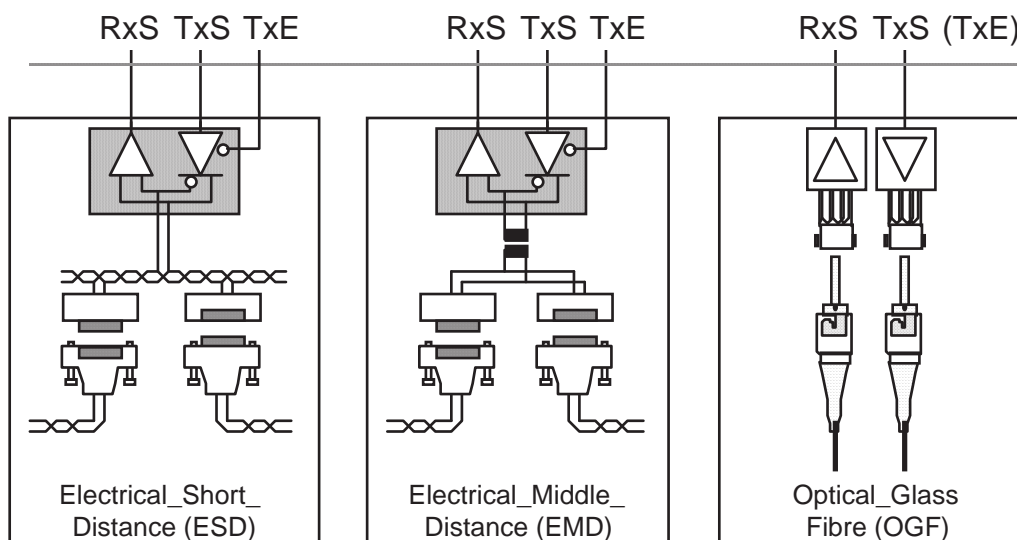
b) TxE: Transmitter\_Enable

Ce signal devient un "1" pour activer l'émetteur. Ce signal n'est pas nécessaire pour la transmission optique. Sa durée est définie pour chaque support;

c) RxS: Receiver\_Signal

Ce signal indique l'état du support. Il s'agit d'un "0" lorsque la ligne est au niveau LOW, et d'un "1" lorsque la ligne est au niveau HIGH. Le récepteur interprète les niveaux indéfinis soit comme LOW, soit comme HIGH. Il n'y a pas de niveau défini lorsqu'aucun émetteur n'est actif, bien que certains supports définissent un niveau inactif (généralement LOW).

La Figure 4 montre l'interface émetteur-récepteur pour l'ensemble des trois supports.



**Légende**

Anglais	Français
Electrical_Short_Distance (ESD)	Electrique Courte Distance (ESD)
Electrical_Middle_Distance (EMD)	Electrique Moyenne Distance (EMD)
Optical_Glass_Fibre (OGF)	Fibre de Verre Optique (OGF)

**Figure 4 – Interface émetteur-récepteur**

#### 4.3.4 Support redondant (option)

Pour les applications nécessitant une grande disponibilité, ce paragraphe définit un système de redondance selon lequel le bus peut être entièrement dupliqué (tous les segments sont à ligne double) ou partiellement dupliqué (à la fois les segments à ligne simple et les segments à ligne double). Si cette option est utilisée, les spécifications suivantes s'appliquent.

##### 4.3.4.1 Identification

Un segment à ligne double doit être constitué de deux lignes fonctionnant en parallèle.

Tous les dispositifs connectés à un segment à ligne double doivent identifier la même ligne comme étant la Line\_A et l'autre ligne comme étant la Line\_B.

##### 4.3.4.2 Agencement

L'agencement des bus et de leur charge doit garantir des conditions de propagation proches sur les deux lignes d'un segment à ligne double afin de satisfaire à 5.2.2.2.

L'agencement doit faire l'objet de mesures de prévention pour garantir l'indépendance des lignes redondantes par rapport aux défaillances.

##### 4.3.4.3 Connexions entre segments à ligne simple et segments à ligne double

Lorsqu'un segment à ligne double est connecté à un autre segment à ligne double par un répéteur, il doit y avoir un répéteur indépendant pour chaque Line\_A et chaque Line\_B et l'identification des lignes Line\_A et Line\_B doit être conservée par cette connexion.

Lorsqu'un segment à ligne double est connecté à un segment à ligne simple par un répéteur, ce dernier doit être connecté aux deux lignes d'un côté et à une ligne de l'autre côté, comme spécifié en 5.3.2.

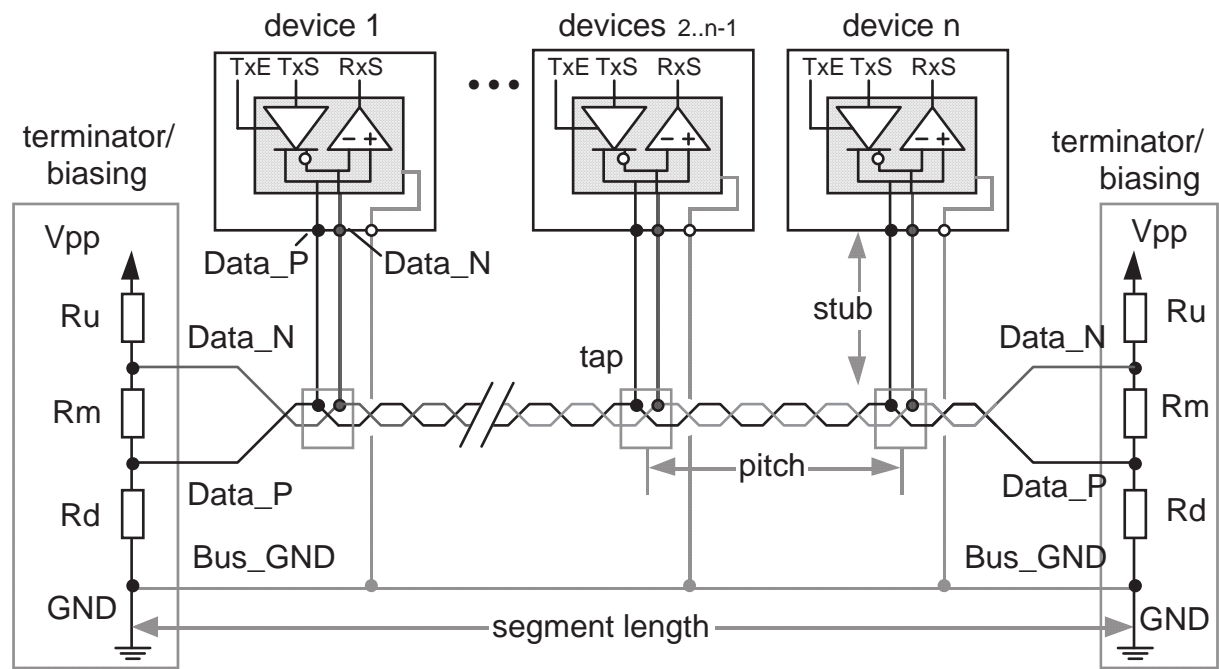
4.4 Support Électrique Courte Distance (choix)

Les spécifications du Support Electrique Courte Distance permettent la présence de segments comprenant jusqu'à 32 dispositifs sur une distance allant jusqu'à 20 m sans séparation galvanique. En fait, la distance couverte et le nombre de dispositifs sont limités par la distorsion causée par les câbles, les connecteurs et les dispositifs, par la qualité de la terre et par les niveaux de perturbation, et ils peuvent être augmentés par l'emploi d'une isolation galvanique.

4.4.1 Topologie ESD

Le support ESD doit être constitué de deux conducteurs (plus un conducteur équipotentiel), adaptés et polarisés à chaque extrémité, auxquels les dispositifs sont rattachés par de courtes branches.

EXEMPLE Le principe de la connexion d'un segment ESD est illustré à la Figure 5.



Légende

Anglais	Français
device	dispositif
terminator/biasing	terminaison/polarisation
tap	embranchement
stub	branche
pitch	écartement
equipotential conductor	conducteur équipotentiel
segment length	longueur de segment

Composants	Type	Valeur	Connexions	Type	Valeur
$R_u$	Résistance	383 $\Omega$	$V_{pp}$	Tension d'alimentation	5,0 V
$R_m$	Résistance	143 $\Omega$	GND	Tension de référence	0,0 V
$R_d$	Résistance	383 $\Omega$			

Figure 5 – Exemple de segment ESD

#### **4.4.2 ESD - Règles de configuration**

##### **4.4.2.1 ESD - Support de transmission**

Une ligne ESD doit être constituée d'une paire de conducteurs, de préférence torsadés et blindés, accompagnés d'un conducteur équipotentiel en parallèle.

##### **4.4.2.2 ESD - Identification**

Les deux conducteurs d'une ligne doivent être appelés Data\_P (données positif) et Data\_N (données négatif).

Les conducteurs Data\_P et Data\_N doivent être marqués de manière distincte tout au long de la ligne.

L'identité des conducteurs doit être maintenue à tous les points de connexion ou d'épissure.

Le conducteur équipotentiel doit être appelé Bus\_GND et marqué de manière distincte.

##### **4.4.2.3 ESD - Impédance caractéristique**

L'impédance caractéristique de la ligne non chargée doit être de  $120,0 \Omega \pm 10 \%$  à 1,0 BR.

##### **4.4.2.4 ESD - Terminaison**

Un segment ESD doit être terminé électriquement à chaque extrémité par une terminaison.

La terminaison doit présenter une impédance de  $120,0 \Omega \pm 2 \%$  mesurée à 1,0 BR.

La terminaison doit polariser la paire de câbles de manière à ce que le potentiel du câble Data\_P soit inférieur d'au moins  $0,750 \text{ V} \pm 10 \%$  par rapport au câble Data\_N lorsqu'aucun émetteur n'est actif.

**EXEMPLE** La terminaison indiquée en Figure 6 présente une impédance de  $120,5 \Omega$  et polarise la ligne avec environ  $0,786 \text{ V}$ . Noter que la source  $V_{pp}$  doit avoir une impédance intrinsèque très faible dans la plage de fréquences comprise entre 0,5 BT et 2 BT de sorte que l'impédance équivalente de la terminaison observée par la ligne se situe dans la limite de tolérance spécifiée. Les résistances relèvent de la tolérance de 1% de la série E96.

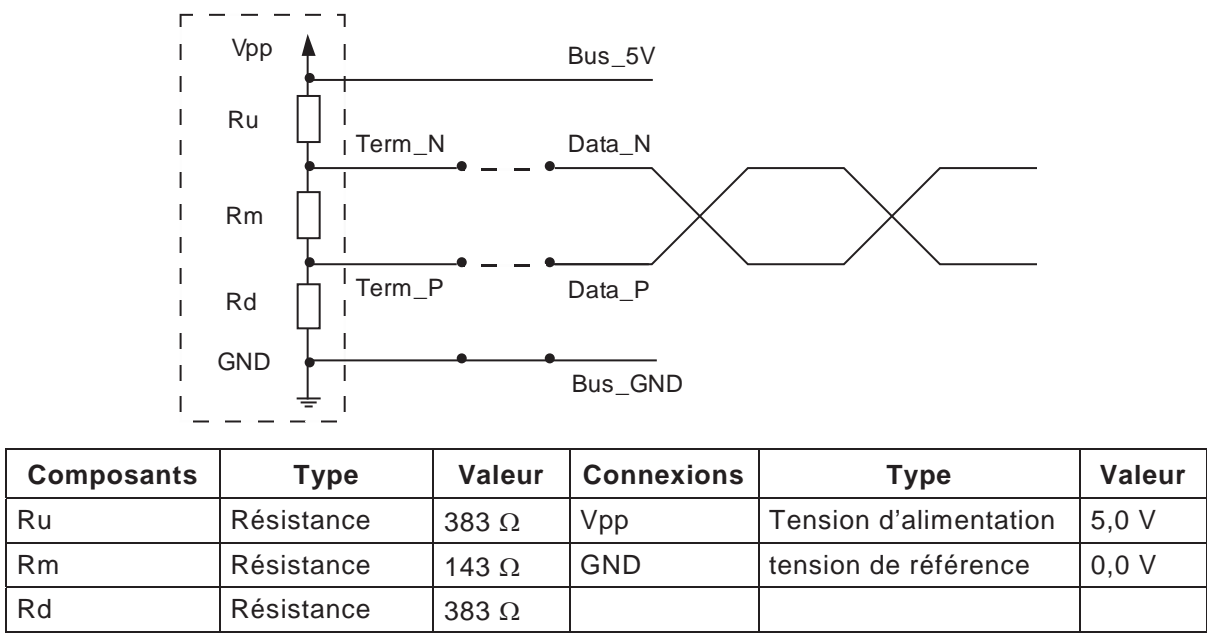


Figure 6 – Exemple de terminaison

4.4.2.5 ESD - Atténuation due au support

L'atténuation totale de la tension entre deux dispositifs situés sur la même ligne ne doit pas dépasser 8,0 dB.

Pour cet essai, l'atténuation est mesurée en appliquant à une terminaison un signal sinusoïdal d'amplitude différentielle de 4,0 Vpp ± 10 % (1,414 Veff) à 2,0 BR et en mesurant le signal à l'autre terminaison.

4.4.2.6 ESD – gigue due aux dispositifs, aux connecteurs et au câblage

Une ligne dont tous les dispositifs et connecteurs sont en place, et dont les extrémités sont terminées par son impédance caractéristique, ne doit pas ajouter plus de ± 0,1 BT de gigue frontale par rapport aux passages par zéro théoriques.

Pour cet essai, la gigue est mesurée en alimentant la ligne à une terminaison par une source d'amplitude différentielle de 4,0 Vpp ± 10 %, centrée sur 0,0 V, ayant une impédance de source de 22,0 Ω ± 10 %, et produisant comme signal une séquence pseudo-aléatoire de symboles Manchester "0" et "1" avec une période de répétition d'au moins 511 bits, le signal étant mesuré à l'autre terminaison.

NOTE 1 Les perturbations et les réflexions dues aux désadaptations d'impédance entre les sections de ligne, les branches, les connecteurs ou les accumulations de charge peuvent provoquer une gigue dans la cadence des passages par zéro.

NOTE 2 Cette méthode d'essai est spécifiée dans l'ISO/CEI 8802-3.

4.4.3 ESD – spécifications des sections

Les spécifications ESD s'appliquent à deux types de sections:

- a) la section de fond de panier est utilisée dans des enceintes telles que des paniers et des armoires, lorsqu'aucune séparation galvanique n'est requise, sur des distances jusqu'à 20,0 m;

- b) la section de câble est employée pour connecter des dispositifs hors des enceintes, avec ou sans séparation galvanique. En cas d'utilisation d'une séparation galvanique, des distances jusqu'à 200,0 m peuvent être couvertes.

#### **4.4.3.1 ESD – section de fond de panier**

Si un bus de fond de panier est utilisé comme support, les spécifications suivantes s'appliquent.

L'extension maximale d'une branche doit être de 10,0 cm, distance mesurée entre le câble principal du bus et l'entrée de l'émetteur-récepteur.

L'écartement entre des embranchements adjacents ne doit pas être inférieur à 2,0 cm pour éviter l'accumulation de charge.

#### **4.4.3.2 ESD – section de câble**

Si une section de câble est utilisée comme support, les spécifications relatives au câble EMD doivent s'appliquer.

Par ailleurs, les spécifications suivantes s'appliquent:

- a) un câble utilisé pour le support ESD doit comporter au moins un fil équipotentiel par paire de ligne;
- b) le fil équipotentiel d'un câble ESD doit être identifié comme A.Bus\_GND (pour Line\_A) et comme B.Bus\_GND (pour Line\_B, dans le cas où deux lignes redondantes sont employées).

NOTE La section du fil équipotentiel peut être différente de celle de la paire de fils de données.

#### **4.4.4 ESD – blindage**

Le blindage du câble doit être connecté au boîtier de chaque connecteur, qui doit être d'un matériau conducteur.

Le boîtier du connecteur doit établir un contact électrique avec le boîtier du dispositif une fois inséré.

Le blindage du câble ne doit pas être utilisé à la place de Bus\_GND.

#### **4.4.5 ESD - interface dépendante du support**

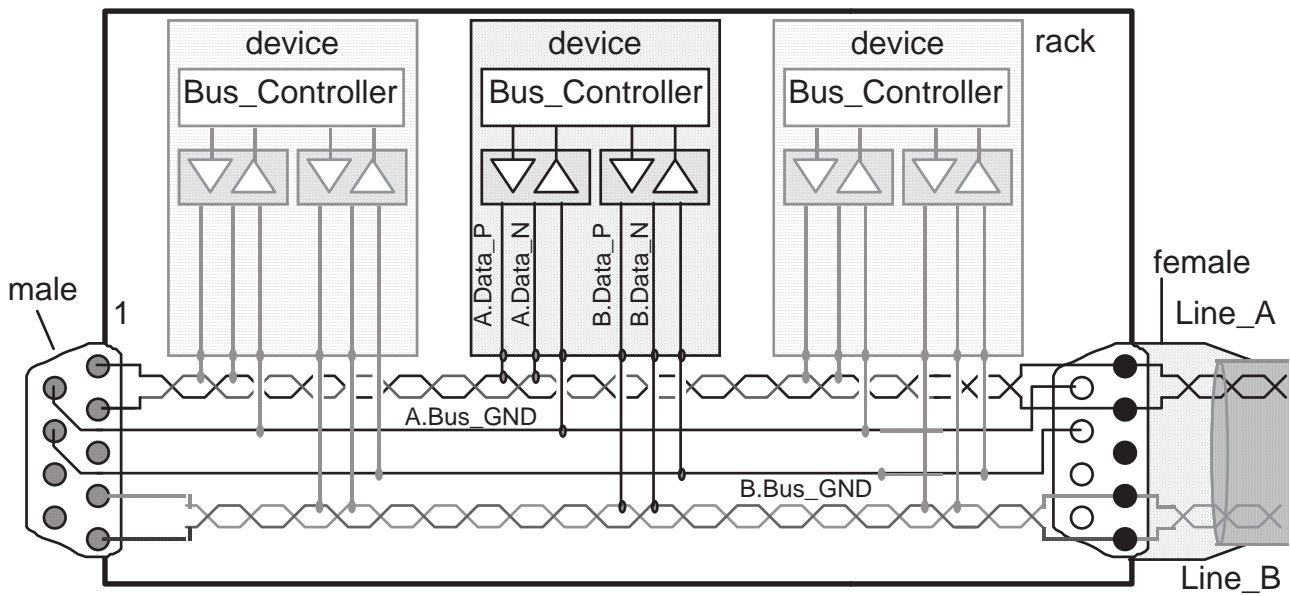
NOTE L'interface dépendante du support est décrite pour un segment à ligne double, même lorsque cette option n'est pas retenue.

##### **4.4.5.1 ESD – raccordement**

Un dispositif doit être raccordé à la Line\_A et/ou à la Line\_B par des points de connexion indépendants qui sont identifiés tel que le montre la Figure 7, sous la forme suivante:

- A.Data\_P et A.Data\_N;
- B.Data\_P et B.Data\_N, respectivement.





Légende

Anglais	Français
device	dispositif
rack	panier
male	mâle
female	femelle

Figure 7 – Section ESD de fond de panier (ligne double)

NOTE 1 Ce schéma englobe une méthode de redondance dans laquelle les deux lignes redondantes utilisent des paires torsadées indépendantes qui partagent les mêmes câble et connecteur. Cette méthode protège contre les défaillances des fils, des fiches et des émetteurs-récepteurs, mais pas contre la rupture d'un câble ou le retrait d'un connecteur. L'expérience montre que ces derniers risques sont mineurs.

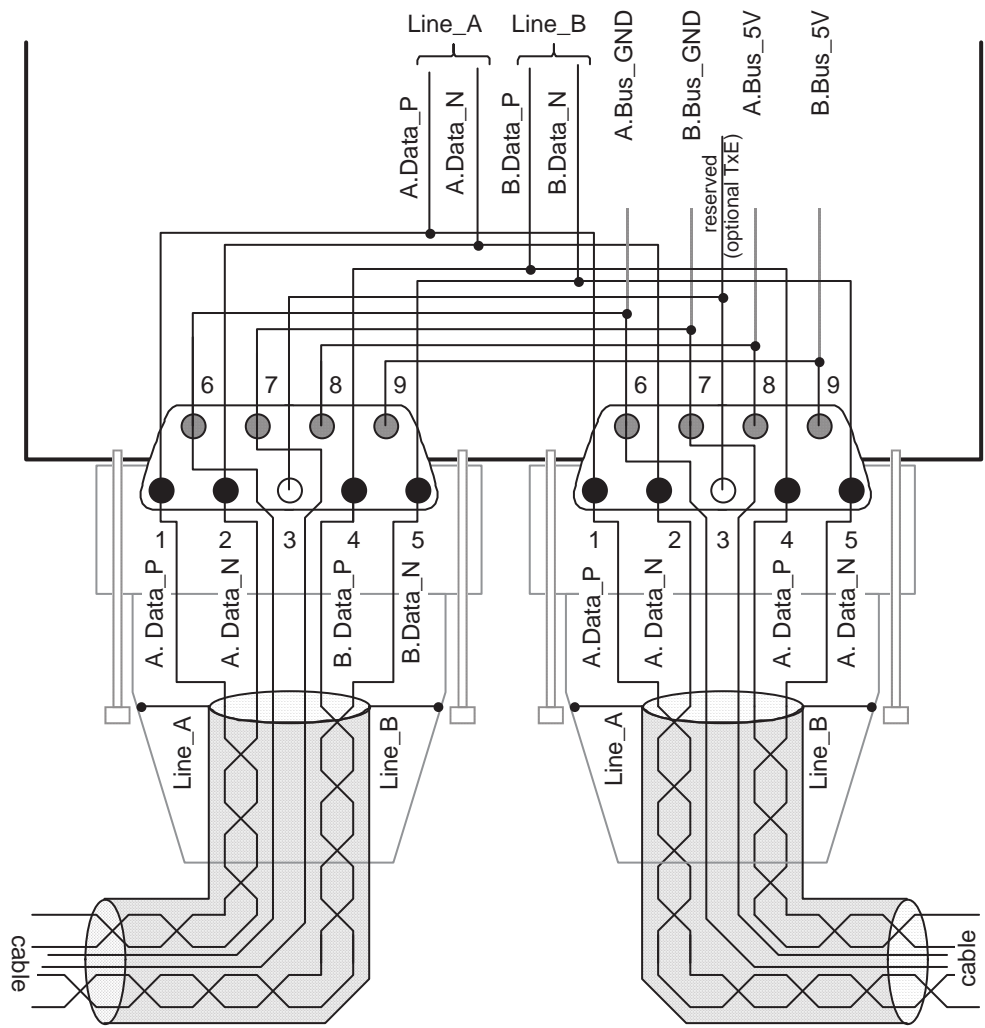
NOTE 2 Dans des applications où des connecteurs et des câbles indépendants sont requis, il convient que le brochage du connecteur soit identique à celui utilisé pour le raccordement à deux connecteurs et qu'il ait la même polarité (mâle ou femelle) qu'une section à ligne simple.

4.4.5.2 ESD – connecteur

Lorsque l'interchangeabilité est nécessaire, les dispositifs doivent être raccordés au câble comme suit:

- a) chaque dispositif doit être inséré dans le câble par deux connecteurs Sub-D9 à 9 broches avec vis métriques (CEI 60807), appelés Connector\_1 (connecteur 1) et Connector\_2 (connecteur 2);
- b) le connecteur doit avoir un capot conducteur connecté au blindage du câble et qui établit un contact électrique avec le réceptacle une fois fixé;
- c) il doit être possible de connecter et de fixer ensemble des connecteurs de câble afin de garantir la continuité du câble et du blindage;
- d) l'affectation des broches du connecteur (mâle ou femelle) doit être celle indiquée au Tableau 6;
- e) les réceptacles des connecteurs doivent être marqués "MVB-S1" et "MVB-S2" pour identifier le support ESD;
- f) une section de câble ou de fond de panier doit avoir un connecteur mâle à une extrémité et un connecteur femelle à l'autre extrémité;
- g) les connecteurs doivent avoir la polarité et l'agencement illustrés à la Figure 8;

- h) Connector\_1 doit utiliser le connecteur mâle sur le dispositif et le connecteur femelle sur le câble;
- i) Connector\_2 doit utiliser le connecteur femelle sur le dispositif et le connecteur mâle sur le câble;
- j) un dispositif ou un panier doit fournir à travers le connecteur, et pour chaque ligne, une alimentation de 5,0 V  $\pm$  5 % avec une intensité minimale de 20,0 mA, limitée à 300,0 mA, dont la terre est reliée au conducteur équipotentiel;



Légende	
Anglais	Français
reserved (optional)	réservé (facultatif)
cable	câble

Figure 8 – Agencement du connecteur ESD

**Tableau 6 – Affectation des broches du connecteur ESD**

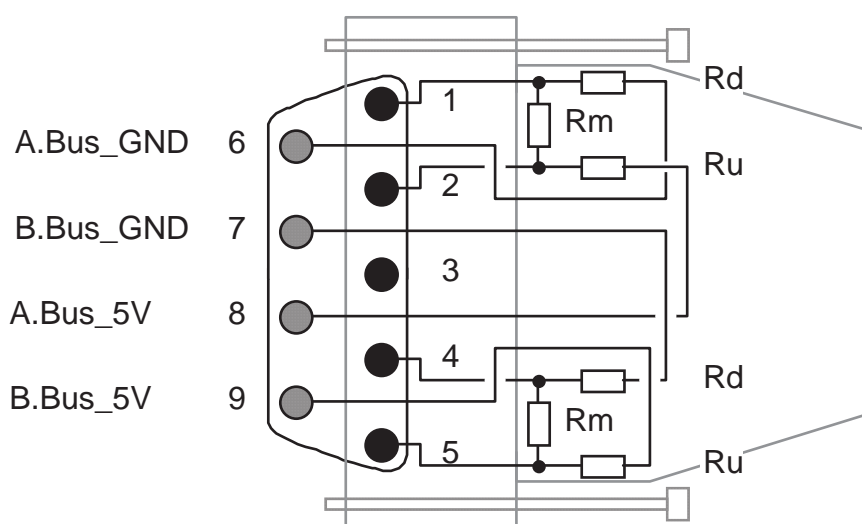
1	A.Data_P, fil positif Line_A	6	A.Bus_GND, mise à la masse Line_A
2	A.Data_N, fil négatif Line_A	7	B.Bus_GND, mise à la masse Line_B
3	TxE, voir 4.3.3 (facultatif)	8	A.Bus_5V, alimentation positive Line_A
4	B.Data_P, fil positif Line_B	9	B.Bus_5V, alimentation positive Line_B
5	B.Data_N, fil négatif Line_B		

NOTE 1 Il est recommandé de monter les deux connecteurs côte à côte sur la plaque frontale d'un dispositif, le Connector\_1 à gauche et le Connector\_2 à droite, pour simplifier le câblage.

NOTE 2 Il peut être nécessaire de garantir que le boîtier du connecteur est isolé du boîtier du dispositif si la connexion des boîtiers de dispositifs est déconseillée, par exemple, lorsque des courants vagabonds importants sont prévisibles.

#### 4.4.5.3 ESD – connecteur de la terminaison

La terminaison doit être incluse dans un connecteur à raccorder dans le réceptacle vide d'un dispositif en bout de segment, selon le câblage illustré à la Figure 9:

**Figure 9 – Agencement du connecteur de terminaison ESD**

Le connecteur contenant la terminaison pour ESD doit être marqué "MVB-S1", respectivement "MVB-S2".

NOTE Il existe deux types de connecteur de terminaison, un mâle et un femelle. Les connecteurs de terminaison sont différents dans le cas de l'ESD et de l'EMD.

#### 4.4.6 ESD – Spécifications Line\_Unit

##### 4.4.6.1 ESD – conventions (indications)

Les caractéristiques de chaque dispositif sont mesurées aux points où la ligne est raccordée au dispositif, à savoir Data\_P, Data\_N et Bus\_GND, comme le montre la Figure 7.

Lorsque la mesure concerne un émetteur, le circuit du récepteur est en état de réception normal. Lorsque la mesure concerne un récepteur, le circuit de son émetteur est dans un état d'impédance élevée.

Si le dispositif est raccordé par des connecteurs, ceux-ci sont inclus dans les mesures.

#### 4.4.6.2 ESD – pertes d'insertion d'un dispositif

La charge présentée par un dispositif doit être conforme à l'ISO/CEI 8482 (RS-485).

NOTE L'ISO/CEI 8482 spécifie que cette valeur est de 12 k à 1,0 BR.

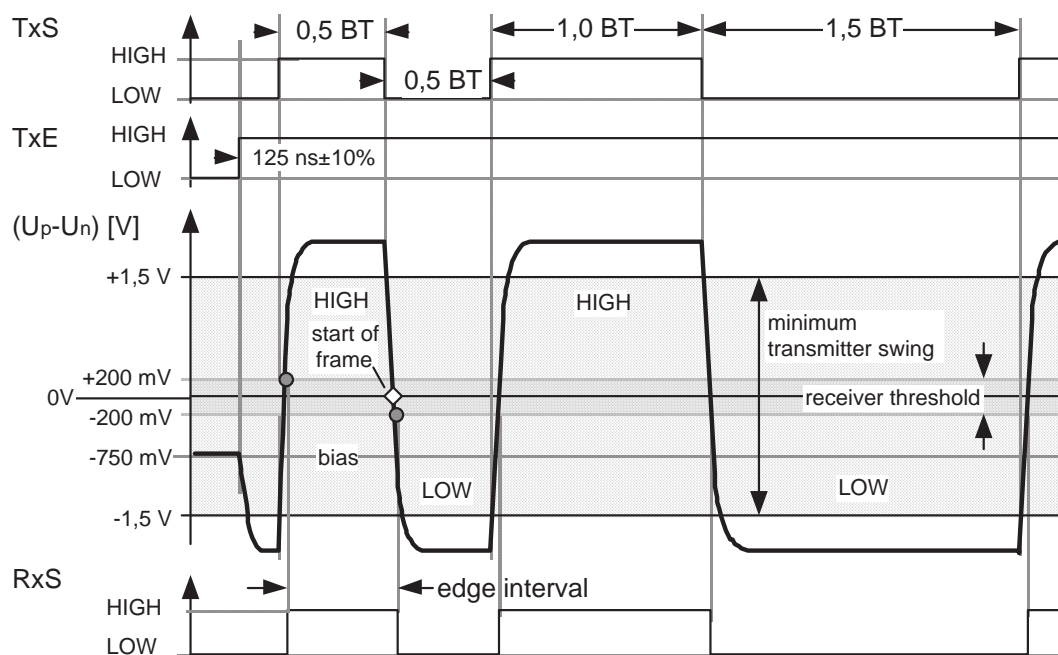
#### 4.4.7 ESD – forme d'onde du signal

Le support doit prendre deux niveaux distincts:

- un niveau 'HIGH' pour lequel la tension  $U_p$  sur Data\_P est plus élevée que la tension  $U_n$  sur Data\_N, ce qui doit correspondre à un niveau 'HIGH' des signaux TxS ou RxS;
- un niveau 'LOW' pour lequel la tension  $U_p$  sur Data\_P est plus basse que la tension  $U_n$  sur Data\_N, ce qui doit correspondre à un niveau 'LOW' des signaux TxS ou RxS.

La polarisation doit garantir que la ligne prend le niveau 'LOW' lorsqu'elle n'est pas alimentée.

EXEMPLE Le début d'une trame vue à l'émetteur (TxS et TxE), sur la ligne ( $U_p - U_n$ ) et au récepteur (RxS), sans tenir compte des délais temporels, est illustré à la Figure 10.



#### Légende

Anglais	Français
start of frame	début de trame
minimum transmitter swing	oscillation minimale de l'émetteur
receiver threshold	seuil du récepteur
bias	polarisation
edge interval	intervalle entre les fronts

Figure 10 – Exemple de début de trame (ESD)

NOTE Le premier passage par zéro "HIGH" - "LOW" de la trame définit le début de la trame.

#### 4.4.8 ESD – émetteur

##### 4.4.8.1 ESD – signal de l'émetteur

Un émetteur doit être conforme à l'ISO/CEI 8482 (RS-485), compte tenu des restrictions suivantes:

- a) le temps de montée du signal (10 % – 90 %) doit être inférieur à 0,03 BT (20,0 ns à 1,5 Mb/s) sur une charge de 54,0  $\Omega$  et une capacité parallèle de 50,0 pF;
- b) l'émetteur doit fournir une source de tension différentielle à basse impédance présentant deux niveaux actifs:
  - 'HIGH', lorsque la différence de tension ( $U_p - U_n$ ) est comprise entre:  
 $+1,5 \text{ V} < (U_p - U_n) < +5,0 \text{ V}$  lorsqu'il est chargé par une résistance de 54,0  $\Omega$ , et  
 $+1,5 \text{ V} < (U_p - U_n) < +6,0 \text{ V}$  à vide;
  - 'LOW', lorsque la différence de tension ( $U_p - U_n$ ) est comprise entre:  
 $-1,5 \text{ V} > (U_p - U_n) > -5,0 \text{ V}$  lorsqu'il est chargé par une résistance de 54,0  $\Omega$ , et  
 $-1,5 \text{ V} > (U_p - U_n) > -6,0 \text{ V}$  à vide.

NOTE Cette spécification étant plus stricte que l'ISO/CEI 8482, il convient d'accorder une attention toute particulière au choix des émetteurs-récepteurs commerciaux. Les émetteurs conformes à la CEI 61158-2 respectent cette spécification.

##### 4.4.8.2 ESD – gigue de l'émetteur

Lors de l'envoi d'une trame, la gigue entre deux fronts consécutifs entre le Start\_Bit et le End\_Delimiter ne doit pas dépasser 10,0 ns.

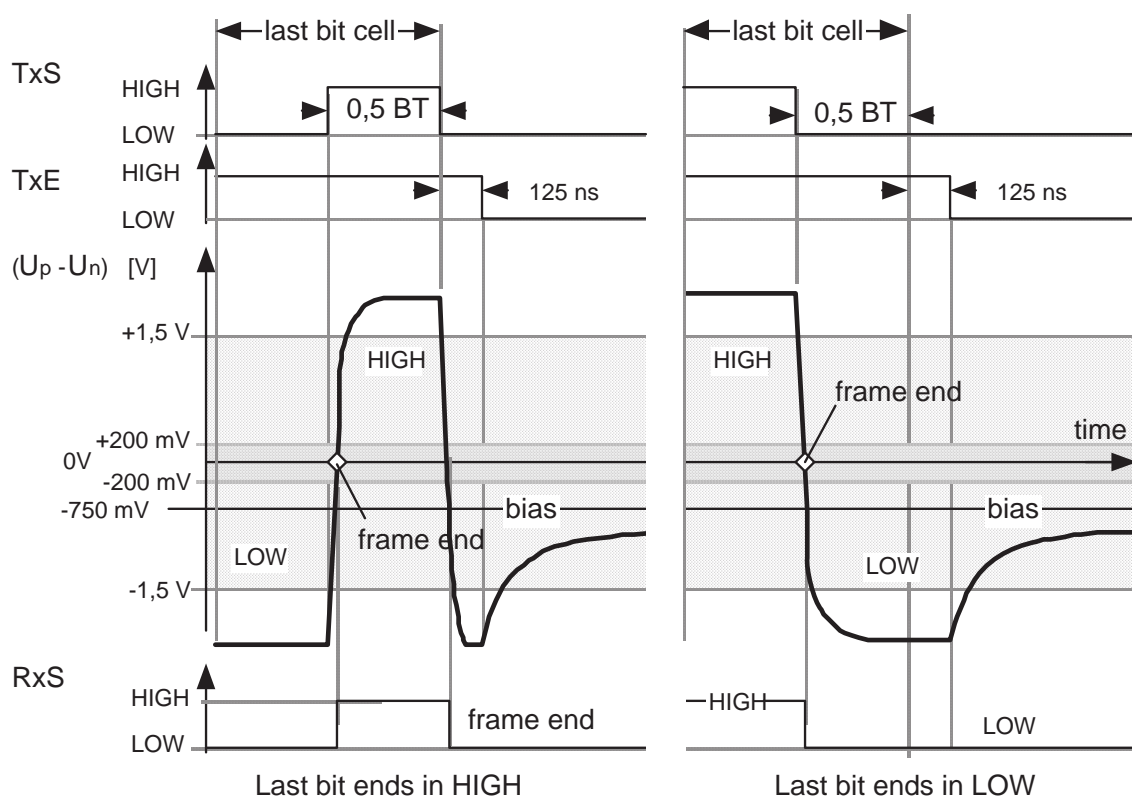
##### 4.4.8.3 ESD – début de trame de l'émetteur

Avant d'envoyer le Start\_Bit, l'émetteur doit forcer la ligne vers l'état 'LOW' pendant au moins 0,125  $\mu\text{s} \pm 0,010 \mu\text{s}$ , comme le montre la Figure 10.

NOTE Forcer la ligne vers l'état 'LOW' avant le Start\_Bit permet d'obtenir un front de départ plus raide au récepteur.

##### 4.4.8.4 ESD – fin de trame de l'émetteur

Après envoi du dernier bit d'une trame, l'émetteur doit forcer la ligne vers l'état 'LOW' pendant au moins 0,125  $\mu\text{s}$  et au plus 1,0 BT, comme le montre la Figure 11.



#### Légende

Anglais	Français
last bit cell	dernière cellule de bit
frame end	fin de trame
bias	polarisation
Last bit ends in HIGH	Fin du dernier bit en HIGH
Last bit ends in LOW	Fin du dernier bit en LOW
time	durée

Figure 11 – Fin d'une trame ESD (les deux cas)

#### 4.4.9 ESD – récepteur

Un récepteur doit être conforme à l'ISO/CEI 8482 (RS-485), compte tenu des restrictions suivantes:

- selon la tension de ligne différentielle, un récepteur doit générer deux niveaux sur sa sortie RxS:
  - un niveau 'HIGH' si la différence de tension ( $U_p - U_n$ ) est supérieure à +0,200 V, lorsque la ligne est pilotée en niveau 'HIGH';
  - un niveau 'LOW' si la différence de tension ( $U_p - U_n$ ) est inférieure à -0,200 V, lorsque la ligne est pilotée en niveau 'LOW' ou lorsque la ligne n'est pas pilotée et que seule la tension de polarisation apparaît;
- le récepteur doit comprendre une hystérésis d'au moins 0,050 V, mais d'au plus 0,200 V;
- le récepteur doit fonctionner correctement en présence de la tension de mode commun spécifiée dans le document RS-485 par rapport à la ligne Bus\_GND.

4.5 Support Electrique Moyenne Distance (choix)

Les spécifications du Support Electrique Moyenne Distance permettent l'utilisation de segments comprenant jusqu'à 32 dispositifs sur une distance allant jusqu'à 200 m lorsque le support spécifié est utilisé. La distance réelle et le nombre de dispositifs sont limités par la distorsion provoquée par les câbles, les connecteurs et les dispositifs, par la qualité de la terre et par les niveaux de perturbation. Bien que l'EMD ait été conçu pour un couplage par transformateurs, d'autres raccordements sont possibles.

4.5.1 Topologie EMD

Le support EMD doit être formé de deux conducteurs, enchaînés de dispositif en dispositif et terminés à chaque extrémité, comme le montre la Figure 12.

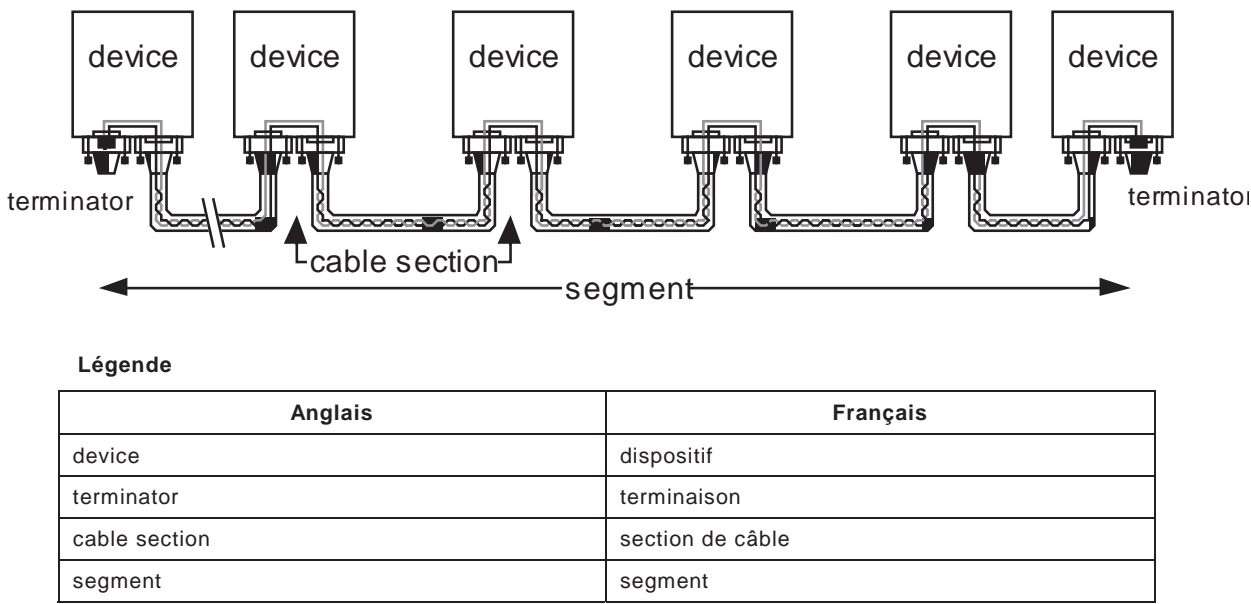


Figure 12 – Support EMD

4.5.2 EMD – règles de configuration

4.5.2.1 EMD – support de transmission

Le support EMD doit être formé d'une paire de fils torsadée et blindée.

Le signal transmis doit être une tension différentielle, de polarité non indifférente, entre ces fils.

4.5.2.2 EMD – identification

Les fils doivent être marqués de manière distincte tout au long d'un segment.

L'identité des fils doit être maintenue en tous points de connexion ou d'épissure.

4.5.2.3 EMD – atténuation due au support

L'atténuation totale du signal sur le support entre deux dispositifs, exprimée comme le rapport entre la tension au dispositif d'émission et la tension au dispositif de réception, ne doit pas dépasser 8,0 dB, mesurée à une fréquence comprise entre 0,5 BT et 2,0 BT.

NOTE L'atténuation limite la longueur d'un segment pour un nombre donné de dispositifs. Compte tenu d'une atténuation par le câble de 15,0 dB/km et d'une perte d'insertion de 0,15 dB/dispositif à 1,5 MHz, une distance de 200 m peut être couverte.

#### 4.5.2.4 EMD – gigue due aux dispositifs, aux connecteurs et au câblage

Une ligne dont tous les dispositifs et connecteurs sont en place et dont les extrémités sont terminées par son impédance caractéristique, ne doit pas ajouter plus de  $\pm 0,1$  BT de gigue frontale par rapport aux passages par zéro théoriques.

Pour cet essai, la gigue est mesurée en alimentant la ligne à une terminaison par une source d'amplitude différentielle de  $4,0 \text{ Vpp} \pm 10 \%$ , centrée sur  $0,0 \text{ V}$ , ayant une impédance de source de  $22,0 \Omega \pm 10 \%$ , et produisant comme signal une séquence pseudo-aléatoire de symboles Manchester "0" et "1" avec une période de répétition d'au moins 511 bits, le signal étant mesuré à l'autre terminaison.

NOTE 1 Les perturbations et les réflexions dues aux désadaptations d'impédance entre les sections de ligne, les branches, les connecteurs ou les accumulations de charge peuvent provoquer une gigue dans la cadence des passages par zéro.

NOTE 2 Cette méthode d'essai est spécifiée dans l'ISO/CEI 8802-3.

NOTE 3 Il est recommandé de ne pas grouper plus de 4 dispositifs sur une longueur de câble de  $10,0 \text{ m}$  et d'insérer au moins  $10,0 \text{ m}$  de câble entre un tel groupe et le suivant.

#### 4.5.3 EMD – terminaison

Une ligne doit être adaptée à chaque extrémité par une terminaison présentant une impédance de  $Z_w = 120,0 \Omega \pm 2 \%$  et un angle de phase inférieur à  $0,087 \text{ rad}$  sur la plage de fréquences comprise entre  $0,5 \text{ BR}$  et  $2,0 \text{ BR}$ .

#### 4.5.4 Section de câble

Les spécifications du câble s'appliquent aux deux supports ESD et EMD, permettant de n'employer qu'un type de câble. Les particularités du câble ESD sont indiquées en 4.4.3.2.

##### 4.5.4.1 Mécanique

Toutes les sections doivent être constituées d'un câble gainé, blindé, comportant au moins une paire de fils torsadée.

Le câble ne doit pas comporter moins de 12 torsades au mètre.

NOTE Il est recommandé que la section transversale de chaque fil soit comprise entre  $0,34 \text{ mm}^2$  (AWG 22) et  $0,56 \text{ mm}^2$  (AWG 20).

##### 4.5.4.2 Marquage

Les différents fils de la paire torsadée doivent être identifiés comme suit:

- a) pour Line\_A (même si une seule paire est utilisée): A.Data\_P et A.Data\_N;
- b) pour Line\_B (dans le cas où deux paires de lignes sont utilisées): B.Data\_P et B.Data\_N, respectivement.

Les différents fils du câble doivent être marqués distinctement.

Lorsque le câble comporte deux lignes, les paires torsadées redondantes doivent être marquées distinctement.

NOTE Les deux paires de fils peuvent être arrangées en quarte, auquel cas les fils en diagonale doivent former une paire.

##### 4.5.4.3 Impédance caractéristique

Les sections de bus doivent présenter une impédance caractéristique différentielle de  $Z_w = 120,0 \Omega (\pm 10 \%)$  par rapport aux fils de données, mesurée avec un signal sinusoïdal et à une fréquence comprise entre  $0,5 \text{ BR}$  et  $2,0 \text{ BR}$ .



#### 4.5.4.4 Atténuation due au câble

L'atténuation par le câble d'un signal sinusoïdal envoyé sur les fils de données doit être inférieure à 15,0 dB/km à 1,0 BR, et inférieure à 20,0 dB/km à 2,0 BR.

#### 4.5.4.5 Capacité répartie du câble

La capacité répartie différentielle (fil à fil) du câble ne doit pas dépasser 46,0 pF/m à 1,0 BR.

#### 4.5.4.6 Différence de capacité par rapport au blindage

La différence de capacité par rapport au blindage ne doit pas dépasser 1,5 pF/m à 1,0 BR.

#### 4.5.4.7 Rejet de diaphonie

Lorsque le même câble comporte deux paires de fils, le rejet du signal d'une paire redondante vers l'autre doit excéder 45,0 dB dans la plage comprise entre 0,5 BR et 2,0 BR.

NOTE Un même câble pour segments à ligne double peut comporter deux paires de fils.

#### 4.5.4.8 Qualité de blindage

L'impédance de transfert du câble doit être inférieure à 0,020  $\Omega$ /m à une fréquence de 20,0 MHz.

L'impédance de transfert différentielle du câble doit être inférieure à 0,002  $\Omega$ /m.

#### 4.5.4.9 Qualité de connecteur de câble

Toutes les connexions par câble doivent présenter une continuité de fils et de blindage, avec une résistance inférieure à 0,010  $\Omega$ .

L'impédance de transfert du connecteur, mesurée à 20,0 MHz, doit être inférieure à 0,020  $\Omega$  entre une broche et le blindage, respectivement 0,002  $\Omega$  entre deux broches.

NOTE 1 Ces exigences ne s'appliquent pas aux connecteurs entre véhicules.

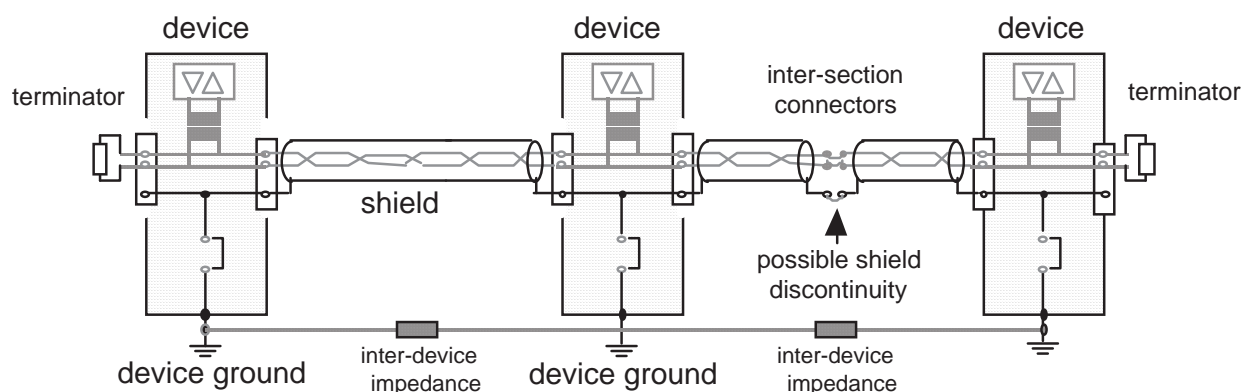
### 4.5.5 EMD – blindage

#### 4.5.5.1 EMD – continuité de blindage

Un dispositif doit connecter les blindages des deux sections de câble auxquelles il est raccordé.

Lorsqu'un dispositif est enlevé, il doit être possible de connecter les blindages entre eux, par exemple, par les connecteurs.

Le dispositif doit fournir un moyen de connecter le blindage à la masse afin de trouver un emplacement approprié pour la mise à la terre du blindage, ou de mettre en œuvre un concept de mise à la terre dans lequel tous les boîtiers sont connectés entre eux, comme le montre la Figure 13.



**Légende**

Anglais	Français
device	dispositif
terminator	terminaison
inter-section connectors	connecteurs entre sections
shield	blindage
possible shield discontinuity	discontinuité éventuelle de blindage
device ground	masse
inter-device impedance	impédance entre dispositifs

**Figure 13 – Blindage (segment à ligne simple)**

NOTE Le boîtier d'un dispositif n'est pas nécessairement connecté au réceptacle de ses connecteurs, bien que cet arrangement soit recommandé.

#### 4.5.5.2 EMD – discontinuité de blindage

Dans des applications où d'importantes différences de tension entre les dispositifs sont prévisibles, les sections de câble peuvent être connectées en des emplacements sélectionnés sans établir de continuité de blindage, comme le montre la Figure 13.

En ces emplacements, des câbles non standard, à double blindage par exemple, peuvent être utilisés pour satisfaire aux exigences de compatibilité électromagnétique.

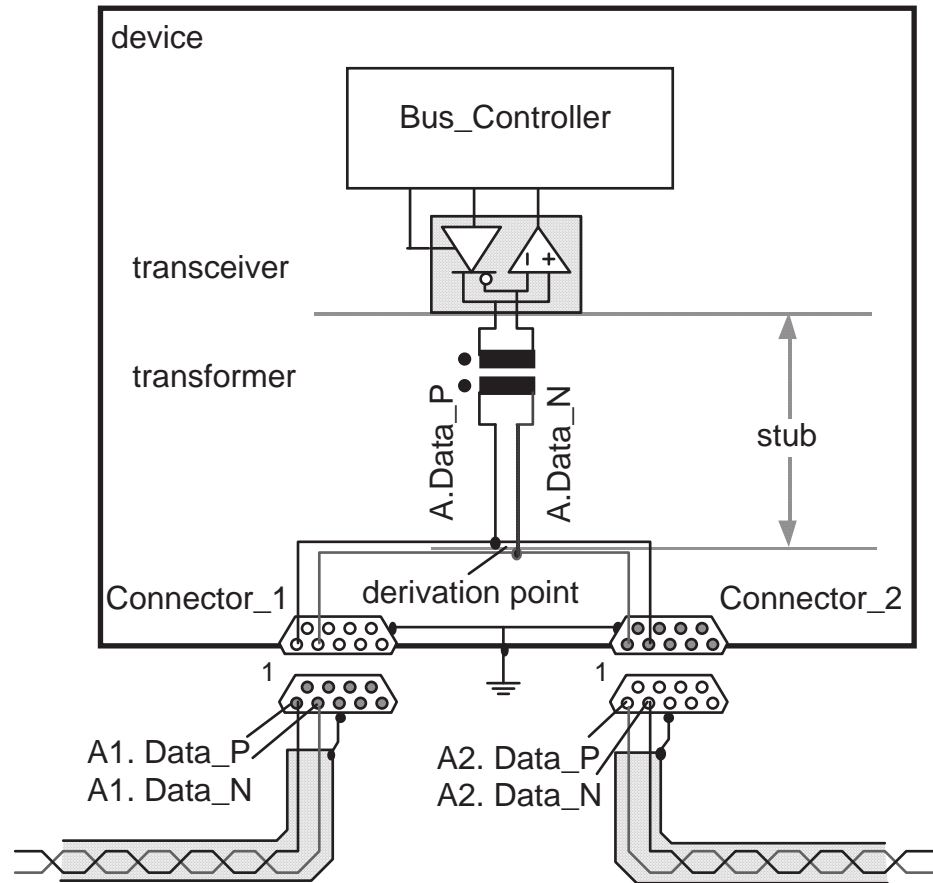
NOTE Cette exception vaut pour du câble de jonction ou des accouplements automatiques entre véhicules où des courants vagabonds élevés sont prévisibles.

#### 4.5.6 EMD - Interface dépendant du support

##### 4.5.6.1 EMD – raccordement à une ligne simple

Un dispositif doit être raccordé à la ligne par un embranchement passif.

La longueur de la branche, mesurée à partir du point d'embranchement (voir Figure 14) jusqu'à l'émetteur-récepteur, ne doit pas dépasser 10,0 cm.



Légende

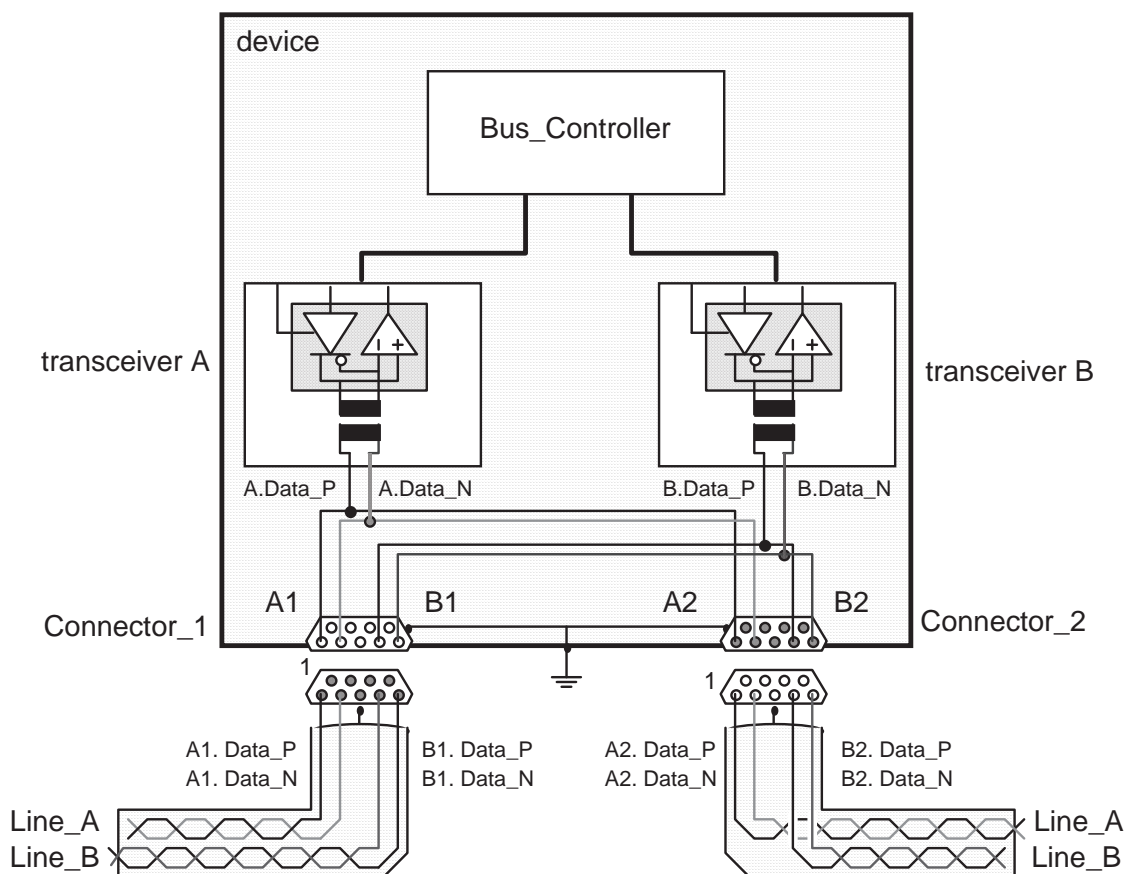
Anglais	Français
device	dispositif
transceiver	émetteur-récepteur
transformer	transformateur
stub	branche
derivation point	point d'embranchement

Figure 14 – Raccordement du dispositif à ligne simple

NOTE Pour satisfaire à cette spécification, il est recommandé que le dispositif soit raccordé par deux connecteurs, comme le montre la Figure 14.

4.5.6.2 EMD - raccordement à une ligne double (option)

Un dispositif comportant un raccordement à ligne double doit pouvoir être raccordé aussi bien à un segment EMD à ligne simple qu'à un segment EMD à ligne double, comme le montre la Figure 15.



#### Légende

Anglais	Français
device	dispositif
transceiver	émetteur-récepteur

NOTE 1 Cette méthode englobe une méthode de redondance dans laquelle les deux lignes redondantes utilisent des paires torsadées indépendantes qui partagent les mêmes câble et connecteur. Cette méthode protège contre les défaillances des fils, des fiches et des émetteurs-récepteurs, mais pas contre la rupture d'un câble ou le retrait d'un connecteur. L'expérience montre que ces derniers risques sont mineurs.

NOTE 2 Pour des applications où il convient que les lignes redondantes passent par des connecteurs séparés, il convient, dans une configuration à quatre connecteurs, que le brochage du connecteur soit le même que pour le raccordement à deux connecteurs et il convient qu'il ait la même polarité (mâle ou femelle) à une extrémité de la section d'un câble.

**Figure 15 – Raccordement du dispositif à ligne double à l'EMD**

#### 4.5.6.3 MD – connecteur

Lorsque l'interchangeabilité est nécessaire, les dispositifs doivent être raccordés au câble comme suit:

- chaque dispositif doit être inséré dans le câble par deux connecteurs Sub-D9 à 9 broches avec vis métriques (CEI 60807), appelés Connector\_1 (connecteur 1) et Connector\_2 (connecteur 2);
- les connecteurs doivent avoir un capot conducteur blindé connecté au blindage du câble et qui établit un contact électrique avec le réceptacle une fois fixé;
- il doit être possible de connecter et de fixer ensemble des connecteurs de câble afin de garantir la continuité du câble et du blindage;
- l'affectation des broches du connecteur (mâle ou femelle) doit être celle indiquée au Tableau 7;

- e) le réceptacle des connecteurs pour l'EMD doit être marqué "MVB-M1" (pour Connector\_1) et "MVB-M2" (pour Connector\_2) pour identifier le support EMD;
- f) une section de câble doit comporter un connecteur mâle à une extrémité et un connecteur femelle à l'autre extrémité;
- g) les connecteurs doivent présenter la polarité et la disposition suivantes, comme le montre la Figure 16;
- h) le Connector\_1 doit utiliser le connecteur mâle sur le dispositif et le connecteur femelle sur le câble;
- i) le Connector\_2 doit utiliser le connecteur femelle sur le dispositif et le connecteur mâle sur le câble.

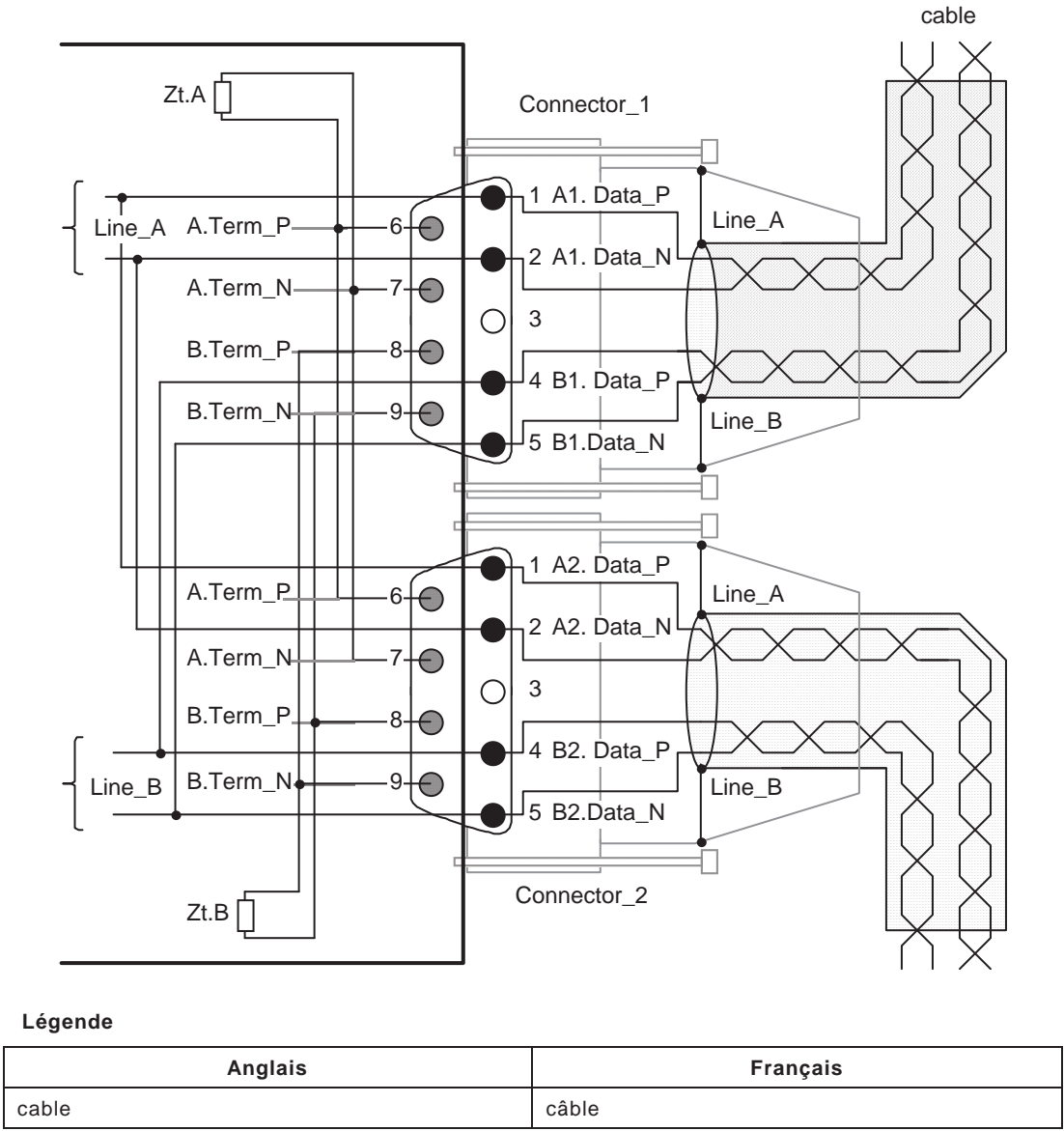


Figure 16 – Disposition des connecteurs EMD

**Tableau 7 – Affectation des broches du connecteur EMD**

1	A.Data_P fil positif de la Line_A	6	A.Term_P, pôle positif de la Terminaison Line_A
2	A.Data_Nfil négatif de la Line_A	7	A.Term_N, pôle négatif de la Terminaison Line_A
3	réservé pour TxE, voir 4.3.3 (option)	8	B.Term_P, pôle positif de la Terminaison Line_B
4	B.Data_P fil positif de la Line_B	9	B.Term_N, pôle négatif de la Terminaison Line_B
5	B.Data_Nfil négatif de la Line_B		

NOTE 1 Il est recommandé de monter les deux connecteurs côte à côte sur un dispositif, le Connecteur\_1 à gauche et le Connector\_2 à droite, pour simplifier le câblage.

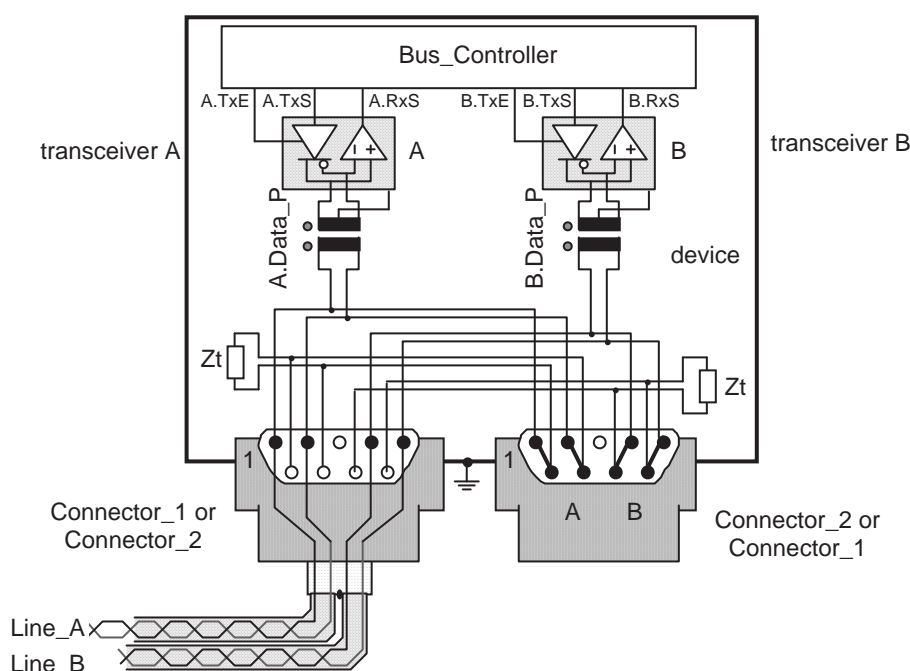
NOTE 2 Il peut être nécessaire de garantir que le boîtier du connecteur soit isolé du boîtier du dispositif si la connexion des boîtiers de dispositifs est déconseillée, par exemple, lorsque des courants vagabonds importants sont prévisibles.

#### 4.5.6.3.1 MD - Connecteur femelle (option)

Lorsqu'aucune interchangeabilité n'est nécessaire, les dispositifs peuvent être raccordés au câble comme suit: les Connector\_1 et Connector\_2 peuvent utiliser le connecteur femelle sur le dispositif et le connecteur mâle sur le câble.

#### 4.5.6.4 EMD – rebouclage de la terminaison

Pour connecter la terminaison dans le dernier dispositif d'un segment, le connecteur doit être rebouclé en connectant la broche 1 à la broche 6, la broche 2 à la broche 7, la broche 4 à la broche 8 et la broche 5 à la broche 9, comme le montre la Figure 17.



#### Légende

Anglais	Français
transceiver	émetteur-récepteur
device	dispositif
or	ou

**Figure 17 – Rebouclage de la terminaison EMD**

Les connecteurs de terminaison pour EMD doivent être marqués “MVB-M1” (pour être connectés comme Connector\_1) et “MVB-M2” (pour être connectés comme Connector\_2).

#### 4.5.7 EMD – Spécifications de Line\_Unit

Même lorsque seule Line\_Unit A est mentionnée, le présent paragraphe s'applique également à Line\_Unit B.

##### 4.5.7.1 EMD – pertes d'insertion d'un dispositif

Un dispositif:

- a) avec un récepteur fonctionnant normalement et un émetteur en état de haute impédance; ou
- b) sans énergie appliquée au récepteur ou à l'émetteur,

doit introduire une atténuation inférieure à 0,15 dB aux fréquences de 0,5 BR et 2,0 BR.

##### 4.5.7.2 EMD – séparation galvanique

Entre la masse du dispositif et chacun des points:

- a) A1.Data\_P,
- b) A1.Data\_N,
- c) A2.Data\_P, et
- d) A2.Data\_N,

la tension d'isolation et la résistance doivent dépasser la valeur spécifiée par la CEI 60571.

NOTE Dans l'édition actuelle, ces valeurs sont respectivement 0,500 kV eff et 1,0 MΩ.

#### 4.5.8 EMD – forme d'onde du signal

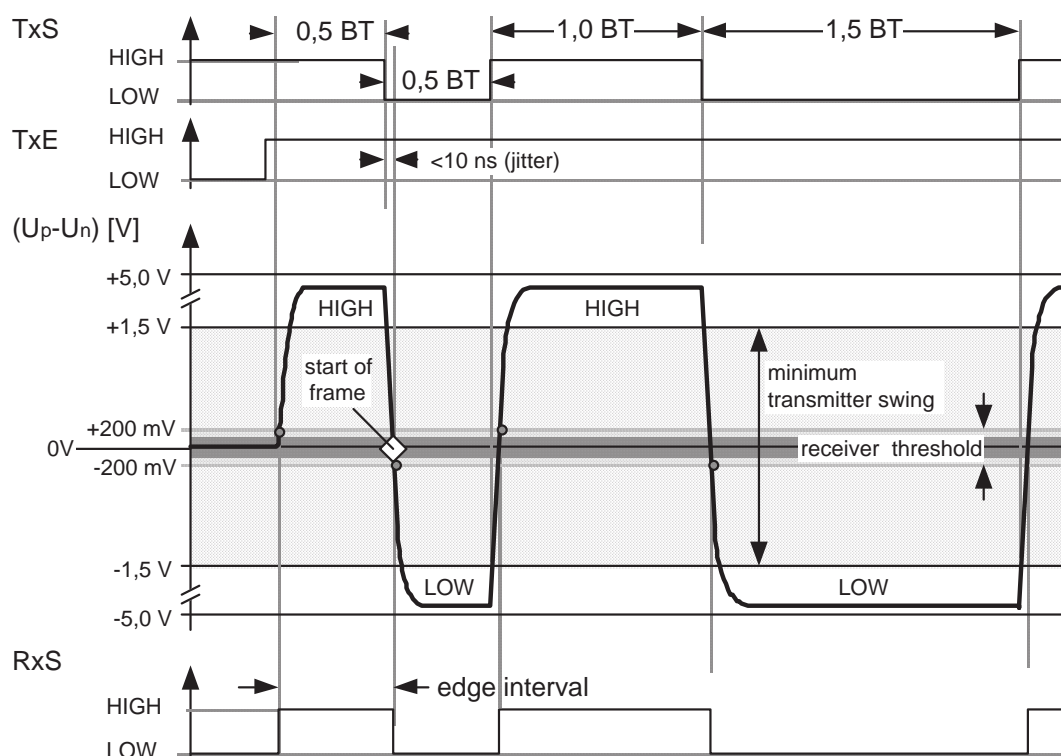
Le support doit prendre deux niveaux distincts:

- a) un niveau 'HIGH' lorsque la différence de potentiel ( $U_p - U_n$ ) entre Data\_P et Data\_N est positive, ce qui doit correspondre à un niveau 'HIGH' des signaux TxS ou RxS;
- b) un niveau 'LOW' lorsque la différence de potentiel ( $U_p - U_n$ ) entre Data\_P et Data\_N est négative, ce qui doit correspondre à un niveau 'LOW' des signaux TxS ou RxS.

Le début de la trame est défini par le premier passage par zéro 'HIGH-'LOW' de la trame.

NOTE L'état de la ligne n'est pas défini lorsque la ligne n'est pas alimentée.

EXEMPLE La Figure 18 montre le début d'une trame vue au niveau de l'émetteur (TxS), sur la ligne ( $U_p - U_n$ ) et au niveau du récepteur (RxS), sans tenir compte des délais sur la ligne ou dans les dispositifs.



#### Légende

Anglais	Français
jitter	gigue
start of frame	début de trame
minimum transmitter swing	oscillation minimale de l'émetteur
receiver threshold	seuil du récepteur
edge interval	intervalle entre les fronts

Figure 18 – Exemple de début de trame EMD

### 4.5.9 EMD – spécifications de l'émetteur

#### 4.5.9.1 EMD – signal de sortie de l'émetteur

L'émetteur doit avoir un étage de sortie différentiel.

NOTE En raison du codage des données (5.1), l'émetteur génère des impulsions dont la longueur est de un bit (1,0 BT), d'un demi-bit (0,5 BT) ou d'un bit et demi (1,5 BT).

Ces spécifications s'appliquent à des impulsions de 0,5 BT, 1,0 BT et 1,5 BT, positives ou négatives.

Lorsqu'il est connecté à la fois au circuit d'essai lourd et au circuit d'essai léger, l'émetteur doit être conforme aux exigences suivantes:

- le signal de sortie doit être alternativement positif et négatif;
- l'amplitude du signal de sortie doit être d'au moins  $\pm 1,5$  V en cas de connexion avec le circuit d'essai lourd et de  $\pm 5,5$  V au maximum en cas de connexion avec le circuit d'essai léger;

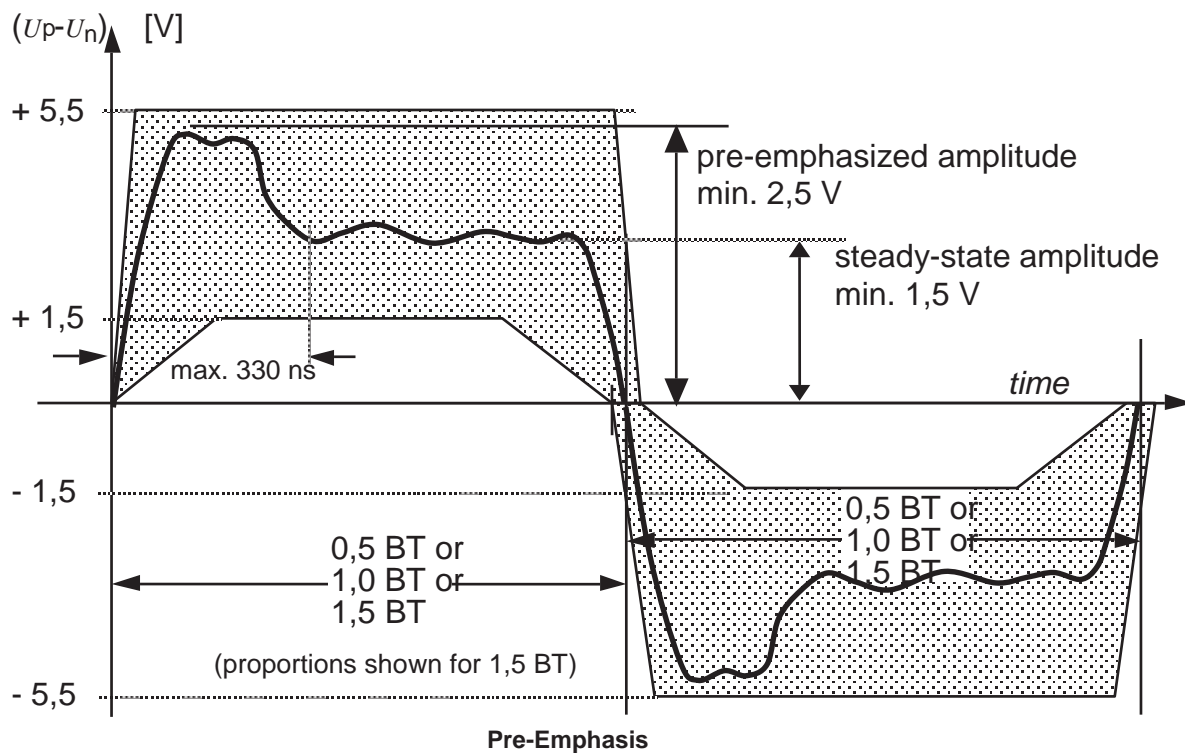
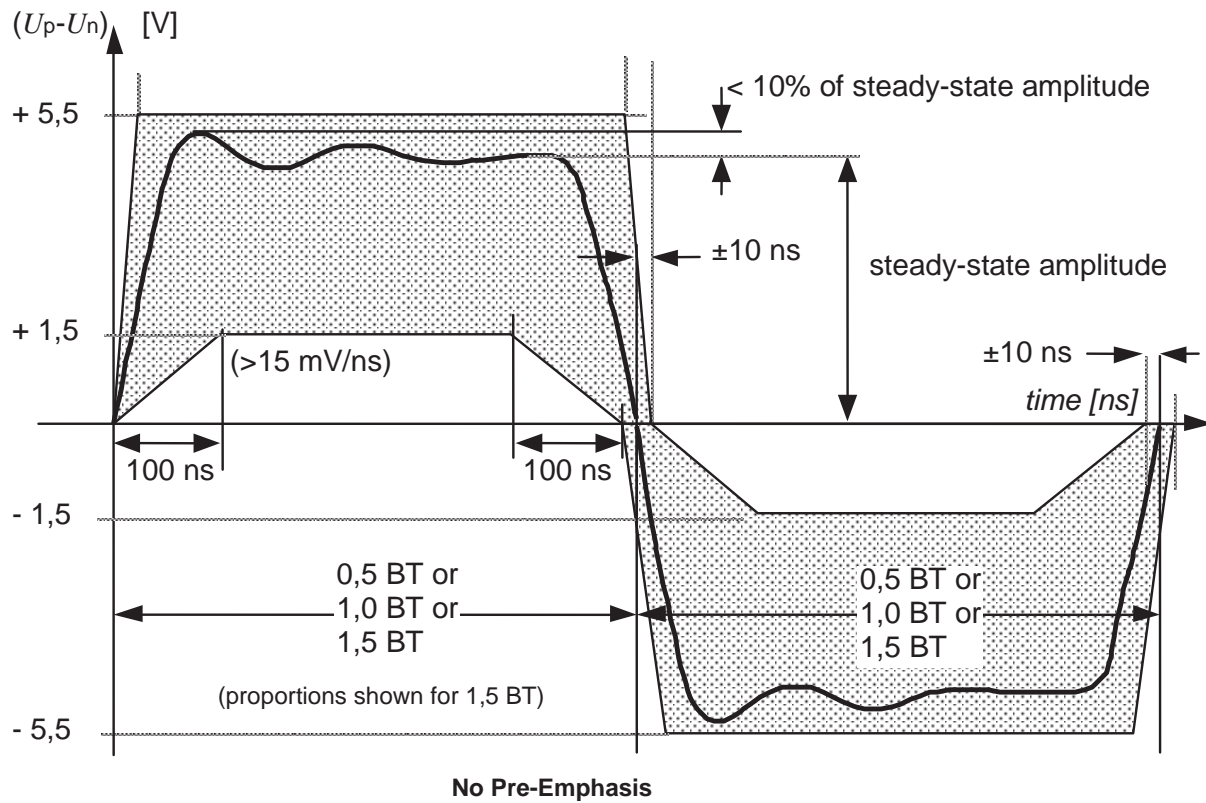


- c) la différence entre l'amplitude en régime permanent positive et négative dans deux impulsions consécutives ne doit pas dépasser 0,10 V;
- d) la vitesse de variation du signal de sortie doit être supérieure à 15 mV/ns dans une période de 0,100  $\mu$ s du passage par zéro;
- e) la suroscillation du signal de sortie, définie comme le rapport entre l'amplitude maximale et l'amplitude en régime permanent ne doit pas dépasser 10 % de son amplitude en régime permanent;
- f) la distorsion du front du signal de sortie, définie comme la différence de temps entre le passage par zéro théorique et réel, ne doit pas dépasser  $\pm 2$  % d'un Temps de Bit.

L'émetteur peut, en option, utiliser une technique appelée Pré-Emphase afin d'améliorer la qualité du signal. Dans ce cas, l'émetteur doit satisfaire aux exigences susmentionnées, à l'exception de l'exigence e) substituée par les exigences suivantes:

- g) le rapport de l'amplitude de Pré-Emphase et de l'amplitude de régime permanent doit se situer dans la plage comprise entre 165 % et 235 %;
- h) la durée de l'impulsion de Pré-Emphase, mesurée à partir de la partie frontale de la forme d'onde, doit être de 330 ns;
- i) la différence entre l'amplitude en régime permanent positive et négative dans deux impulsions consécutives ne doit pas dépasser 0,20 V.

**EXEMPLE** La Figure 19 montre une forme d'onde typique au milieu d'une trame. Le diagramme supérieur fait référence au cas de non utilisation de l'option de pré-emphase, le diagramme inférieur faisant pour sa part référence au cas d'utilisation de l'option de pré-emphase. Les valeurs minimales sont consignées.



**Figure 19 – Exemple de forme d'onde d'une impulsion au niveau de l'émetteur EMD**

**Légende**

Anglais	Français
< 10 % of steady-state amplitude	<10 % de l'amplitude en régime permanent
steady-state amplitude	amplitude en régime permanent
time	durée
proportions shown for 1,5 BT	ratios indiqués pour 1,5 BT
or	ou
No Pre-Emphasis	Pas de Pré-Emphase
pre-emphasized amplitude	amplitude de pré-emphase
Pre-Emphasis	Pré-Emphase
time	durée

**Figure 19 – Exemple de forme d'onde d'une impulsion au niveau de l'émetteur EMD**  
(suite)

#### 4.5.9.2 EMD – bruit de l'émetteur

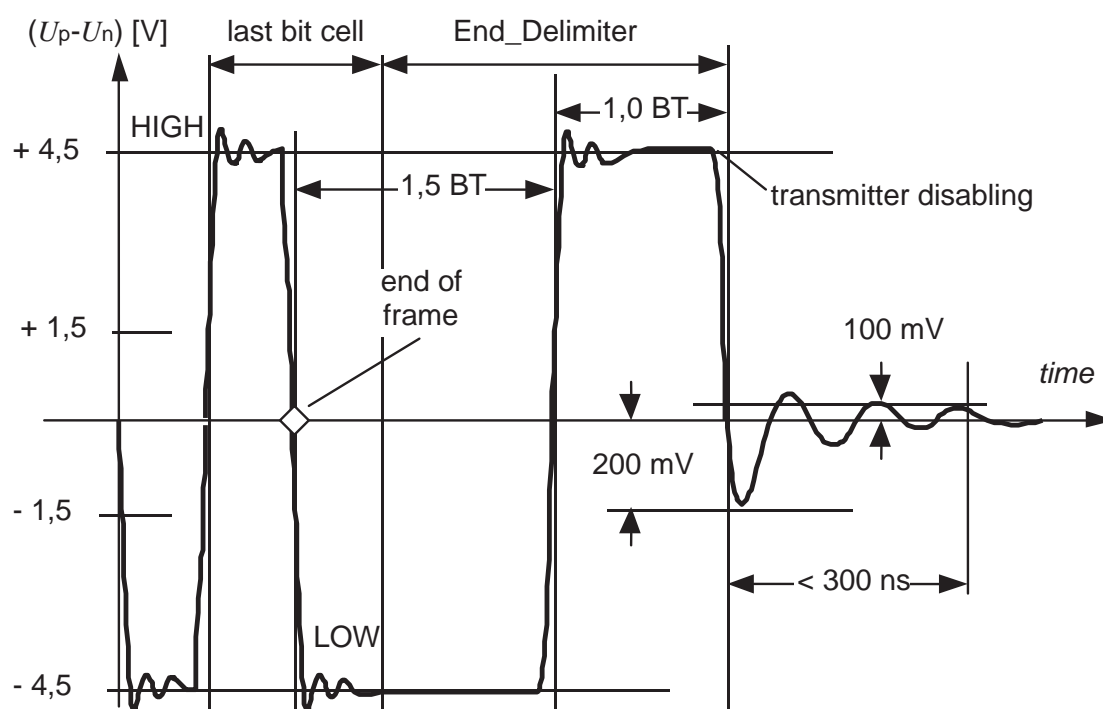
Tout bruit généré par un émetteur qui n'est pas en train d'émettre ne doit pas dépasser une valeur de 5,0 mV<sub>eff</sub> sur la plage de fréquences comprise entre 1,0 kHz et 4,0 BR.

#### 4.5.9.3 EMD – fin de trame de l'émetteur

Après le Délimiteur de Fin de Trame, le signal de sortie doit atteindre 0,10 V dans un délai de 0,30 µs et ne doit à aucun moment, après le premier passage par zéro, dépasser 0,20 V dans les conditions d'essai suivantes:

- l'émetteur émet la trame la plus longue possible;
- les bits de données de la trame sont une séquence aléatoire de "1" et de "0";
- la trame est fermée par le Délimiteur de Fin de Trame comme spécifié en 5.1.6;
- l'émetteur commande le circuit d'essai à vide;
- l'amplitude du signal de sortie est supérieure à 4,5 V avant que l'émetteur ne soit désactivé.

EXEMPLE La Figure 20 montre la fin de la trame lorsque le dernier bit de données est un "1".



**Légende**

Anglais	Français
last bit cell	dernière cellule de bit
transmitter disabling	désactivation de l'émetteur
end of trame	fin de trame
time	durée

**Figure 20 – Exemple de fin de trame EMD**

NOTE L'oscillation de la ligne après désactivation de l'émetteur peut être réduite au minimum en équilibrant les composants de courant continu dans chaque cellule de bit.

Le dernier passage par zéro avant le Délimiteur de Fin de Trame doit définir la fin de trame.

#### 4.5.9.4 EMD – tolérance de l'émetteur aux défaillances

Un émetteur, qu'il soit actif ou non, doit tolérer la mise en œuvre du circuit d'essai de court-circuit au point de connexion jusqu'à ce que la stabilité thermique soit atteinte, et doit reprendre un fonctionnement normal après suppression du circuit d'essai de court-circuit.

Le courant de court-circuit ne doit pas dépasser 1,0 A.

NOTE Pour les essais de conformité, il convient d'atteindre la stabilité thermique après 1 h.

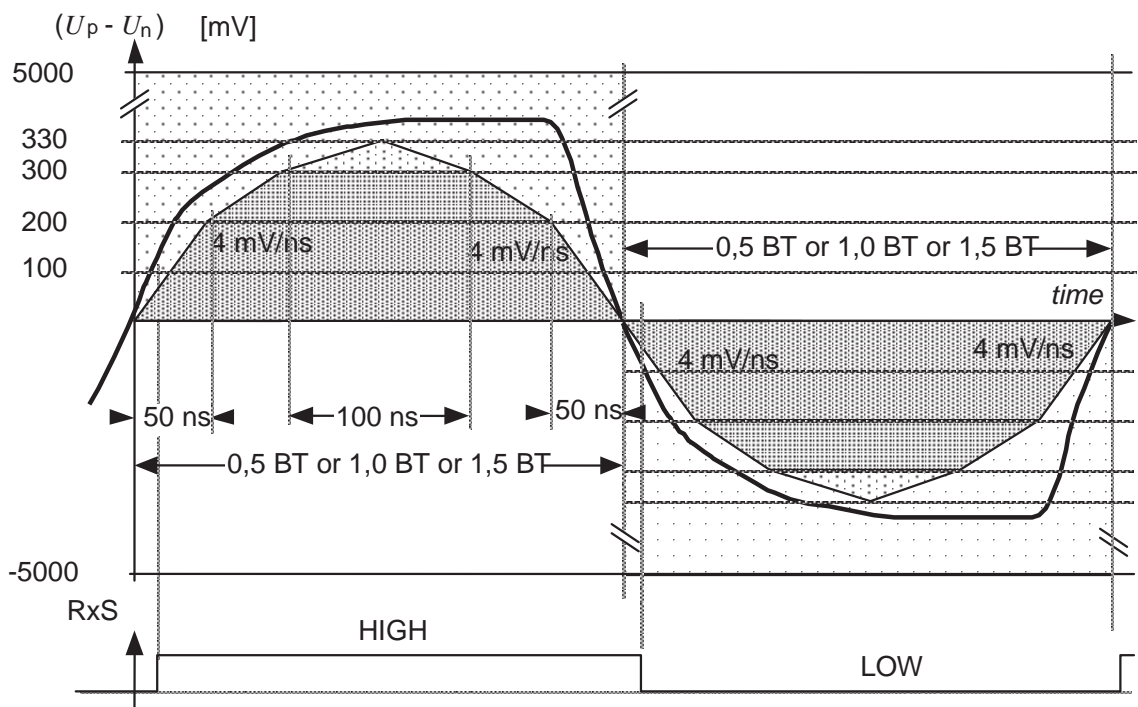
#### 4.5.10 EMD – spécifications du récepteur

##### 4.5.10.1 EMD – signal d'essai du récepteur (indications)

Les caractéristiques d'un récepteur sont vérifiées par essai en appliquant un signal d'essai aux points de connexion et en observant les trames reçues, soit par accès direct au dispositif, soit par une réponse interne au dispositif, comme la réponse à une trame-maître.

L'essai du récepteur consiste à appliquer au point de connexion un signal d'essai dont la forme est la suivante, comme le montre la Figure 21:

- a) lorsque l'amplitude du signal d'essai est inférieure à 0,20 V, la pente de ce signal est supérieure à 4,0 mV/ns (0,20 V / 50,0 ns);
- b) pour vérifier par essai la sensibilité, le signal d'essai reste au-dessus de 0,30 V pendant une période de temps qui commence 0,1 µs après le passage par zéro précédent et qui dure au moins (0,5 BT – 0,2 µs), ou (1,0 BT – 0,2 µs) et (1,5 BT – 0,2 µs) respectivement, alors que son amplitude crête varie entre 0,33 mV et 5,00 V;
- c) pour vérifier par essai l'insensibilité, l'amplitude du signal d'essai ne dépasse pas une valeur prescrite;
- d) pour vérifier par essai la distorsion du front, le passage par zéro de la tension varie d'une valeur prescrite.



Légende

Anglais	Français
time	durée
or	ou

Figure 21 – Signal d'essai du récepteur EMD

Le récepteur détecte une erreur de trame en tant que silence, collision, trame invalide ou trame incorrecte (taille de trame incorrecte, bits de Données de Trame incorrects ou CRC incorrect).

4.5.10.2 EMD – polarité du récepteur

NOTE Cette spécification ne fait pas partie des essais de conformité, étant donné que l'interface Line\_Unit n'est pas nécessairement exposée. Elle fait partie des essais de type de l'émetteur-récepteur.

Selon l'état de la ligne, un récepteur doit fournir un signal RxS binaire comportant deux états, HIGH et LOW, comme suit:

- a) un niveau HIGH si la tension de ligne différentielle  $(U_p - U_n)$  dépasse +0,200 V;
- b) un niveau LOW si la tension de ligne différentielle  $(U_p - U_n)$  devient inférieure à -0,200 V.

Le récepteur doit avoir une hystérésis d'au moins 50 mV, sans excéder toutefois 0,25 V.

En option, dans les environnements soumis à une perturbation électromagnétique élevée, les exigences a) et b) deviennent:

- c) un niveau HIGH si la tension de ligne différentielle ( $U_p - U_n$ ) dépasse le seuil de réception positif;
- d) un niveau LOW si la tension de ligne différentielle ( $U_p - U_n$ ) devient inférieure au seuil de réception négatif.

Le seuil de réception positif doit se situer dans la plage comprise entre 0,200 V et 0,500 V.

Le seuil de réception négatif doit se situer dans la plage comprise entre –0,200 V et –0,500 V.

Il doit être noté que la longueur maximale du segment peut ne pas être atteinte.

#### 4.5.10.3 EMD – sensibilité du récepteur

Un récepteur qui reçoit des trames comportant 64 bits de données aléatoires, avec un débit de 1 000 trames par seconde, ne doit pas détecter plus de trois erreurs de trame pour  $3 \times 10^6$  trames lorsque l'amplitude du signal d'essai définie en 4.5.10.1 varie entre sa valeur minimale et maximale.

#### 4.5.10.4 EMD – insensibilité du récepteur

Le récepteur ne doit pas détecter de trame valide lorsque l'amplitude du signal d'essai est inférieure à 0,100 V.

NOTE Ce seuil peut être augmenté dans certaines applications.

#### 4.5.10.5 EMD – distorsion du front du récepteur

Un récepteur qui reçoit des trames comportant 64 bits de données aléatoires, avec un débit de 1 000 trames par seconde, ne doit pas détecter plus de trois erreurs de trame pour  $3 \times 10^6$  trames lorsqu'un signal d'essai est appliqué, dont les flancs passent à la tension zéro de manière aléatoire dans un intervalle de  $\pm 10\%$  de 1,0 BT autour de la position de passage prévisible.

#### 4.5.10.6 EMD – rejet du bruit du récepteur

Un récepteur qui reçoit des trames comportant 64 bits de données aléatoires, avec un débit de 1 000 trames par seconde, et avec une amplitude de signal d'essai de 0,700 V (1,400 V crête à crête) ne doit pas détecter plus de trois erreurs de trame pour  $3 \times 10^6$  trames lorsqu'il fonctionne:

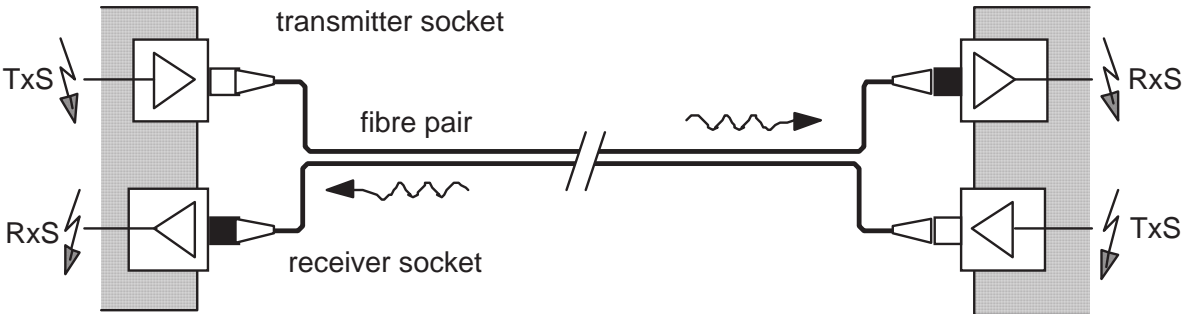
- a) en présence d'un signal sinusoïdal de mode commun (appliqué entre la masse du dispositif et à la fois Data\_P et Data\_N) avec une amplitude de 4,0 V<sub>eff</sub> dans la plage de fréquences comprise entre 65,0 Hz et 1,5 MHz ou
- b) en présence d'un bruit gaussien quasi blanc supplémentaire (appliqué entre Data\_P et Data\_N) réparti sur une largeur de bande comprise entre 1,0 kHz et 4,0 MHz, à une amplitude de 0,140 V<sub>eff</sub>.

### 4.6 Support à fibres optiques (choix)

Les spécifications du support à fibres optiques permettent l'utilisation de fibres optiques mises en bus par des coupleurs en étoile sur des distances allant jusqu'à 2 000 m lorsque le support spécifié est utilisé. La distance réelle est limitée par les distorsions des fibres et des connecteurs, ainsi que par les caractéristiques des répéteurs et des coupleurs en étoile.

4.6.1 OGF – topologie

Le support optique doit consister en une paire de fibres formant une liaison duplex point à point intégrale, comme le montre la Figure 22.



Légende

Anglais	Français
transmitter socket	prise de l'émetteur
fibre pair	paire de fibres
receiver socket	prise du récepteur

Figure 22 – Liaison optique

Les fibres optiques peuvent être connectées de dispositif en dispositif ou mises en bus par un coupleur en étoile, de type actif de préférence (4.6.7).

4.6.2 OGF – câble et fibre optique

La spécification suivante, concernant le câble et la fibre optique, s'applique à un câble monté avec un connecteur à chaque extrémité d'une fibre:

- a) la fibre doit être une fibre en silice multimode à saut d'indice, avec un cœur de 200/230  $\mu\text{m}$  et une erreur d'excentricité de gaine inférieure à 5,0  $\mu\text{m}$ ;
- b) l'atténuation d'un câble de 10,0 m doit être inférieure à 1,8 dB de connecteur à connecteur, mesurée à une longueur d'onde de 840,0 nm  $\pm$  30,0 nm. Pour des distances plus grandes, l'atténuation supplémentaire doit être inférieure à 6,0 dB /km;
- c) le rayon de courbure minimum d'un câble doit être de 50,0 mm;
- d) le câble doit être conforme à la CEI 60571 en ce qui concerne l'essai de vibration;
- e) le câble doit être conforme à la CEI 60794-1-1 (E3, E6, E8) en ce qui concerne la stabilité mécanique;
- f) le câble doit être conforme à la CEI 60332-1-1 en ce qui concerne la résistance au feu;
- g) le câble doit être conforme à la CEI 60245-1 en ce qui concerne la résistance à l'huile;
- h) le câble doit être conforme à la CEI 60304 en ce qui concerne les marquages.

Un même câble peut comporter deux fibres ou plus.

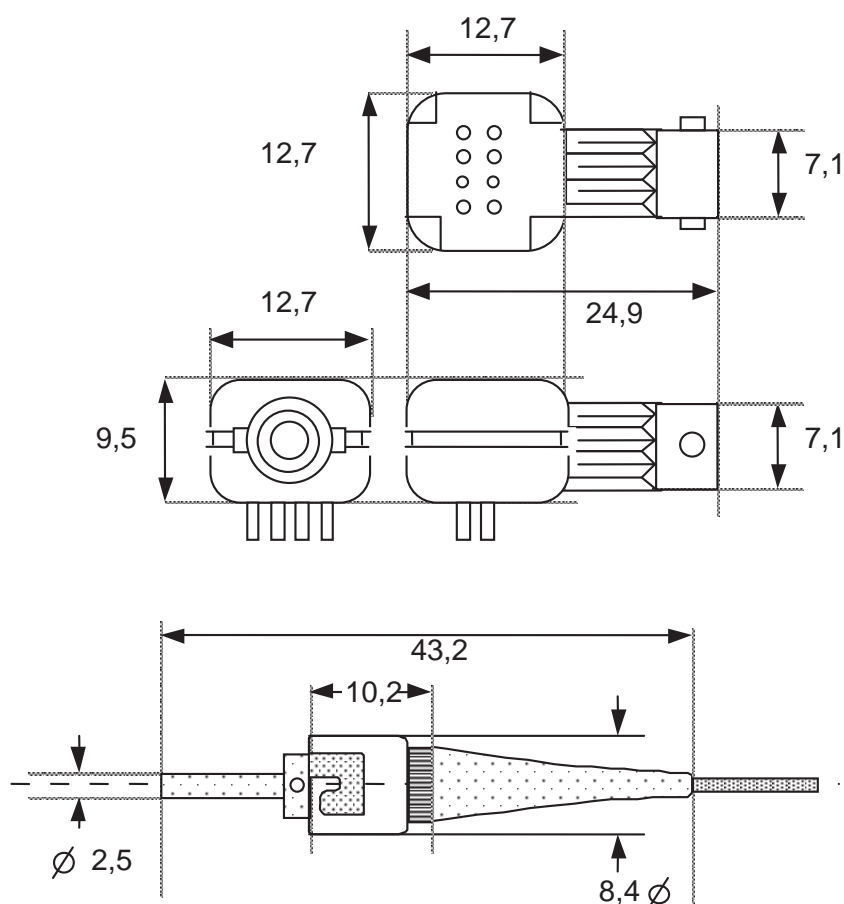
Le codage couleur des bottes est autorisé lors de l'utilisation de câbles multifibres.

NOTE Il est recommandé d'identifier distinctement les deux extrémités d'une fibre, l'une pour le côté émetteur et l'autre pour le côté récepteur.

#### 4.6.3 OGF - interface dépendante du support

Le connecteur (ST-baïonnette) adapté aux fibres simples est spécifié, selon le type de fibre, dans les normes suivantes qui ont été retirées: CEI 60874-10-1, CEI 60874-10-2 et CEI 60874-10-3.

Il convient de préférence que ce composant comprenne l'émetteur et le récepteur, comme le montre la Figure 23.



**Figure 23 – Connecteur optique (dimensions en mm)**

Le connecteur de l'émetteur et celui du récepteur doivent être identifiés de manière appropriée, de préférence comme suit:

- gris clair pour l'émetteur;
- gris foncé pour le récepteur.

NOTE Les composants et les outils du connecteur ST et de la fibre peuvent être utilisés pour un assemblage à fibres multiples. Le connecteur (ST-baïonnette) adapté aux fibres simples est spécifié, selon le type de fibre, dans les normes suivantes qui ont été retirées: CEI 60874-10-1, CEI 60874-10-2 et CEI 60874-10-3.

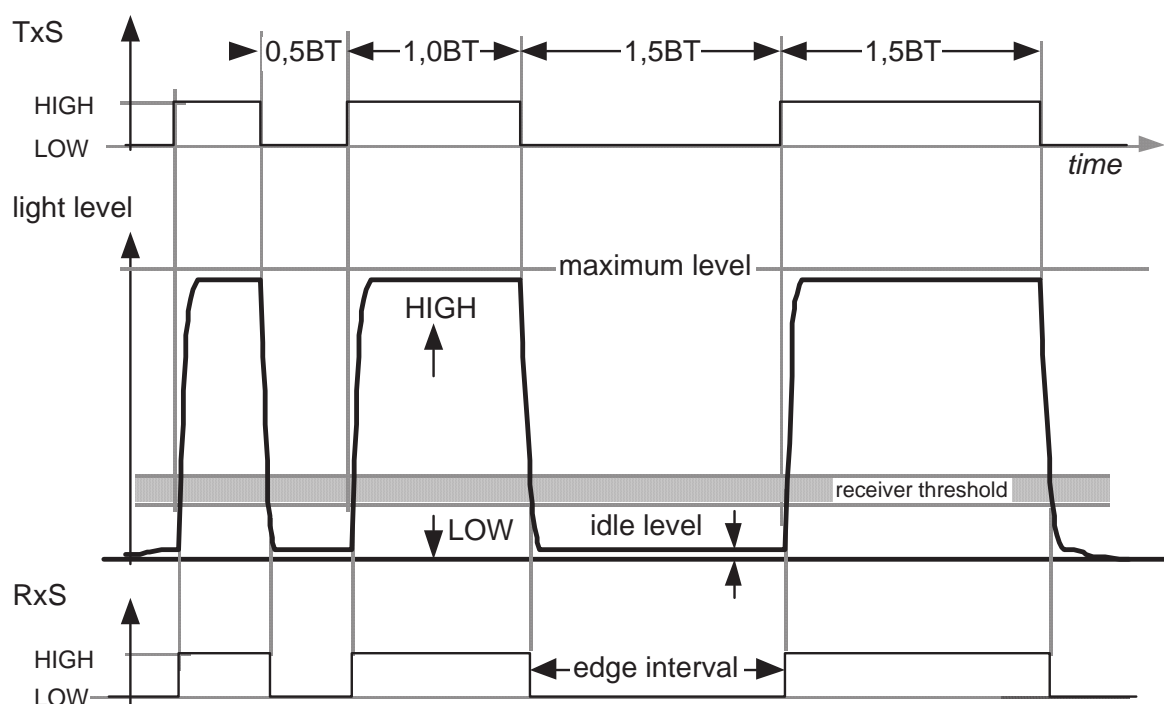
#### 4.6.4 OGF – signal d'essai (indications)

La transmission binaire utilise deux niveaux optiques, le niveau d'énergie le moins élevé étant appelé LOW et le niveau d'énergie le plus élevé étant appelé HIGH.

Le niveau LOW correspond au niveau LOW sur RxS ou TxS, le niveau HIGH correspond au niveau HIGH sur RxS ou TxS.



Le signal d'essai est une trame bien formée qui consiste en des impulsions d'amplitude variable avec des fronts situés à une distance de 0,5 BT, 1,0 BT et 1,5 BT et une gigue réglable entre deux fronts consécutifs, comme le montre la Figure 24.



#### Légende

Anglais	Français
time	durée
light level	niveau de lumière
maximum level	niveau maximum
receiver threshold	seuil de récepteur
idle level	niveau de repos
edge interval	Intervalle entre les fronts

Figure 24 – Exemple de début de trame OGF

### 4.6.5 OGF – spécifications de l'émetteur

#### 4.6.5.1 OGF – niveaux optiques

L'émetteur optique doit être conforme aux spécifications suivantes:

- l'émetteur doit transmettre deux niveaux optiques distincts;
- la longueur d'onde d'émission de crête doit être de  $840,0 \text{ nm} \pm 30,0 \text{ nm}$ ;
- la largeur spectrale doit être inférieure à  $40,0 \text{ nm}$ ;
- la puissance crête de sortie, mesurée après  $1,0 \text{ m}$  de fibre et correspondant à l'état HIGH, doit être comprise entre  $-7,5 \text{ dBm}$  et  $-4,5 \text{ dBm}$ ;
- la puissance minimale, correspondant à l'état LOW, doit être inférieure à  $-40,0 \text{ dBm}$ ;
- le temps de montée des signaux pour passer de LOW à HIGH (10 % à 90 %) doit être inférieur à  $10,0 \text{ ns}$ ;
- le temps de descente des signaux pour passer de HIGH à LOW (90 % à 10 %) doit être inférieur à  $10,0 \text{ ns}$ ;

- h) la différence entre le temps de descente et le temps de montée doit être inférieure à 10,0 ns.

#### 4.6.5.2 OGF – distorsion du front de l'émetteur

La distorsion du front du signal de sortie, définie comme la différence de temps entre les transitions théoriques et réelles, ne doit pas dépasser 10,0 ns sur une trame entière.

#### 4.6.5.3 OGF – fin de trame

L'émetteur doit s'éteindre après le Délimiteur de Fin de trame, tel que spécifié en 5.1.6.

#### 4.6.6 OGF – spécifications du récepteur

##### 4.6.6.1 OGF – récepteur

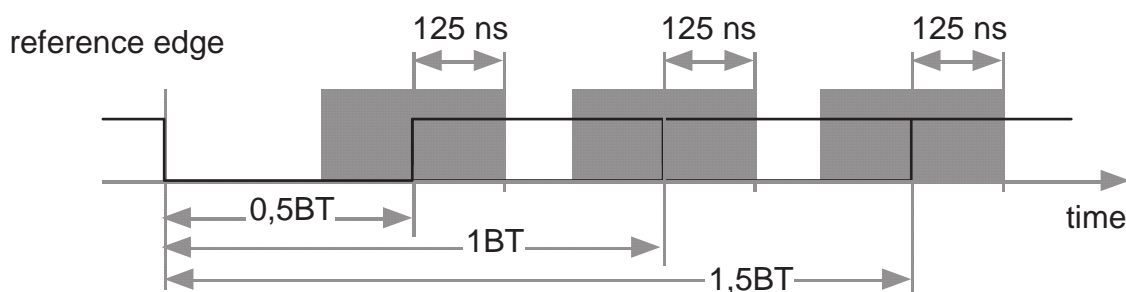
NOTE L'essai de ces spécifications présuppose que le côté électrique du récepteur est accessible. Si le récepteur optique fait l'objet d'un essai de type conformément à cette spécification, cet essai peut être omis lors des essais de conformité du dispositif.

Le récepteur doit être conforme aux spécifications suivantes:

- la puissance d'entrée correspondant à un signal LOW doit être  $< -40,0$  dBm ( $0,1 \mu\text{W}$ ) à 840,0 nm;
- la puissance d'entrée correspondant à un signal HIGH doit être  $> -24,0$  dBm ( $4,0 \mu\text{W}$ ) à 840,0 nm;
- le récepteur doit fonctionner correctement jusqu'à un niveau de lumière de  $-10,0$  dBm ( $63,0 \mu\text{W}$ );
- le temps de montée du signal de sortie (LOW à HIGH) doit être inférieur à 10,0 ns;
- le temps de descente du signal de sortie (HIGH à LOW) doit être inférieur à 10,0 ns;
- la différence entre le temps de descente et le temps de montée doit être inférieure à 10,0 ns.

##### 4.6.6.2 OGF – distorsion du front du récepteur

Le récepteur doit reconnaître une trame correctement même si le front suivant apparaît 0,125  $\mu\text{s}$  trop tard ou trop tôt par rapport à la position prévisible du front, comme le montre la Figure 25.



#### Légende

Anglais	Français
reference edge	front de référence
time	durée

Figure 25 – Gigue frontale

4.6.7 OGF – coupleur en étoile actif

Il est préférable de réaliser un bus de fibres optiques à l'aide d'un coupleur en étoile actif, comme le montre la Figure 26.

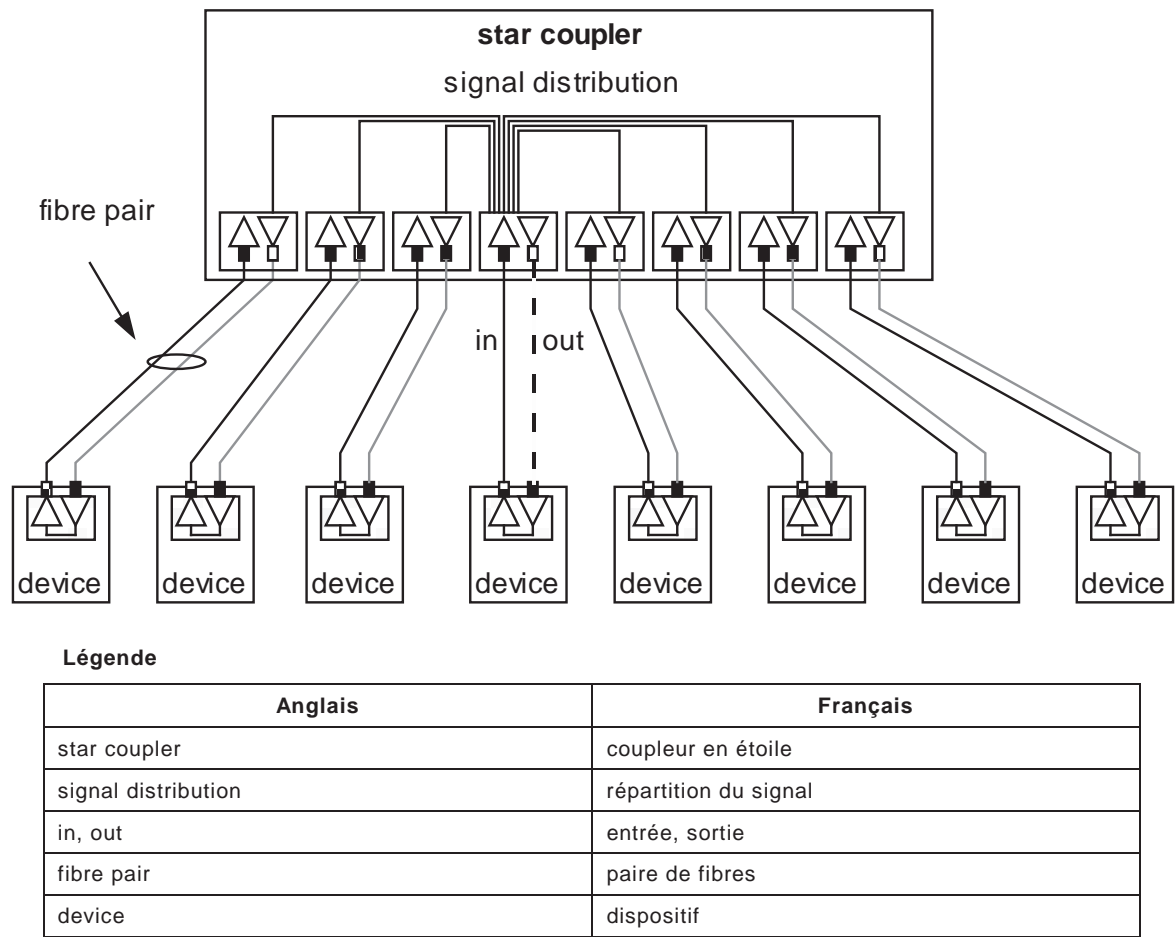


Figure 26 – Exemple de coupleur en étoile actif

NOTE Lorsque le nombre de dispositifs raccordés est peu élevé, il est possible de réaliser un bus de fibres au moyen d'étoiles ou d'embranchements passifs, à condition que la plage dynamique du récepteur le permette.

Un coupleur en étoile actif doit redistribuer sur toutes les sorties un signal lumineux reçu, à l'exception de la sortie correspondant au signal reçu.

L'entrée et les sorties du coupleur en étoile actif doivent être conformes aux niveaux de lumière exigés en 4.6.5 et 4.6.6.

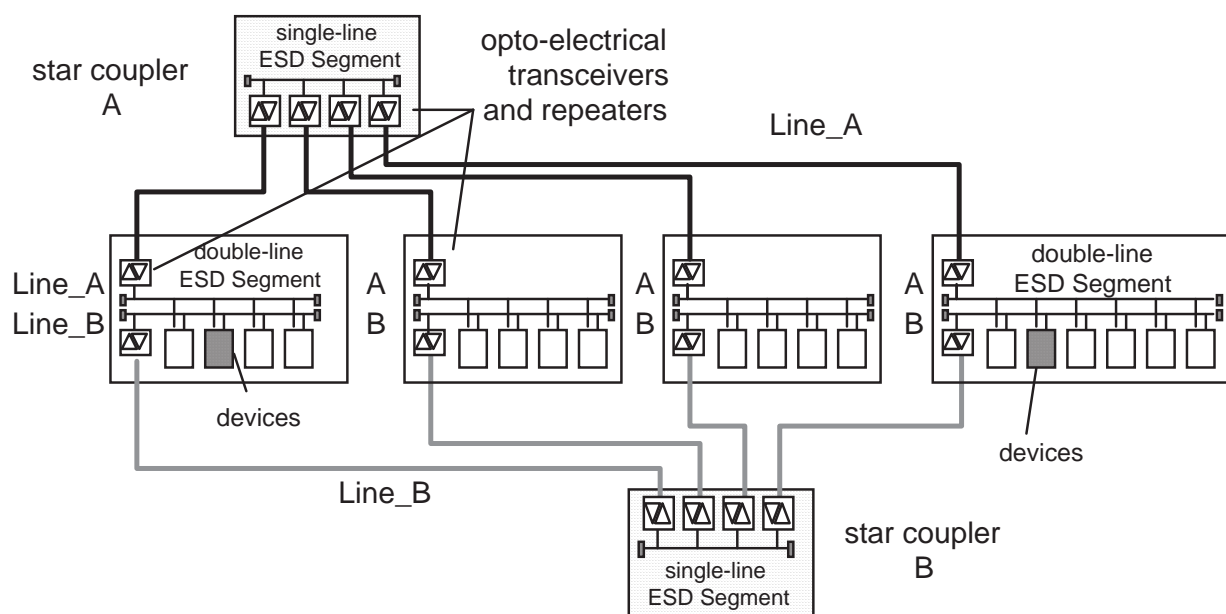
Le temps maximum entre une variation d'un signal d'entrée et la variation correspondante des signaux de sortie doit être de deux temps de répéteur (2,0 T\_repeat).

4.6.8 OGF - configuration à ligne double (option)

Dans le cas d'un segment optique à ligne double, un coupleur en étoile doit être connecté à la Line\_A ou à la Line\_B, mais pas aux deux en même temps.

NOTE Ceci exclut l'insertion de dispositifs sur un coupleur en étoile utilisé dans un segment à ligne double.

EXEMPLE La Figure 27 montre une configuration redondante avec fibres optiques.



#### Légende

Anglais	Français
star coupler	coupleur en étoile
single-line ESD segment	segment ESD à ligne simple
opto-electrical transceivers and repeaters	émetteurs-récepteurs et répéteurs optoélectriques
double-line ESD segment	segment ESD à ligne double
devices	dispositifs

Figure 27 – Exemple de coupleur en étoile dupliqué

## 5 Signalisation dépendant du support

Le présent paragraphe spécifie la signalisation, qui est commune aux supports électriques et optiques. Les différences liées au support sont spécifiées.

### 5.1 Codage et décodage des trames

#### 5.1.1 Conventions

Le codage et le décodage d'une trame supposent un support binaire assumant deux niveaux distincts appelés HIGH et LOW.

L'émetteur impose ces niveaux sur la ligne conformément à son signal émis TxS et le récepteur les détecte comme le signal reçu RxS. La relation entre le signal TxS, les niveaux de ligne et le signal RxS est définie pour chaque support.

Les spécifications suivantes considèrent les signaux TxS et RxS comme des niveaux de ligne théoriques.

#### 5.1.2 Codage des bits

Les différents bits de données, "1" et "0", doivent être codés dans une cellule de bit de la manière suivante, comme le montre la Figure 28:

- Un "1" doit être codé par un niveau HIGH au cours de la première moitié d'une cellule de bit, suivi par un niveau LOW au cours de la seconde moitié d'une cellule de bit;

- b) Un "0" doit être codé par un niveau LOW au cours de la première moitié d'une cellule de bit, suivi par un niveau HIGH au cours de la seconde moitié d'une cellule de bit.

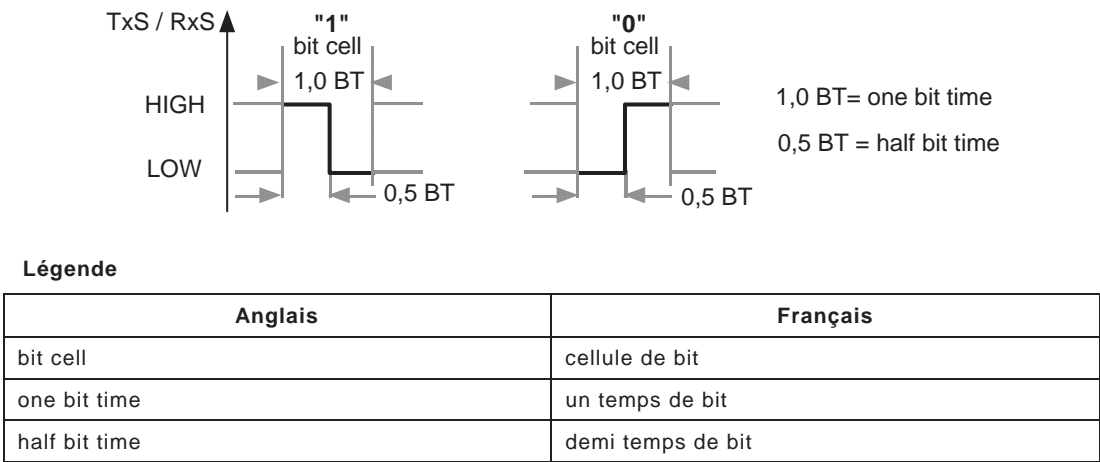


Figure 28 – Codage des données "0" et "1"

NOTE Ce codage est connu sous le nom de codage Manchester ou codage Bi-phase-L. Il n'existe pas de convention générale concernant la polarité du signal.

5.1.3 Symboles de non-données

Les symboles Non\_Data, NH et NL, doivent être codés dans une cellule de bit de la manière suivante, comme le montre la Figure 29:

- a) Un "NH" doit être codé par un niveau HIGH au cours d'une cellule de bit.
- b) Un "NL" doit être codé par un niveau LOW au cours d'une cellule de bit.

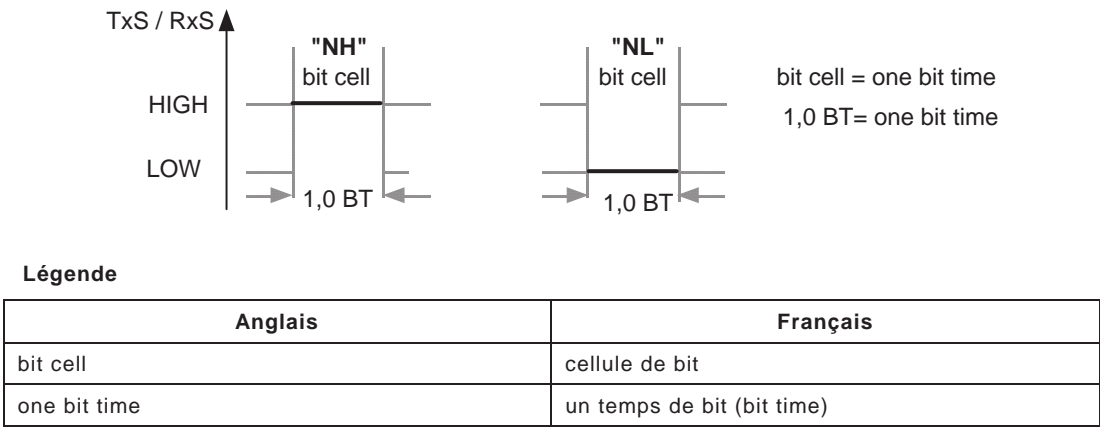


Figure 29 – Codage des symboles Non\_Data

5.1.4 Bit de Début

L'émetteur doit commencer une trame avec un Start Bit (Bit de Début de Trame), qui doit être émis comme symbole "1".

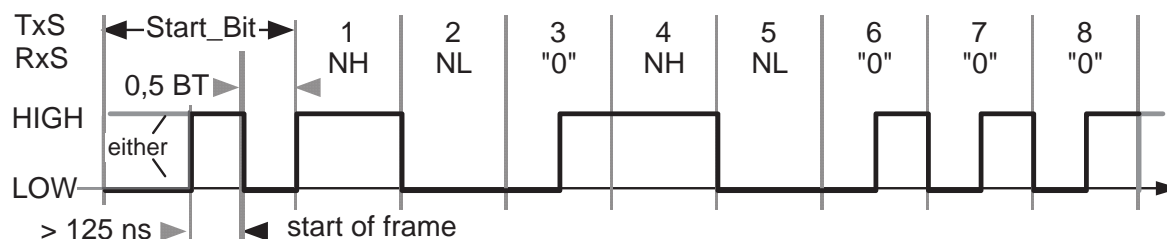
NOTE 1 Le temps de bit de début de trame dépend du support.

NOTE 2 Le bit de début de trame peut être reçu avec une forme ou un temps différent en raison d'effets transitoires et de la définition du niveau de repos.

5.1.5 Délimiteur de Début de Trame

Les Trames-mâîtres doivent commencer par un Délimiteur de Début de Trame, appelé Délimiteur de Début de Trame-Mâitre, qui consiste en la séquence:

{Bit de Début de Trame, "NH", "NL", "0", "NH", "NL", "0", "0", "0"}, comme le montre la Figure 30.



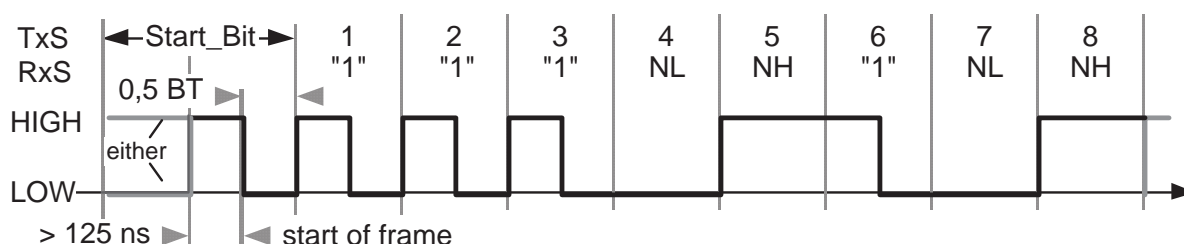
Légende

Anglais	Français
either	ou
start of frame	début de trame

Figure 30 – Délimiteur de Début de Trame-Maître

Les Trames-Esclaves doivent commencer par un Délimiteur de Début de Trame, appelé Délimiteur de Début de Trame-Esclave, qui consiste en la séquence:

{Bit de Début de Trame, "1", "1", "1", "NL", "NH", "1", "NL", "NH"}, comme le montre la Figure 31.



Légende

Anglais	Français
either	ou
start of frame	début de trame

Figure 31 – Délimiteur de Début de Trame-Esclave

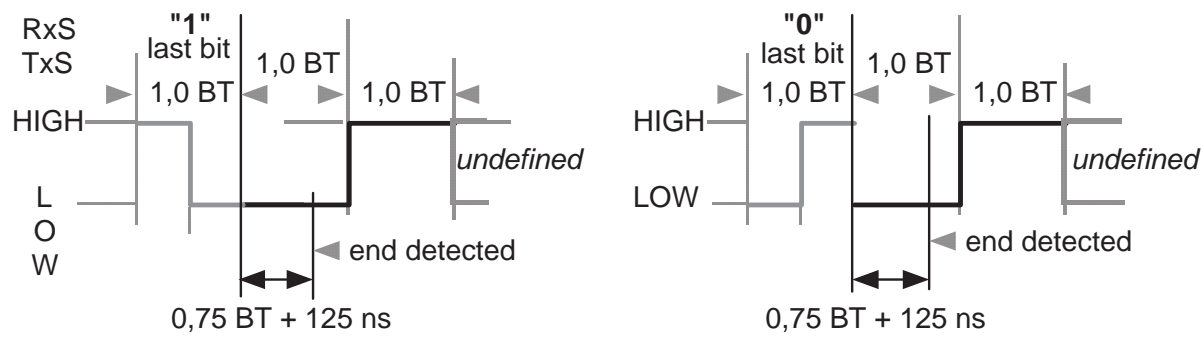
### 5.1.6 Délimiteur de Fin de Trame

L'émetteur doit, après la dernière cellule de bit d'une trame, émettre un Délimiteur de Fin de Trame, dépendant du support:

- dans le cas du support ESD, en ajoutant un symbole 'NL' puis en cessant la transmission comme spécifié en 4.4.8.4;
- dans le cas du support EMD, en ajoutant un symbole "NL" suivi d'un symbole "NH" puis en cessant la transmission comme spécifié en 4.5.9.3;
- dans le cas du support OGF, en ajoutant un symbole "NL" puis en cessant la transmission.

NOTE La spécification du Délimiteur de Fin de Trame dépend du support.

EXEMPLE La Figure 32 montre un Délimiteur de Fin de Trame pour le support EMD.



**Légende**

Anglais	Français
last bit	dernier bit
undefined	non défini
end detected	fin détectée

Figure 32 – Exemple de Délimiteur de Fin de Trame dans le cas du support EMD

Le décodeur doit considérer qu'un Délimiteur de Fin de Trame est présent lorsque le signal RxS reste au niveau LOW pendant un temps supérieur à 0,75 BT + 0,125 µs.

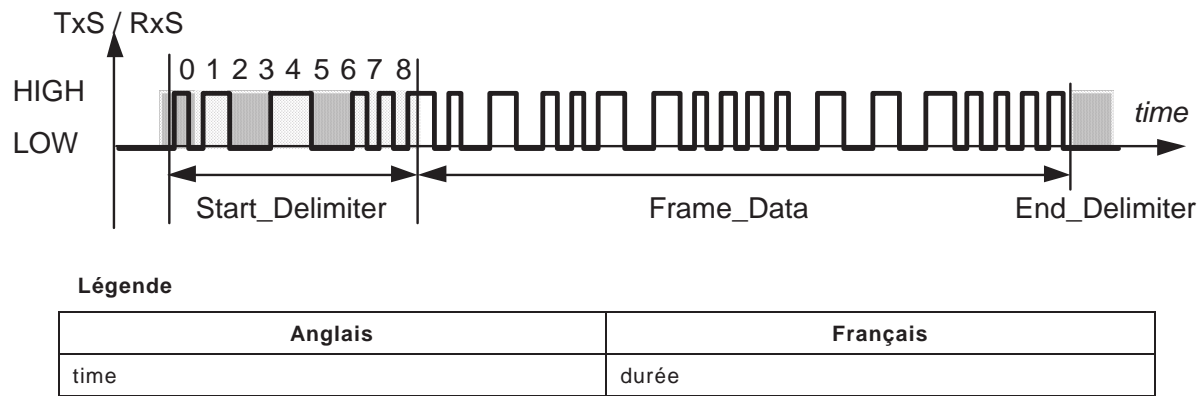
Un décodeur doit ignorer toutes impulsions éventuelles, après la fin de la trame, jusqu'à ce qu'il détecte le prochain Délimiteur de Début de Trame.

5.1.7 Trame valide (définition)

Un décodeur doit reconnaître une trame valide consistant en:

- a) un Délimiteur de Début de Trame (5.1.5); suivi de
- b) des Données de Trame sous forme de symboles "0" et "1" (5.1.2); et suivies par
- c) un Délimiteur de Fin de Trame (5.1.6).

EXEMPLE La Figure 33 montre une trame valide typique avec 16 bits de données utiles.



**Légende**

Anglais	Français
time	durée

Figure 33 – Exemple de trame valide (support OGF)

Le décodeur peut détecter la polarité correcte des signaux en décodant le Délimiteur de Début de Trame, mais il peut ne pas inverser automatiquement le signal si la polarité est fausse.

Afin de définir la cadence:

- le début de la trame est défini comme la transition de HIGH à LOW du Bit de Début de Trame;
- la fin de la trame est définie comme la dernière transition de HIGH à LOW ou de LOW à HIGH avant le Délimiteur de Fin de Trame.

NOTE 1 Le décodeur peut vérifier par ailleurs que le nombre de bits de données est plausible.

NOTE 2 Dans la mesure où le décodeur de la Trusted\_Line et celui de la Observed\_Line peuvent vérifier des critères différents, ils peuvent interpréter une trame différemment.

### 5.1.8 Détection de ligne au repos

Le décodeur doit pouvoir détecter que la ligne est au repos.

NOTE Le décodeur est censé distinguer une quantité limitée de bruit parasite (par exemple, provenant de réflexions) de l'absence de transmission.

### 5.1.9 Détection de collision

Le décodeur doit pouvoir distinguer entre une ligne au repos et une collision.

NOTE Le décodeur n'est pas censé distinguer un bruit parasite durable d'une collision.

### 5.1.10 Comportement du récepteur en cas d'erreur

Si le décodeur détecte une collision, il doit ignorer toutes les trames sur cette ligne jusqu'à la réception du prochain Master Frame\_Delimiter.

### 5.1.11 Détection de bavardage

Un dispositif peut comprendre un moyen permettant de détecter que son émetteur est actif pendant un temps plus long qu'un délai configurable  $T_{jabber}$ , et un moyen permettant de désactiver l'émetteur dans ce cas.

Dans ce cas, la temporisation  $T_{jabber}$  doit être réglée sur la durée de la trame la plus longue que le dispositif est capable de transmettre dans des conditions normales de fonctionnement plus 10 %.

La détection de bavardage doit permettre une nouvelle transmission après la suivante.

## 5.2 Redondance de ligne (option)

NOTE Le présent paragraphe spécifie l'utilisation de lignes redondantes pour le cas où cette option est utilisée.

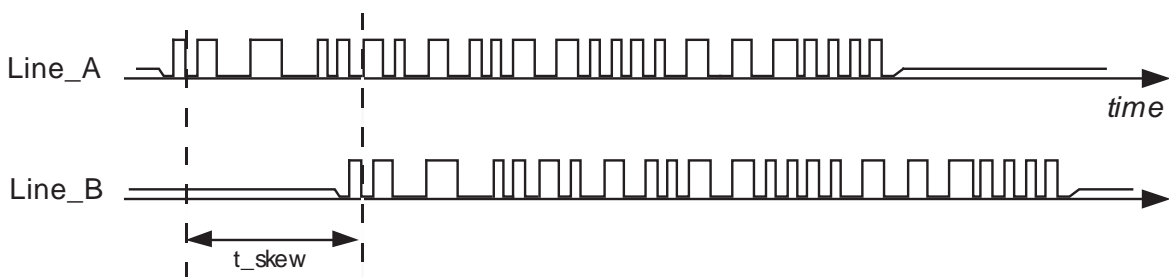
### 5.2.1 Principe

La redondance de ligne repose sur le principe qu'un dispositif transmet simultanément les mêmes données sur les deux lignes Line\_A et Line\_B, et qu'un dispositif accepte les données provenant d'une seule ligne, appelée Trusted\_Line, tout en surveillant l'autre ligne, appelée Observed\_Line.

Chaque dispositif choisit sa Trusted\_Line et son Observed\_Line sur la base des signaux générés par sa couche physique, ou à la demande de sa Couche de liaison, indépendamment des autres dispositifs.

Le décalage  $t_{skew}$  est défini comme la différence de temps entre les signaux sur les deux lignes, mesurée à l'émetteur, au récepteur, ou en tout point sur la ligne, comme le montre la Figure 34.





Légende

Anglais	Français
time	durée

Figure 34 – Décalage du signal

5.2.2 Transmission redondante

5.2.2.1 Fonctionnement

Un dispositif avec raccordement à ligne double doit émettre une trame sur les deux lignes redondantes Line\_A et Line\_B de sorte que le décalage des deux signaux de sortie ne dépasse pas  $T_{skew\_t} = 0,100 \mu s$ .

5.2.2.2 Décalage de propagation

La configuration du bus doit garantir que le décalage entre les signaux reçus sur Line\_A ou sur Line\_B ne dépasse pas  $T_{skew} = 8,0 \mu s$  en tout point du bus.

5.2.3 Réception redondante

Un dispositif ne doit recevoir les données que de la Trusted\_Line et doit surveiller en permanence la Observed\_Line.

Un récepteur doit fonctionner correctement et ne signaler aucune erreur s'il reçoit des trames valides sur les deux lignes avec un décalage inférieur à  $T_{skew\_r} = 8,0 \mu s$ .

5.2.4 Commutation

Un dispositif doit intervertir la Trusted\_Line et la Observed\_Line dans les conditions suivantes:

- a) si aucune trame valide n'a été reçue de la Trusted\_Line pendant  $T_{switchover} = 1,4$  ms depuis la fin de la dernière Trame-Maître valide ou depuis la dernière commutation;
- b) si le décodeur de la Observed\_Line détecte une trame valide et le décodeur de la Trusted\_Line ne détecte pas de trame valide dans l'intervalle  $T_{skew\_r}$ , alors qu'aucune collision n'est prévisible sur cette trame (voir 6.3.2), et après que la Observed\_Line a été au repos pendant plus de 2,0 BT, et à condition que le bit Redundant\_Line\_Disturbed ne soit pas mis à «1» (5.2.5);
- c) si une trame valide a été reçue sur la Trusted\_Line, mais sa longueur ou le Code Détecteur d'Erreur est incorrect(e), à condition que le bit Redundant\_Line\_Disturbed ne soit pas mis à «1» (facultatif);
- d) si un dispositif répond à une Device\_Status\_Request avec une Device\_Status\_Response, à condition que le bit Redundant\_Line\_Disturbed ne soit pas mis à «1», cette condition étant obligatoire pour les dispositifs de Classe 1 et une option de configuration pour les autres Classes;
- e) si un dispositif reçoit une requête de sa Couche de Liaison, cette condition étant une option de configuration non applicable aux dispositifs de Classe 1.

NOTE 1 La temporisation  $T_{\text{switchover}}$  (1,4 ms) est liée à la distance maximale entre trames-maîtres  $T_{\text{alive}}$  (1,3 ms).

NOTE 2 Une fois qu'une commutation a lieu, un retour sur la ligne antérieure n'est possible que lorsque le bit  $\text{Redundant\_Line\_Disturbed}$  a été remis à "0" par une  $\text{Device\_Status\_Response}$  ou par la Couche de liaison de ce dispositif.

NOTE 3 Si un brouilleur continu émet sur la  $\text{Trusted\_Line}$ , il peut s'écouler 1,4 ms jusqu'à ce qu'une commutation forcée ait lieu.

## 5.2.5 Rapport d'état de redondance

Le MAU doit indiquer l'état de la redondance à l'Interface de Supervision de Liaison par l'intermédiaire de deux bits d'état, appelés LAT et RLD:

- LAT ( $\text{Line\_A\_Trusted}$ ) doit être mis à «1» avec la  $\text{Line\_A}$  correspondant à la  $\text{Trusted\_Line}$ ;
- RLD ( $\text{Redundant\_Line\_Disturbed}$ ) doit être:
  - mis à «1» si une commutation a eu lieu ou si le décodeur de la  $\text{Trusted\_Line}$  détecte une trame valide et le décodeur de la  $\text{Observed\_Line}$  ne détecte aucune trame valide décalée de moins de  $T_{\text{skew}_r}$ , alors qu'aucune collision n'est prévisible sur cette trame (voir 6.3.2);
  - remis à "0" après envoi de la  $\text{Device\_Status\_Response}$  ou par la Couche de Liaison (au choix de l'application).

Dans un raccordement à ligne simple, un dispositif doit considérer la ligne non utilisée comme étant continuellement perturbée, RLD étant par ailleurs toujours mis à "1".

Il doit y avoir un compteur d'erreurs pour chaque ligne, qui doit être incrémenté de un par un signal  $\text{Line\_Disturbed}$ , défini comme suit:

Le signal  $\text{Line\_Disturbed}$  doit être:

- mis à "1":
  - si une collision se produit sur la  $\text{Trusted\_Line}$ , mais qu'aucune collision n'est prévisible (voir 6.3.2);
  - si une trame valide est reçue sur la  $\text{Observed\_Line}$  et aucune trame valide n'apparaît sur la  $\text{Trusted\_Line}$  avec un décalage  $T_{\text{skew}_r}$ , alors qu'aucune collision n'est prévisible (voir 6.3.2), et
- remis à "0":
  - par la réception d'une trame valide sur la  $\text{Trusted\_Line}$ .

NOTE Cette définition du compteur d'erreurs empêche que le compteur ne soit rapidement incrémenté en cas de défaillance permanente sur une ligne, par exemple, en cas de rupture de ligne ou si la redondance n'est pas utilisée.

## 5.3 Répéteur

La régénération du signal se produit dans les répéteurs (coupleur optique ou isolé galvaniquement). Les spécifications suivantes ont pour objet de permettre au minimum jusqu'à quatre répéteurs en série.

### 5.3.1 Répéteur placé entre segments à ligne simple

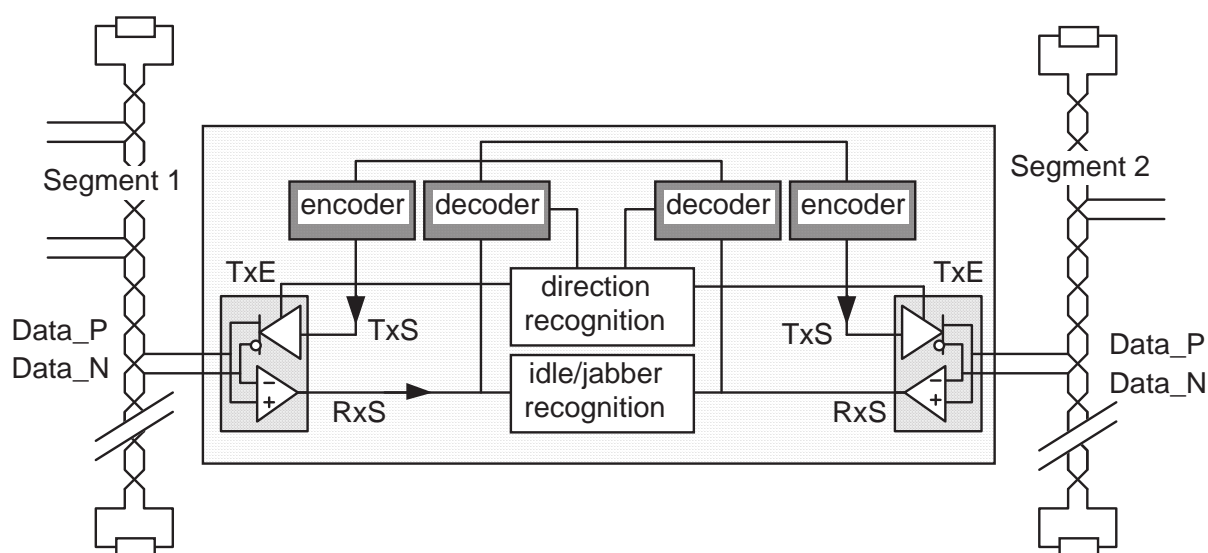
Un répéteur doit être conforme aux spécifications suivantes:

- a) il doit reconnaître la direction initiale comme étant celle de l'impulsion de signal reçue en premier et transmettre le signal dans la direction opposée. Cette direction ne doit pas changer jusqu'à ce que le signal reçu de la direction initiale reste stable pendant 2,0  $\mu\text{s}$ ;
- b) il doit accepter une distorsion des impulsions à l'entrée inférieure à  $\pm 0,125 \mu\text{s}$ ;

- c) il doit régénérer les niveaux physiques de manière transparente, sans interprétation du contenu des trames, les seuls éléments connus étant les fronts, à intervalles de 0,5 BT, 1,0 BT et 1,5 BT;
- d) il doit produire une distorsion des impulsions à la sortie inférieure à 10,0 ns;
- e) si une collision se produit, les fronts mal placés doivent être transmis sans re-synchronisation, les collisions ne doivent toutefois pas être propagées dans la direction initiale;
- f) il doit introduire un délai  $T_{repeat}$  d'au plus 3,0  $\mu s$  entre un front à une entrée et le front correspondant à l'autre sortie, tandis que la différence entre le délai minimum et le délai maximum doit être inférieure à 0,20 BT, sauf si g) s'applique;
- g) il doit retarder une trame qui arrive trop près d'une autre de manière à ce que l'intervalle entre les trames reste au-dessus de la valeur spécifiée en 6.2.3;
- h) il doit reconnaître et isoler un segment dans lequel un émetteur continu est actif pendant un temps supérieur à la temporisation  $T_{jabber\_all}$ , égale à la trame la plus longue que tout dispositif sur ce segment est censé envoyer, plus 15 % (5.1.11).

NOTE La temporisation  $T_{jabber\_all}$  d'un répéteur est 5 % plus longue que le  $T_{jabber}$  d'un dispositif pour garantir qu'un dispositif est capable de s'arrêter de lui-même en cas de défaillance, avant que celle-ci n'affecte le segment.

EXEMPLE La Figure 35 montre le circuit principal d'un répéteur entre deux segments EMD.



Légende

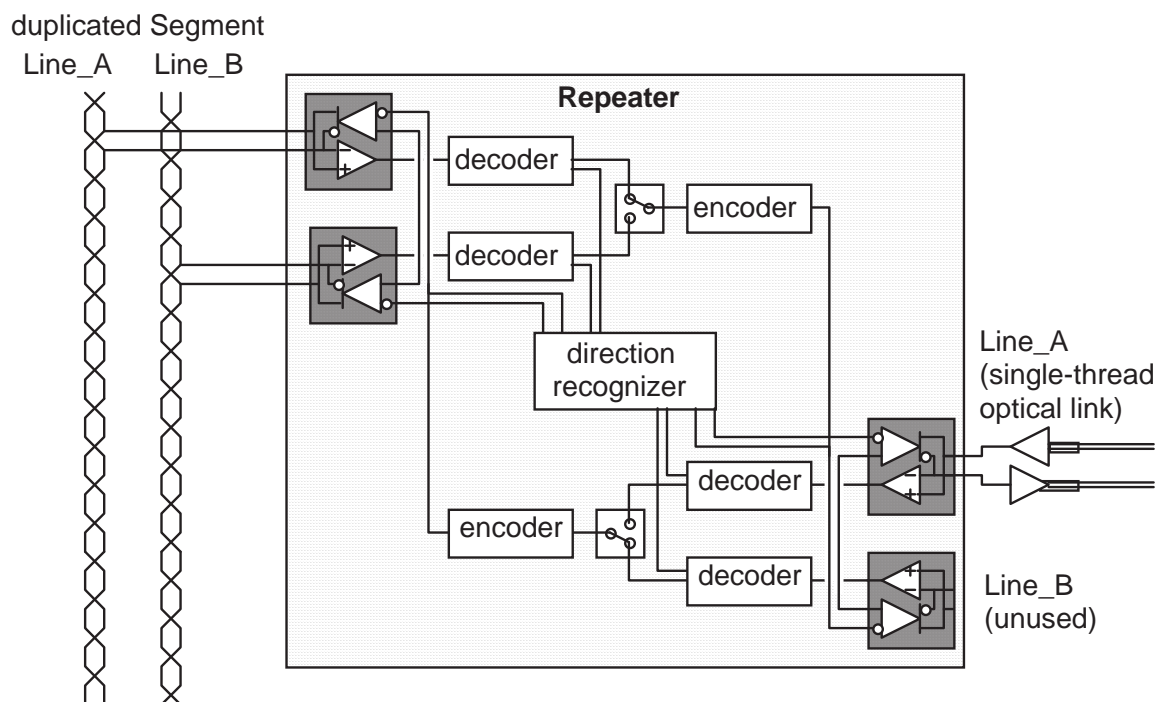
Anglais	Français
segment	segment
encoder	codeur
decoder	décodeur
direction recognition	reconnaissance de direction
idle/jabber recognition	reconnaissance repos/bavardage

Figure 35 – Exemple de répéteur pour un raccordement en ligne simple

### 5.3.2 Répéteur pour support redondant (option)

Si un répéteur est employé pour connecter un segment à ligne simple à un segment à ligne double, il doit être capable de mettre en œuvre la commutation de redondance. Il doit également être possible de configurer une connexion fixe du segment à ligne simple à une ligne du segment à ligne double.

EXEMPLE La Figure 36 présente le schéma d'un répéteur pour support redondant.



#### Légende

Anglais	Français
duplicated Segment	Segment dupliqué (ligne double)
Repeater	Répéteur
decoder	décodeur
encoder	codeur
direction recognizer	dispositif de reconnaissance de direction
single-thread optical link	liaison optique à une ligne (ligne simple)
unused	non utilisée

**Figure 36 – Exemple de répéteur connectant un segment à ligne double à un segment à ligne simple**

## 6 Trames et télégrammes

### 6.1 Format des trames

#### 6.1.1 Format de Trame-Maître

Une Trame-Maître doit, comme le montre la Figure 37:

- commencer par le Délimiteur de Début de Trame-Maître (5.1.5);
- suivi de 16 bits de Données de Trame (7.2.1);
- suivi du Code Détecteur d'Erreur de 8 bits (6.1.3).

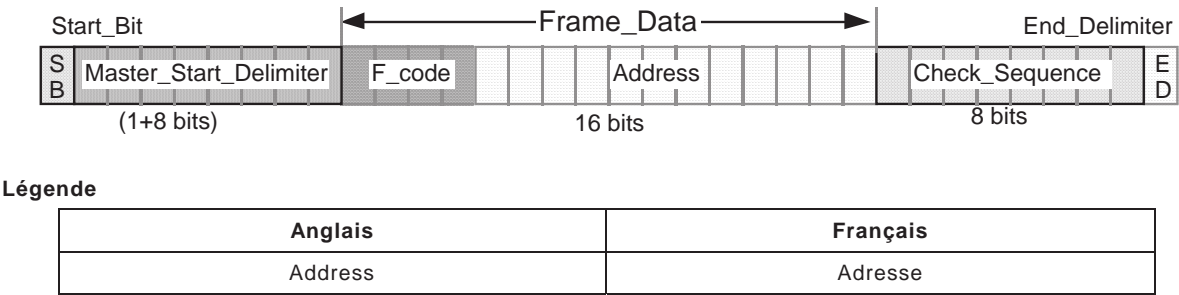


Figure 37 – Format de Trame-Maître

6.1.2 Format de Trame-Esclave

Une Trame-Esclave doit, comme le montre la Figure 38:

- a) commencer par le Délimiteur de Début de Trame-Esclave (5.1.5);
- b) suivi de 16, 32, 64, 128 ou 256 bits de Données de Trame (7.3.1);
- c) inclure un Code Détecteur d'Erreur de 8 bits après chaque mot de 64 bits de données ou être ajoutée à la trame lorsque les Données de Trame correspondent à 16 ou 32 bits de données (6.1.3).

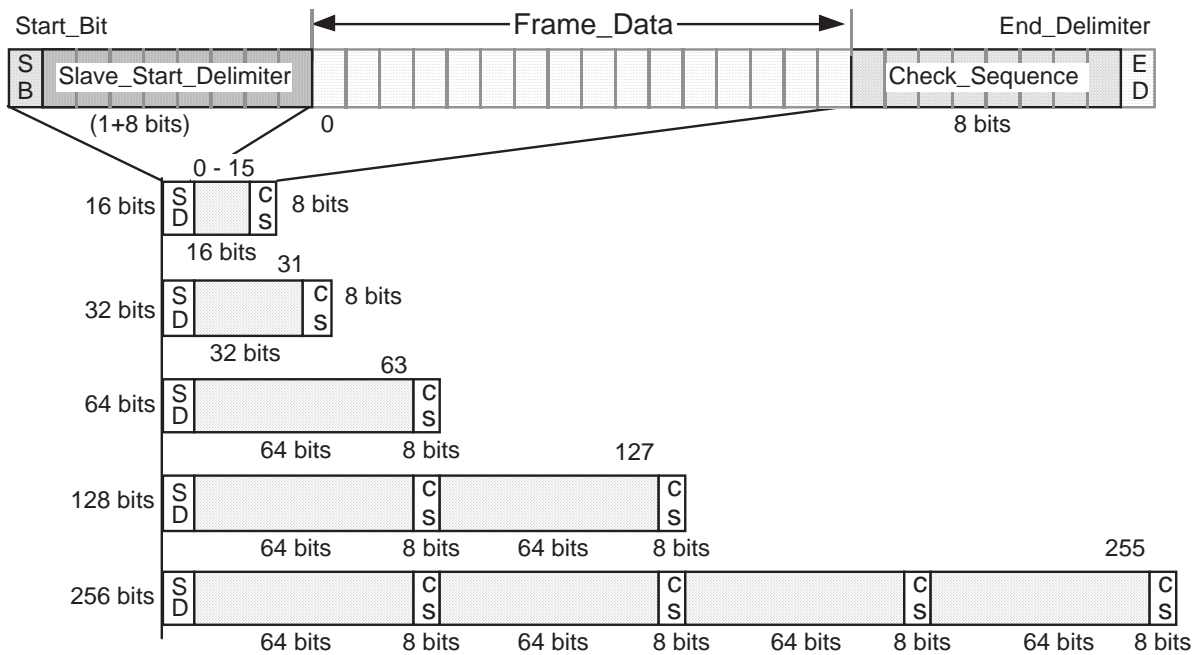


Figure 38 – Trames-esclaves

NOTE 1 Le Bit de Début de Trame (SB) et le Délimiteur de Fin de Trame (ED) sont implicites dans les spécifications suivantes et ont été omis dans les Figures.

NOTE 2 Des trames plus longues peuvent être construites sur le même modèle, c'est-à-dire un Code Détecteur d'Erreur après chaque mot de 64 bits.

6.1.3 Code Détecteur d'Erreur

Les trames doivent être protégées par un ou plusieurs Codes Détecteurs d'Erreur de 8 bits.

Le contenu des données doit être traité comme un mot de code de 64 bits (de 16 et 32 bits pour les trames plus petites), sans tenir compte du Délimiteur de Début de Trame et du Délimiteur de Fin de Trame.

Le mot de code et le Code Détecteur d'Erreur suivant doivent être transmis avec le bit de poids fort en premier.

Le Code Détecteur d'Erreur doit être calculé comme un code de redondance cyclique (CRC) sur les 16, 32 ou 64 bits de données qu'il protège.

L'algorithme du Code Détecteur d'Erreur doit correspondre à la classe de format FT2 de la CEI 60870-5-1, qui est rappelée ici.

Le Code Détecteur d'Erreur doit être calculé selon le polynôme générateur suivant:

$$G(x) = x^7 + x^6 + x^5 + x^2 + 1$$

Le reste de sept bits doit être prolongé par un bit de parité paire.

Tous les huit bits doivent être transmis inversés.

#### EXEMPLE

L'information de 16 bits

0111 1110 1100 0011

multipliée par  $x^7$

0111 1110 1100 0011 000  
0000

divisée par  $x^7 + x^6 + x^5 + x^2 + 1$ :

1110 0101

génère un reste de:

001 0001

qui est prolongé par un bit de parité paire:

001 00010

et transmis inversé comme octet de contrôle

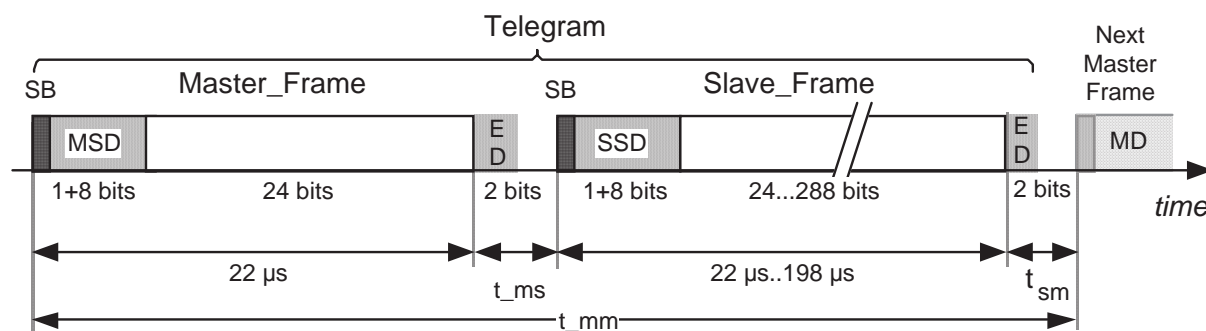
CS = 1101 1101

## 6.2 Cadence des télégrammes

### 6.2.1 Conventions

Un télégramme est défini comme une Trame-Maître suivie de la Trame-Esclave envoyée en réponse.

EXEMPLE La Figure 39 montre la cadence d'un télégramme en un point quelconque du bus.



### Key

MSD Master\_Start\_Delimiter

SSD Slave\_Start\_Delimiter

SB Start\_Bit

$t_{ms}$  time from Master\_Frame to Slave\_Frame sent in response to it.

$t_{sm}$  time from Slave\_Frame to following Master\_Frame.

$t_{mm}$  time from Master\_Frame to Master\_Frame.

### Légende

Anglais	Français
Telegram	Télégramme
Next Master Frame	Trame-Maître suivante
time	durée
Key	Légende
time ... from... to ... in response to it	durée entre ... et ... envoyée en réponse à cette dernière
time... from... to...	durée ... entre ... et ...

**Figure 39 – Cadence d'un télégramme**

L'intervalle entre les trames est compté à partir des points de début et de fin de trame définis pour chaque support.

Dans un segment à ligne double, l'intervalle entre les trames est compté à partir de la fin de la première trame sur l'une des deux lignes, quelle que soit celle qui arrive en dernier, jusqu'au début de la seconde trame, quelle que soit celle qui arrive en premier, et ceci tant que le bit RLD n'est pas à "1".

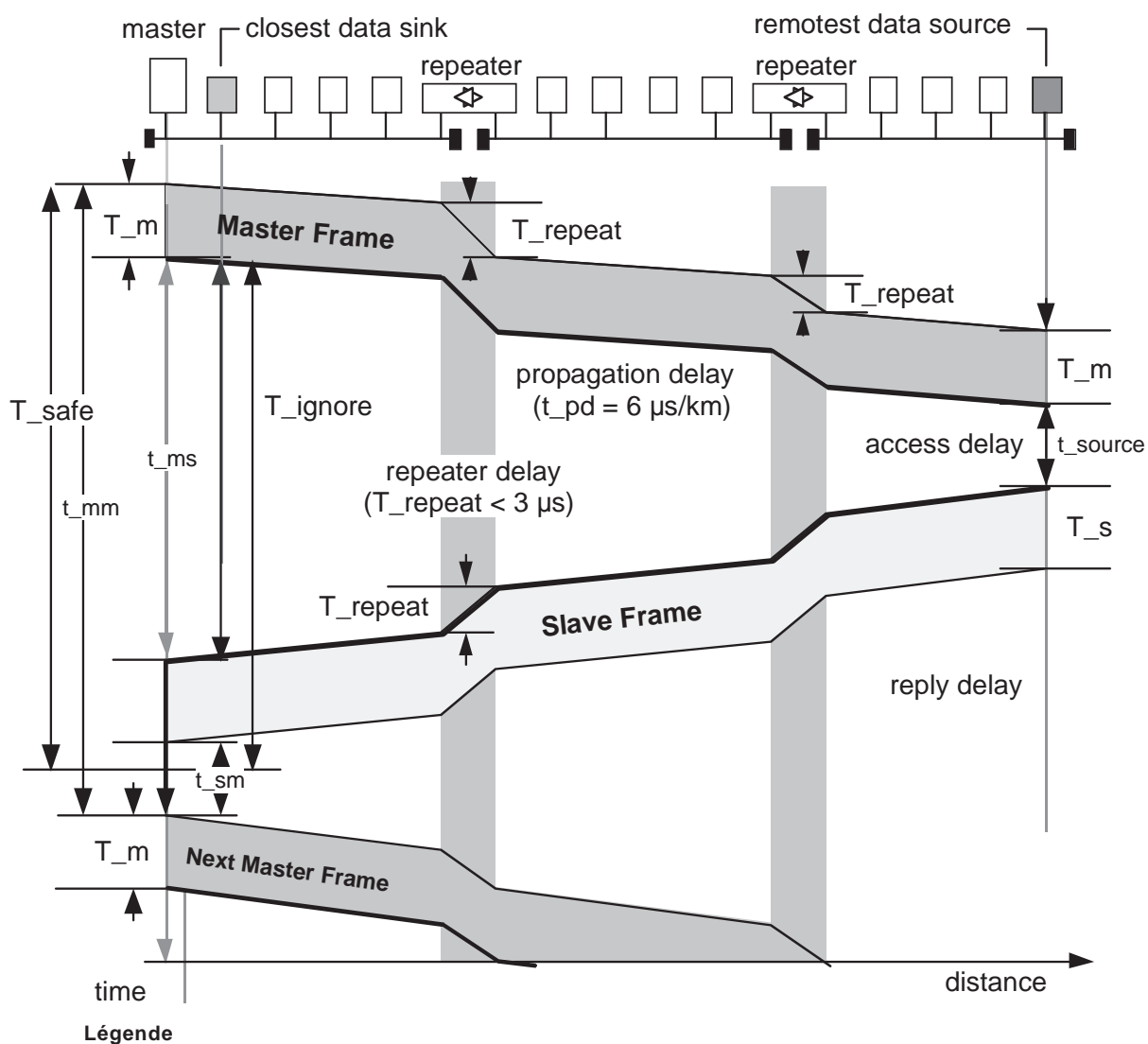
### 6.2.2 Délai de réponse (définition)

Le délai de réponse  $T_{reply}$  est, pour un bus donné, le délai maximum pouvant apparaître entre la fin d'une Trame-Maître et le début de la Trame-Esclave envoyée en réponse, mesuré au niveau du maître.

Le délai de réponse est la somme des délais de propagation, de décodage et d'accès.

$T_{reply}$  est un paramètre de configuration qui indique au maître combien de temps il doit attendre avant d'envoyer la Trame-Maître suivante s'il ne reçoit pas de Trame-Esclave ou s'il s'attend à une collision (voir 6.3.2).

**EXEMPLE** La Figure 40 illustre le délai de réponse dans un bus comprenant 2 répéteurs.



Anglais	Français
master	maître
closest data sink	collecteur de données le plus proche
repeater	répéteur
remotest data source	source de données la plus éloigné
propagation delay	délai de propagation
repeater delay	délai du répéteur
access delay	délai d'accès
Slave Frame	Trame-Esclave
reply delay	délai de réponse
Next Master Frame	Trame-Maître suivante
Master frame	Trame-Maître
Time	durée

Figure 40 – Exemple de délai de réponse



$T_{reply}$ , pour un bus donné, est calculé comme la somme des délais de propagation multipliés par deux (ligne et répéteurs), plus le délai au niveau de l'esclave dans le cas le plus défavorable:

$$T_{reply} [\mu s] = 2 \times (6,0 \times L \times + T_{repeat\_max} \times Nrep) + T_{source\_max}$$

où

6,0  $\mu s/km$  est le délai de propagation dans le cas le plus défavorable, par km, d'une ligne (0,5  $\times$  vitesse de la lumière);

$L$  est la longueur électrique de la ligne en km (qui est supérieure à la distance physique);

$T_{repeat\_max}$  est le délai maximum permis à un répéteur (3,0  $\mu s$  dans une direction);

$Nrep$  est le nombre de répéteurs;

$T_{source\_max}$  (6,0  $\mu s$ ) est le délai maximum à la source (6.2.3).

#### EXEMPLES

$T_{reply} = 6,24 \mu s$  pour un bus de 20,0 m sans répéteur intermédiaire.

$T_{reply} = 44,0 \mu s$  pour un bus de 2,0 km avec deux répéteurs.

La valeur par défaut et minimale de  $T_{reply}$  doit être  $T_{reply\_def} = 42,7 \mu s$ .

NOTE Avec l'option de configuration "Extended Reply Delay",  $T_{reply}$  dépasse la valeur de  $T_{reply\_def}$ .

### 6.2.3 Espacement des trames à la source

Après la fin d'une Trame-Maître, le dispositif émetteur esclave sélectionné doit répondre par une Trame-Esclave comme le montre la Figure 41:

a) au plus tôt après  $T_{source\_min} = 2,0 \mu s$ ; et

b) au plus tard avant  $T_{source\_max} = 6,0 \mu s$ .



#### Légende

Anglais	Français
Slave response within	Réponse de l'esclave dans un délai de
address	adresse
data	données
Received Master Frame	Trame-Maître reçue
Sent Slave_Frame	Trame-Esclave envoyée
Next Master Frame	Trame-Maître suivante
time	durée

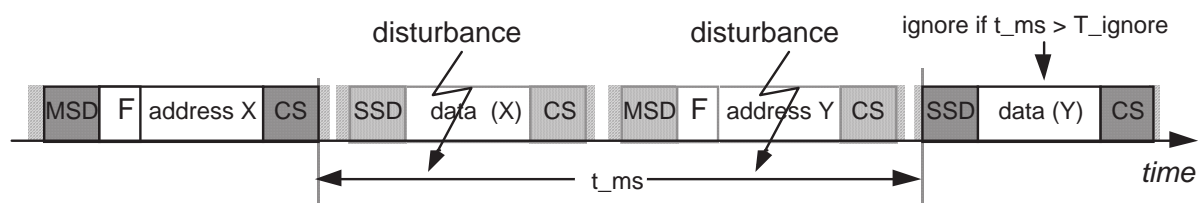
**Figure 41 – Espacement des trames du côté source**

NOTE La valeur recommandée de  $T_{source}$  est 4,0  $\mu s$ . Les dispositifs avec  $T_{source} > 3,0 \mu s$  sont toujours conformes à 4.3.2.

## 6.2.4 Espacement des trames côté destinataire

### 6.2.4.1 Espacement des trames par défaut

Un dispositif de destination doit ignorer toutes les Trames-Esclaves qui commencent plus tard que  $T_{\text{ignore}}$  après la fin de la dernière Trame-Maître reçue, comme le montre la Figure 42.



#### Légende

Anglais	Français
disturbance	perturbation
ignore if	ignore si
address	adresse
data	données
time	durée

**Figure 42 – Espacement des trames côté destinataire(s)**

La temporisation  $T_{\text{ignore}}$  doit avoir une valeur par défaut de  $T_{\text{reply\_def}} = 42,7 \mu\text{s}$ .

**NOTE** Les Trames-esclaves de Données de Processus ne contiennent pas d'adresse, mais sont identifiées par la Trame-Maître précédente. La perte d'une Trame-Esclave (data(x)) et de la Trame-Maître suivante (adresse Y) peut conduire un dispositif à accepter la seconde Trame-Esclave (data(Y)) comme appartenant à la première Trame-Maître (adresse X). Par conséquent, un dispositif soupçonne qu'il y a une double perte s'il mesure un délai  $t_{\text{ms}}$  plus long que le temps nécessaire pour transmettre la plus petite Trame-Esclave et la Trame-maître suivante, qui est de  $49 \mu\text{s}$  ( $22 \mu\text{s} + 2 \mu\text{s} + 22 \mu\text{s} + 3 \mu\text{s}$ ). Une valeur de  $42,7 \mu\text{s}$ , inférieure à  $49 \mu\text{s}$ , est donc attribuée par défaut à  $T_{\text{ignore}}$ . D'autre part, il convient qu'un dispositif soit capable d'accepter une Trame-Esclave retardée de  $T_{\text{reply}}$ , afin de permettre la réception de Trames-esclaves retardées dans le cas le plus défavorable. Lorsque  $T_{\text{reply}}$  est inférieur à  $42,7 \mu\text{s}$ , cette condition est toujours remplie, mais elle ne l'est pas pour le mode Extended Reply Delay qui demande des mesures de prévention particulières (6.2.4.2).

### 6.2.4.2 Espacement des trames en mode Extended Reply Delay

Dans des bus longs où la valeur de  $T_{\text{reply}}$  dépasse la valeur par défaut de  $T_{\text{reply\_def}}$ , une valeur supérieure à  $T_{\text{reply\_def}}$  peut être attribuée à  $T_{\text{ignore}}$ , sous réserve des mesures de prévention suivantes:

- les dispositifs dont la temporisation  $T_{\text{ignore}}$  est réglée à une valeur supérieure à  $T_{\text{reply\_def}}$  doivent indiquer ceci en mettant à "1" le bit ERD (Extended Reply Delay) dans leur Device\_Status;
- la variable de configuration  $T_{\text{safe}}$  doit être égale à la valeur la plus élevée de  $T_{\text{ignore}}$  utilisée par tout dispositif raccordé au bus;
- la variable de configuration  $T_{\text{safe}}$  doit être égale à  $T_{\text{reply\_def}}$  si tous les dispositifs d'un bus donné utilisent la valeur  $T_{\text{ignore}} = T_{\text{reply\_def}}$ .

**NOTE** Dans des applications Extended Reply Delay, le maître a la responsabilité d'espacer les Trames-Maîtres de manière à ce que le temps minimum nécessaire pour transmettre une Trame-Esclave et la Trame-Maître suivante soit supérieur à  $T_{\text{safe}}$ .

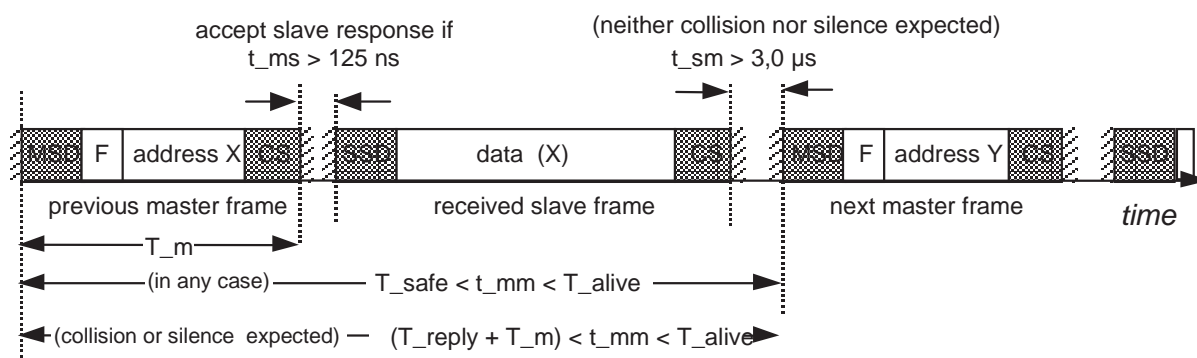
En effet, si un dispositif utilise une valeur de  $T_{\text{ignore}}$  supérieure à  $T_{\text{safe}}$ , ce dispositif crée une atteinte potentielle à l'intégrité. Pour détecter cette situation, il convient que l'administrateur de bus vérifie si un dispositif fonctionne en mode Extended Reply Delay en lisant son Device\_Status.

### 6.2.5 Espacement des trames au niveau du maître

Un maître doit envoyer la Trame-Maître suivante au plus tard  $T_{\text{alive}} = 1,3 \text{ ms}$  après le début de la Trame-Maître précédente, comme le montre la Figure 43.

Un maître peut envoyer la trame-maître suivante:

- au plus tôt  $T_{\text{safe}}$  après le début de sa trame-maître précédente dans tous les cas (6.2.4.2);
- au plus tôt un temps  $T_{\text{reply}} + T_{\text{m}}$  après le début de sa trame-maître précédente si une collision ou un silence est prévisible en réponse à sa trame-maître envoyée;
- au plus tôt  $T_{\text{sm\_min}} = 3,0 \mu\text{s}$  s'il ne prévoit aucune collision, ni aucun silence en réponse à sa trame-maître précédente.



#### Légende

Anglais	Français
accept slave response if	acceptation de la réponse de l'esclave si
neither collision nor silence expected	aucune collision, ni aucun silence prévisibles
address	adresse
data	données
previous master frame	trame-maître précédente
received slave frame	trame-esclave reçue
next master frame	trame-maître suivante
time	durée
in any case	dans tous les cas
collision or silence expected	collision ou silence prévisibles

**Figure 43 – Espacement des trames du côté du maître**

NOTE 1 Le retard  $T_{\text{alive}}$  permet de superviser l'activité du maître en cas de maîtres redondants.

NOTE 2 L'exigence  $t_{\text{mm}} > T_{\text{safe}}$  est toujours satisfaite lorsque tous les dispositifs du bus utilisent la valeur par défaut  $T_{\text{ignore}} = 42,7 \mu\text{s}$ , puisque le temps minimum nécessaire à la transmission d'un télégramme est égal à  $49 \mu\text{s}$  ( $= 22,0 \mu\text{s} + 2,0 \mu\text{s} + 22,0 \mu\text{s} + 3,0 \mu\text{s}$ ), ce qui est supérieur à  $42,7 \mu\text{s}$ . Si un dispositif quelconque fonctionne en mode Extended Reply Delay, la valeur la plus élevée de  $T_{\text{ignore}}$  sera attribuée à  $T_{\text{safe}}$ .

## 6.3 Détection de trames correctes, collision et silence par le maître

### 6.3.1 Trame correcte (définition)

Une Trame-Esclave correcte est définie comme une trame valide (5.1.7) qui est reçue dans les limites fixées par 4.2.4, commençant par un Délimiteur de Début de Trame-Esclave et dont la taille correspond à celle annoncée dans la Trame-Maître. Les CRC corrects y sont par ailleurs insérés.

NOTE Seules les trames correctes peuvent être transmises à l'application.

### 6.3.2 Détection de collision par le maître

Le maître doit détecter une collision si le décodeur de la trusted line (ligne de confiance) fait état d'une collision ou si la Trame-Esclave reçue n'est pas correcte.

Le maître ne doit pas détecter de collision s'il reçoit une Trame-Esclave correcte, même si une seconde trame valide arrive dans l'intervalle  $T_{reply}$  après la première trame en raison de délais de propagation élevés. Le maître doit alors considérer la première Trame-Esclave correcte reçue comme une réponse unique.

Une collision est dite prévisible pour les trames de type General\_Event\_Response, Group\_Event\_Response et Individual\_Event\_Response.

NOTE 1 Une collision a lieu lorsque plusieurs émetteurs sont actifs simultanément. Normalement, ceci se produit uniquement au cours de l'arbitrage des événements. Seul le maître surveille la collision.

NOTE 2 Dans un bus long, il est possible que le maître reçoive une trame correcte bien que plusieurs esclaves répondent simultanément: un émetteur proche peut écraser un émetteur lointain ou une seconde trame peut arriver après une première trame car le délai de réponse  $T_{reply}$  peut dépasser la durée minimale d'une Trame-Esclave ( $= 22,0 \mu s$ ).

### 6.3.3 Détection de silence par le maître

Le maître doit détecter un silence si le décodeur ne fait état d'aucune trame valide ni d'aucune collision durant  $T_{reply}$  après la fin de la Trame-Maître précédente.

Un silence est dit prévisible sur toutes les trames sauf les Process\_Data\_Responses.

NOTE Une perturbation peut conduire le maître à prendre un silence pour une collision. Ceci peut engager la recherche d'événements sur la mauvaise voie. Le décodeur permet de faire la distinction entre une perturbation et une collision.

## 7 Contrôle de la Couche de Liaison

### 7.1 Adressage

#### 7.1.1 Adresse de Dispositif

Un dispositif doit être identifié sur un bus par son Adresse de Dispositif.

L'Adresse de Dispositif doit avoir une taille de 12 bits.

L'Adresse de Dispositif 0 ne doit pas être utilisée.

NOTE 1 Il convient que les dispositifs prenant en charge la capacité Données de Messagerie aient une Adresse de Dispositif inférieure à 256.

NOTE 2 L'adresse du dispositif de diffusion est indiquée par le champ de mode, et non par une adresse particulière.

#### 7.1.2 Logical\_Address

Une source de Données de Processus doit être identifiée par sa Logical\_Address.

Logical\_Address doit avoir une taille de 12 bits.

Logical\_Address 0 ne doit pas être utilisée.

### 7.1.3 Group\_Address

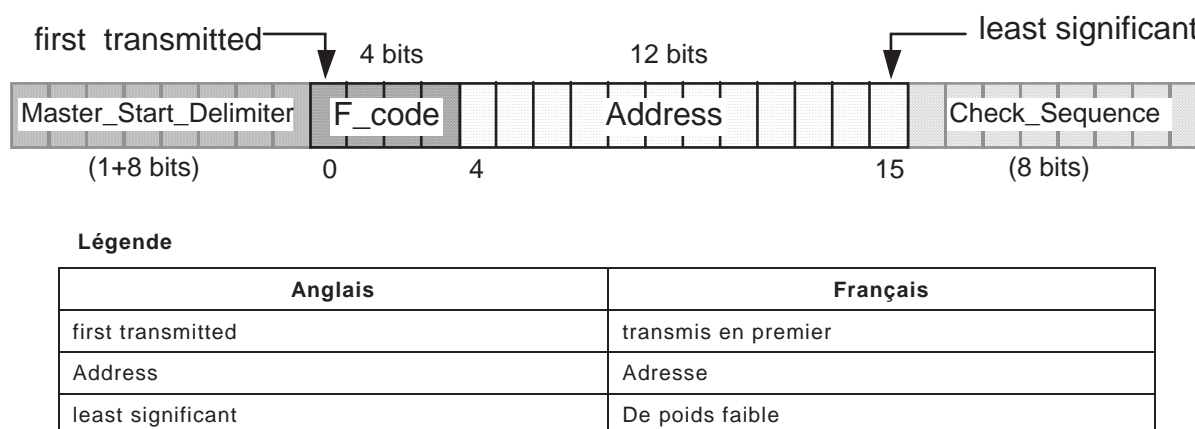
Pour les besoins de Event\_Arbitration, l'adressage des dispositifs peut s'effectuer par leur Group\_Address, comme spécifié en 8.3.1.

## 7.2 Contenu des Trames-maîtres

### 7.2.1 Format de Trame-Maître

La Trame-Maître doit porter un mot de données de 16 bits.

Le premier bit transmis de ce mot doit être le bit de poids fort, identifié comme bit 0, comme le montre la Figure 44.



**Figure 44 – Contenu d'une Trame-Maître**

Les quatre bits de poids fort du mot de 16 bits doivent représenter le F\_code.

Les 12 bits de poids faible doivent représenter une adresse ou des paramètres spécifiés par le F\_code.

### 7.2.2 Codage du F\_code

Le F\_code de 4 bits qualifie les 12 bits suivants et indique la taille de la Trame-Esclave prévisible.

Le F\_code doit être codé selon le Tableau 8.

**Tableau 8 – Types de Trames Maître et F\_code**

Trame-Maître			Trame-Esclave			
F_code	Adresse	Demande	Source	Taille (bits)	Réponse	Destinataire
0	logique	Données de Processus	dispositif unique abonné comme source	16	Données de Processus (dépendant de l'application)	tous les dispositifs abonnés comme destinataire
1				32		
2				64		
3				128		
4				256		
5		Réservée		-		
6		Réservée		-		
7		Réservée		-		
8	dispositif	Mastership_Transfer	maître proposé	16	Mastership_Transfer	maître
9	tous les dispositifs	General_Event (paramètres)	un ou plusieurs dispositifs	16	Event_Identifier	maître
10	dispositif	Réservée	-	-		
11	dispositif	Réservée	-	-		
12	dispositif	Données de Messagerie	dispositif unique	256	Données de Messagerie	dispositif(s) sélectionné(s)
13	groupe de dispositifs	Group_Event	un ou plusieurs dispositifs	16	Event_Identifier	maître
14	dispositif	Single_Event	dispositif unique	16	Event_Identifier	maître
15	dispositif	Device_status	dispositif unique	16	Device_status	maître ou dispositif de supervision

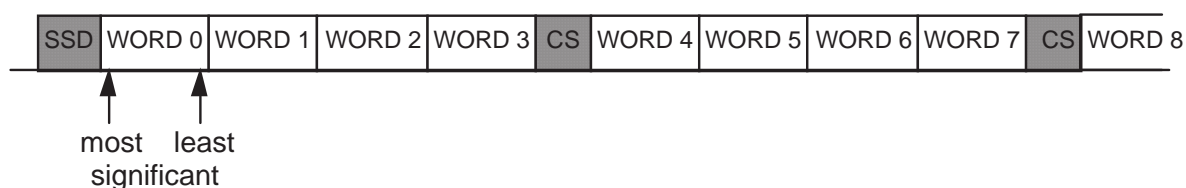
### 7.3 Contenu des Trames-esclaves

#### 7.3.1 Format de Trame-Esclave

Une Trame-Esclave doit transmettre 1, 2, 4, 8 ou 16 mots de 16 bits chacun.

Les bits d'un mot doivent être transmis de manière telle que le bit de poids fort soit en premier (le mot est traité comme un nombre entier de 16 bits non signé, le premier bit transmis ayant un décalage 0).

Les mots doivent être numérotés en commençant par 0, comme le montre la Figure 45.



#### Légende

Anglais	Français
WORD	MOT
most significant	De poids fort
least significant	De poids faible

**Figure 45 – Ordre des mots dans une Trame-Esclave**

Les trames doivent être complétées par des "0" si la taille entière de la trame n'est pas utilisée pour les données de liaison.

NOTE Le contenu d'une Trame-Esclave n'est pas spécifié sauf pour l'en-tête des Données de Messagerie et pour les trames de Données de Supervision.

### 7.3.2 Erreur de taille

Un destinataire doit ignorer une Trame-Esclave si sa taille ne correspond pas à celle indiquée par le F\_code de la Trame-Maître, même si son Code Détecteur d'Erreur est correct par ailleurs.

## 7.4 Types de télégrammes

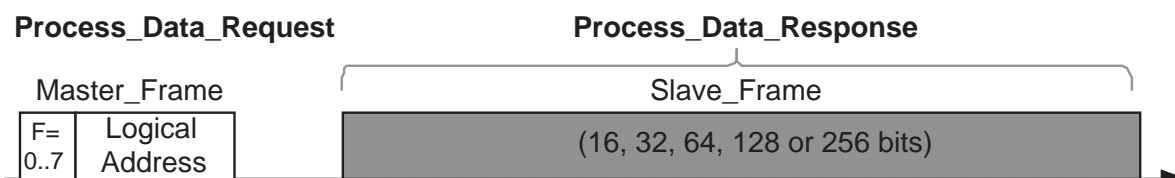
### 7.4.1 Télégramme de Données de Processus

#### 7.4.1.1 Format des Données de Processus

Un maître doit demander la transmission de Données de Processus en envoyant une Process\_Data\_Request qui est une Trame-Maître avec l'un des F\_codes: 0..4.

NOTE Les F\_codes 5..7 sont réservés pour les extensions futures des Données de Processus.

L'esclave qui est la source des Données de Processus correspondant à l'indicatif dans la Process\_Data\_Request doit répondre par une Process\_Data\_Response, comme le montre la Figure 46.



#### Légende

Anglais	Français
Logical Address	Adresse Logique
or	ou

**Figure 46 – Télégramme de Données de Processus**

#### 7.4.1.2 Indicateur de Données de Processus

Les Données de Processus doivent être identifiées à la fois par la Logical\_Address de 12 bits et le F\_code dans la Trame-Maître, qui indique la taille des Données de Processus (7.2.2).

Pour chaque Logical\_Address, la taille des Données de Processus correspondantes doit être configurée dans l'émetteur et dans tous les dispositifs destinataires avant le début de tout fonctionnement normal.

#### 7.4.1.3 Tampons

Un dispositif doit maintenir un tampon pour chaque Logical\_Address à laquelle il est abonné, en tant qu'émetteur ou destinataire.

#### 7.4.1.4 Dispositifs émetteurs

Un dispositif qui est l'émetteur des Données de Processus dont l'indicateur apparaît dans la Process\_Data\_Request, doit lire le contenu de son tampon source et l'envoyer en une seule opération indivisible dans la Trame-Esclave.

Un dispositif ne doit pas répondre si le F\_code de la Process\_Data\_Request indique une longueur de trame différente de la longueur configurée pour cette Logical\_Address.

Un tampon source doit pouvoir être écrasé par des "0" pour invalider ses données.

#### 7.4.1.5 Dispositifs destinataires

Un dispositif collectant des Données de Processus, dont l'indicateur apparaît dans la Process\_Data\_Request, doit stocker cette trame dans le tampon correspondant, écrasant le contenu précédent de ce tampon par le contenu de la trame reçue sur le bus, en une seule opération indivisible.

Un dispositif collectant une trame de Données de Processus, dont l'indicateur apparaît dans la Process\_Data\_Request, ne doit pas accepter de Trame-Esclave si le F\_code indique une longueur de trame différente de la longueur configurée pour cette Logical\_Address.

Un tampon destinataire doit avoir le moyen d'indiquer la date de sa dernière mise à jour.

### 7.4.2 Données de Messagerie

#### 7.4.2.1 Format des Données de Messagerie

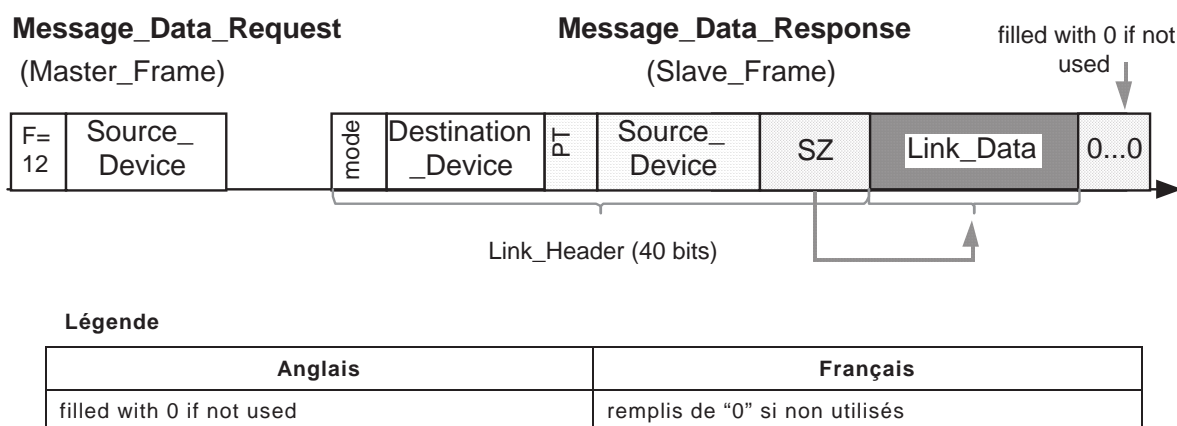
Le maître doit demander la transmission de Données de Messagerie en envoyant une Message\_Data\_Request, qui est une Trame-Maître avec F\_code = 12.

Le dispositif adressé par la Trame-Maître doit répondre par une Message\_Data\_Response, avec le format suivant, en commençant par le premier bit transmis, comme le montre la Figure 47:

- a) les 4 premiers bits définissent le mode de communication:
  - le Mode '0001'B (+1) spécifie le mode d'adressage " envoi unique ",
  - le Mode '1111'B (-1) spécifie le mode d'adressage " diffusion ",
  - tous les autres modes sont réservés pour une utilisation future;
- b) les 12 bits suivants définissent l'adresse du Destination\_Device.  
En cas d'utilisation d'un mode de diffusion, ce champ peut être utilisé par l'application;
- c) les 4 bits suivants définissent le Protocol\_Type. Le Protocol\_Type pour les Protocoles en Temps Réel est défini en 10.3.4;



- d) les 12 bits suivants indiquent l'adresse du Source\_Device;
- e) les 8 bits suivants (SZ) expriment le nombre d'octets significatifs dans les Données de Liaison suivantes;
- f) les bits suivants portent les Données de Liaison;
- g) les octets de la trame restants sont remplis de "0" si la taille des Données de Liaison est inférieure à 216 bits.



**Figure 47 – Télégramme de Données de Messagerie**

NOTE 1 Le contenu des Données de Liaison est défini à l'Article 6 de la CEI 61375-2-1 "Protocoles en Temps Réel".

NOTE 2 Le Mode "envoi unique" et le Mode "diffusion" sont deux codes complémentaires. Par exemple, si un Mode futur exprime l'envoi unique comme 3 ('0011'B), le Mode 'diffusion' correspondant sera -3 ('1110'B).

#### 7.4.2.2 Queues

Chaque dispositif capable de communiquer en Données de Messagerie doit comporter deux queues, l'une pour l'envoi (Send\_Queue) et l'autre pour la réception (Receive\_Queue) de Données de Messagerie.

NOTE Les queues ne sont pas hiérarchisées par ordre de priorité.

#### 7.4.2.3 Protocole des Données de Messagerie

Lorsqu'il reçoit une Message\_Data\_Request, un dispositif spécifié par l'adresse source dans la Message\_Data\_Request doit:

- envoyer une Message\_Data\_Response à partir de sa Send\_Queue et la supprimer de cette dernière; ou
- ne pas répondre du tout si sa Send\_Queue est vide.

Lorsqu'il reçoit une Message\_Data\_Response, un dispositif qui reconnaît sa propre Adresse de Dispositif comme Destination\_Address ou le mode "diffusion" doit:

- ajouter les Données de Messagerie à sa Receive\_Queue, si cela est possible, ou
- ignorer les Données de Messagerie, si cela n'est pas possible.

NOTE La répétition de trames perdues relève de la responsabilité de la Transport\_Layer.

#### 7.4.3 Télégrammes de Données de supervision

##### 7.4.3.1 Format des Données de Supervision

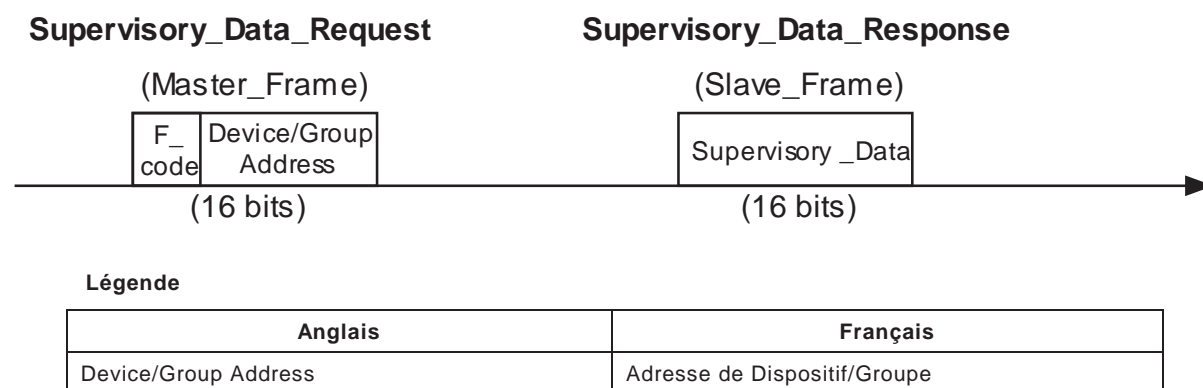
Un maître doit demander des Données de Supervision à un dispositif ou doit envoyer des Données de Supervision à un dispositif ayant une Supervisory\_Data\_Request, qui est une

Trame-Maître utilisant l'un des F\_codes suivants (les 12 bits ajoutés sont spécifiés dans les paragraphes correspondants):

F_code =	8	Mastership_Transfer_Request	(9.3.2);
F_code =	9	General_Event_Request	(8.3.4.1);
F_code =	13	Group_Event_Request	(8.3.4.2);
F_code =	14	Single_Event_Request	(8.3.4.3);
F_code =	15	Device_Status_Request	(8.4.1.1).

La Supervisory\_Data\_Response doit porter 16 bits de Données de Trames, comme le montre la Figure 48. Ces bits sont spécifiés au paragraphe définissant les trames de réponse correspondant au F\_code:

F_code =	8	Mastership_Transfer_Response	(9.3.2);
F_code =	9	Event_Identifier_Response	(8.3.4.4);
F_code =	13	Event_Identifier_Response	(8.3.4.4);
F_code =	14	Event_Identifier_Response	(8.3.4.4);
F_code =	15	Device_Status_Response	(8.4.1.2).



**Figure 48 – Télégramme de Données de Supervision**

### 7.4.3.2 Protocole de Données de Supervision

Les télégrammes de Données de Supervision ne doivent pas être répétés immédiatement si aucune Supervisory\_Data\_Response n'arrive après une Supervisory\_Data\_Request.

## 8 Attribution du support

### 8.1 Organisation

#### 8.1.1 Tour

L'accès au support doit être contrôlé par l'un des administrateurs de bus, à savoir le maître, qui assure le contrôle du bus pendant la durée d'un tour.

A la fin d'un tour, le maître doit tenter de passer la maîtrise à un autre administrateur de bus ou doit l'obtenir à nouveau s'il ne trouve pas d'administrateur disponible.

NOTE Le début d'un tour peut être observé par les dispositifs raccordés, qui peuvent contrôler quel administrateur de bus est actuellement maître, à des fins de synchronisation.

#### 8.1.2 Période de Base

Le maître doit diviser son tour en créneaux de temps fixes, appelés "Périodes de Base", qui ont tous la même durée  $T_{bp}$ .

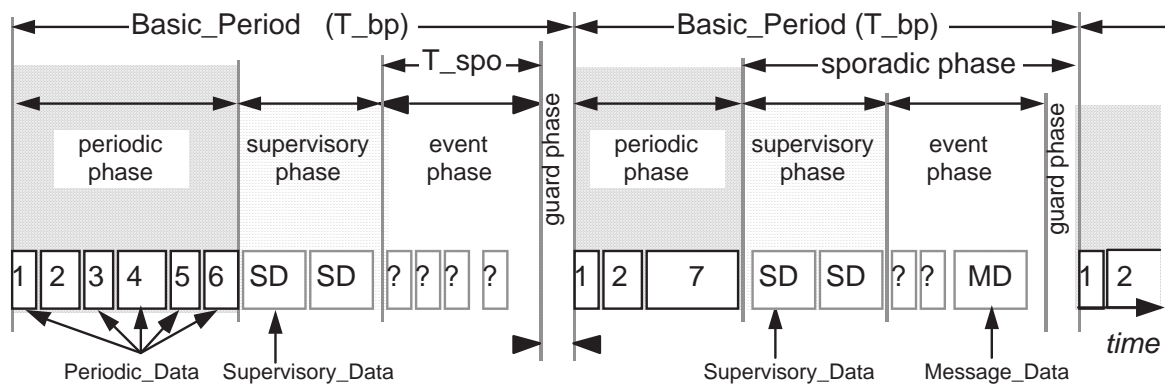
La Période de Base devant être utilisée pour un tour doit prendre une valeur comprise entre:

$1,0\text{ ms} \leq T_{bp} \leq 2,5\text{ ms}.$

NOTE 1 A des fins de synchronisation, toute fraction de la période du processus contrôlé, obtenue en divisant cette période par deux et incluse dans la plage comprise entre 1 ms et 2 ms peut être utilisée, par exemple: 1,042 ms (16 × 60 Hz), 1,250 ms (16 × 50 Hz) et 1,875 ms (32 × 16,66 Hz).

Une Période de Base doit être divisée en quatre phases, comme le montre la Figure 49:

- a) une Phase Périodique;
- b) une Phase de Supervision;
- c) une Phase d'Événements;
- d) une Phase de Garde.



Légende

Anglais	Français
sporadic phase	phase a périodique
periodic phase	phase p ériodique
supervisory phase	phase de supervision
event phase	phase d'événements
guard phase	phase de garde
time	durée

Figure 49 – Périodes de Base

NOTE 2 La Phase d'Événements, la Phase de Supervision et la Phase de Garde forment la Phase Apériodique.

La valeur de T\_spo peut varier d'une Période de Base à l'autre.

La valeur par défaut de T\_spo doit être de 350,0 µs.

S'il devient nécessaire de réduire T\_spo pour un nombre limité de Périodes de Base, T\_spo doit dépasser le temps de transmission du télégramme le plus court, T<sub>mm\_min</sub> = 49 µs + T<sub>reply</sub>.

NOTE 3 La valeur maximale de T\_spo est égale à la durée de la Période de Base.

8.1.2.1 Phase Périodique

Au cours de la Phase Périodique, le maître doit procéder au Scrutin Périodique spécifié en 8.2.

### 8.1.2.2 Phase d'Événements

Au cours de la Phase d'Événements, le maître doit procéder au Scrutin d'Événements spécifié en 8.3.

### 8.1.2.3 Phase de Supervision

Le maître doit utiliser la Phase de Supervision pour le Devices\_Scan (8.4) et pour le Transfert de Maîtrise (Article 9).

NOTE Les Données de Supervision peuvent être transmises comme Periodic\_Data, Sporadic\_Data ou à l'initiative du maître.

### 8.1.2.4 Phase de Garde

Le maître doit réserver la Phase de Garde comme tampon après les transmissions apériodiques pour permettre un début exact de la Phase Périodique suivante.

Le maître ne doit pas envoyer de Trame-Maître si le temps de transmission de la réponse prévisible la plus longue est plus long que le temps restant jusqu'à ce que la Période de Base suivante soit programmée.

## 8.1.3 Remplissage

Chaque Période de Base doit être construite de manière telle que le maître ait la possibilité d'émettre une Trame-Maître (quel que soit son type) au moins une fois toutes les  $T_{alive}$  (4.2.5) = 1,3 ms pour permettre à chaque dispositif de contrôler la ligne (9.2.2).

## 8.2 Scrutin Périodique

### 8.2.1 Liste Périodique

La Liste Périodique doit définir les Trames-maîtres envoyées périodiquement ainsi que le temps alloué à la Phase Apériodique pour chaque Période de Base d'un tour.

Au cours du Scrutin Périodique, le maître doit envoyer une séquence prédéfinie de Trames-Maîtres conformément à sa Liste Périodique.

Les modifications apportées à la Liste Périodique ne doivent pas prendre effet au cours du même tour.

NOTE La Liste Périodique contient les Periodic\_Data, c'est-à-dire principalement des Process Data\_Requests, mais également des Supervisory\_Data\_Requests ou des Message\_Data\_Requests pour des objectifs particuliers.

### 8.2.2 Période Individuelle

Le maître doit scruter chaque Periodic\_Data selon sa Période Individuelle  $T_{ip}$ .

Une Période Individuelle doit être égale à la Période de Base multipliée par une puissance de 2, mais ne doit pas dépasser 1024  $T_{bp}$ , ou en général:

$$T_{ip}(i) = T_{bp} \times 2^n, \text{ avec } n = (0..10).$$

La plus longue Période Individuelle de la Liste Périodique, appelée la Macro\_Period, ne doit pas dépasser 1024,0 ms.

NOTE 1 La Période Individuelle est définie par l'application.

NOTE 2 Si la Période Individuelle la plus courte est plus grande que la Période de Base, la Phase Périodique de certaines Périodes de Base peut être vide.

### 8.2.3 Construction de la Phase Périodique

La Liste Périodique est configurée par l'application. Les paragraphes suivants indiquent comment la Liste Périodique doit être construite.

#### 8.2.3.1 Conventions relatives à la construction de la Liste Périodique

Les conventions suivantes doivent s'appliquer au format de la Liste Périodique tel que spécifié dans l'Interface de Supervision de Liaison:

- a) les Periodic\_Data sont classées selon leur Période Individuelle  $T_{ip}(i)$ ;
- b) un Cycle regroupe les Periodic\_Data avec la même Période Individuelle et reçoit le même nom que sa Période Individuelle en multiple de la Période de Base (par exemple, Cycle\_2 lorsque le scrutin a lieu toutes les 2 ms et lorsque la Période de Base est de 1 ms);
- c) un Macro\_Cycle est constitué de tous les cycles qui forment une Macro\_Period; il est donc formé de:
  - 1024 Périodes de Base si  $T_{bp} = 1$  ms ou
  - 512 Périodes de Base si  $T_{bp} = 2$  ms;
- d) les Périodes de Base sont numérotées  $BP(j)$ , où
  - $j = (0 \dots macro - 1)$ ,
  - $BP(0)$  = première période d'un Macro\_Cycle,
  - $BP(macro - 1)$  = dernière période d'un Macro\_Cycle;
- e) un tour est défini comme le nombre de Macro\_Cycles après lequel un transfert de maîtrise doit avoir lieu. Un tour peut être arrêté après chaque Macro\_Cycle par une commande de l'Interface de Supervision de Liaison;
- f) un Cycle peut être séparé en Sous-cycles répartis sur plusieurs Périodes de Base. Le Cycle\_n a n Sous-cycles, par exemple, le Cycle\_16 a 16 Sous-cycles parmi lesquels certains ou tous peuvent être vides. Un Sous-cycle est identifié par son indice, par exemple, le Cycle\_4.1 est le second Sous-cycle du Cycle\_4;
- g) la dernière Phase Apériodique de chaque Macro\_Cycle est réservée pour le transfert de maîtrise, même si elle n'est pas utilisée pour ledit transfert dans ce Macro\_Cycle.

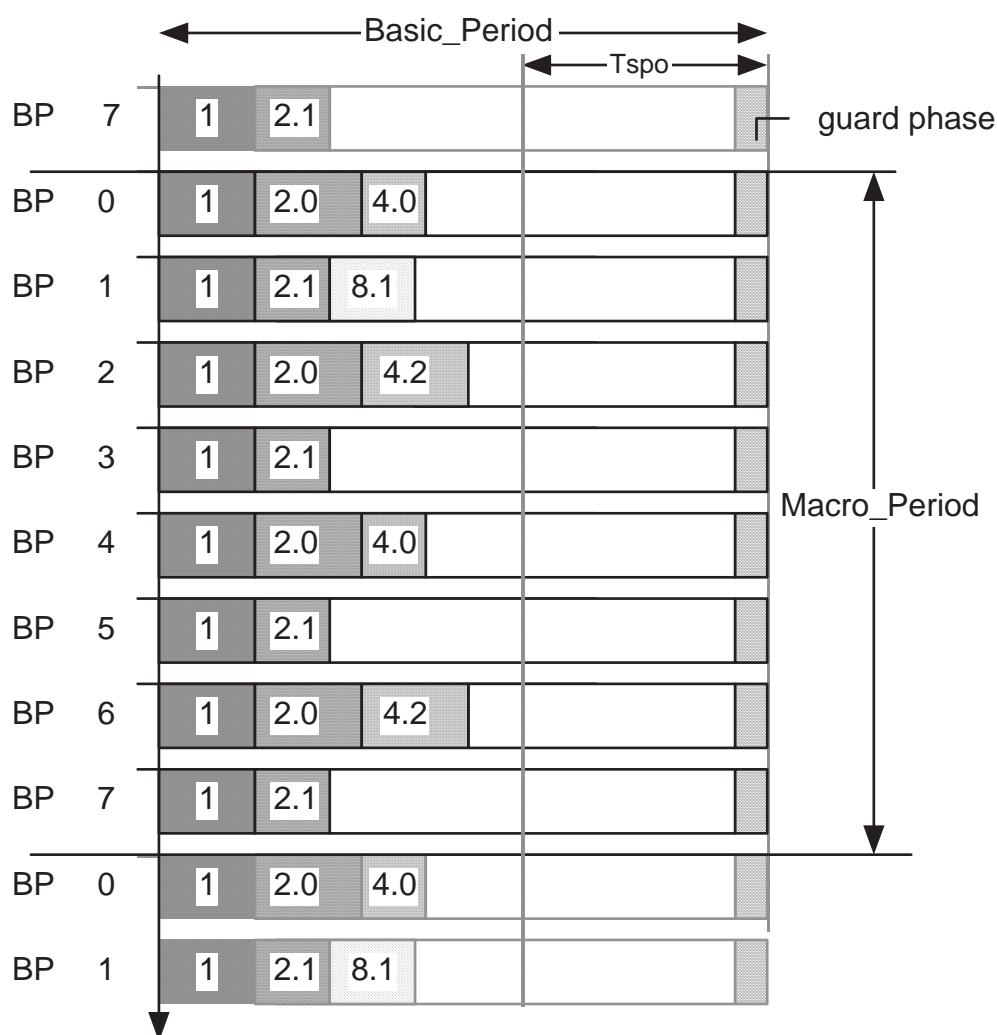
NOTE Il n'existe pas de contraintes relatives à la répartition des Sous-cycles sur les différentes périodes, mais il est recommandé, pour des raisons de performances, de répartir les cycles de manière uniforme sur les périodes.

#### 8.2.3.2 Construction d'un Macro\_Cycle

La Liste Périodique doit contenir deux sous-listes pour chaque Macro\_Cycle:

- a) la Cycle\_List, contenant les Trames-maîtres appartenant à chacun des 11 cycles (Cycle\_1 à Cycle\_1024), et
- b) la Split\_List, contenant le nombre de trames de chaque cycle à envoyer dans chaque Sous-cycle.

EXEMPLE La Figure 50 montre une construction simplifiée d'un Macro\_Cycle formé de 8 Cycles.



**Légende**

Anglais	Français
guard phase	phase de garde

**Figure 50 – Exemple de construction du Macro\_Cycle**

NOTE La Liste Périodique de l'administrateur de bus de cet exemple contient:

- les quatre Cycle\_Lists, une pour chaque cycle 1, 2, 4 et 8;
- les trois Split\_Lists indiquant le nombre de trames pour
  - les Sous-cycles 2.0 et 2.1 (le cycle 1 n'a pas de scission, il est présent dans chaque période),
  - les Sous-cycles 4.0, 4.1 (vide ici), 4.2 et 4.3 (vide ici),
  - les Sous-cycles 8.0, 8.1, 8.2, 8.3, 8.4, 8.4, 8.5, 8.6 et 8.7 (tous vides ici sauf 8.1).

### 8.3 Scrutin d'Événements

#### 8.3.1 Group\_Address

NOTE Pour les besoins de Event\_Arbitration, le maître sélectionne les groupes de dispositifs qui partagent les mêmes bits d'adresse de poids faible.

Les dispositifs du bus qui diffèrent par les bits de poids fort de leur Adresse de Dispositif forment un groupe, identifié par sa Group\_Address.

La Group\_Address est caractérisée par

- $G = \{M, C\}$ ;
- M étant le nombre de chiffres de poids fort par lesquels les adresses de dispositifs diffèrent;
- 2M étant le nombre de dispositifs possibles dans le groupe ( $M = 1..12$ );
- C étant le nombre binaire exprimé par les bits communs de poids faible de l'adresse.

EXEMPLE Les dispositifs avec adresse binaire:

1111 0000 0101,  
1000 0000 0101 et  
0101 0000 0101

appartiennent au même groupe  $\{M = 4, C = 5\}$ .

### 8.3.2 Event\_Round

Un Event\_Round rassemble toutes les demandes d'événement d'une priorité donnée en suspens au moment où le Event\_Round commence.

#### 8.3.2.1 Début

Le maître doit commencer un Event\_Round en envoyant une General\_Event\_Request, qui définit:

- a) qu'il s'agit d'un nouveau Event\_Round et non d'une continuation {new\_round};
- b) qu'une réponse est requise {answer\_now};
- c) un des deux niveaux de priorité des événements pour cet Event\_Round {high\_priority ou low\_priority}.

Cette General\_Event\_Request s'adresse au groupe  $\{M = 12\}$ , c'est-à-dire à tous les 4096 dispositifs possibles.

#### 8.3.2.2 Effacement et autorisation des événements

Un dispositif doit participer à un Event\_Round s'il comprend un événement de priorité correspondante en suspens au moment où il reçoit une General\_Event\_Request {new\_round}.

Un dispositif participant doit répondre par sa Event\_Identifier\_Response à une Event\_Request avec les F\_codes suivants:

- a) F\_code 9: General\_Event\_Request (si "answer\_now" et si la priorité correspond);
- b) F\_code 13: Group\_Event\_Request (si la Group\_Address correspond à celle de ce dispositif);
- c) F\_code 14: Single\_Event\_Request (si la Device\_Address correspond à celle de ce dispositif).

Un dispositif doit effacer l'événement correspondant lorsqu'il reçoit une Event\_Read\_Request qui correspond au F\_code et à l'adresse de la Event\_Identifier\_Response.

NOTE 1 Un esclave n'est pas informé du fait que le maître a lu sa Event\_Identifier\_Response.

NOTE 2 Plusieurs dispositifs peuvent signaler le même événement. Dans ce cas, tous les dispositifs comportant cet événement l'effacent une fois que l'événement a été lu, puisqu'aucun ne sait quel dispositif a signalé cet événement.

### 8.3.2.3 Situations reconnues

Une Event\_Request est définie comme l'une des trois demandes suivantes: General\_Event\_Request, Group\_Event\_Request et Single\_Event\_Request.

Le maître doit distinguer les situations suivantes se produisant en réponse à une Event\_Request, dans les limites de T\_Reply:

- a) correcte: une Trame-Esclave correcte de 16 bits a été reçue (6.3.1);
- b) silence: aucune trame n'a été reçue (6.3.3);
- c) collision: tous les autres cas (5.1.9).

### 8.3.2.4 Algorithme de recherche d'événements

Dans le cas où une collision se produit en réponse à une General\_Event\_Request, le maître doit rechercher tous les événements en suspens en envoyant une Group\_Event\_Request ou une Single\_Event\_Request, dans le but de restreindre le nombre de dispositifs participant à l'interrogation.

NOTE L'algorithme de recherche des événements en suspens n'est pas spécifié. Une méthode générale est indiquée et recommandée en 8.3.3, mais d'autres algorithmes sont autorisés. Les algorithmes peuvent en particulier tirer parti du fait que l'identité des dispositifs participants est connue et que les événements ont tendance à se produire en rafales pour les transferts de Données de Messagerie.

### 8.3.2.5 Fin de Event\_Arbitration

Le maître doit mettre fin à Event\_Arbitration:

- a) lorsqu'il reçoit une Event\_Identifier\_Response;
- b) lorsqu'il détecte une collision ou un silence en réponse à une Single\_Event\_Request.

Dans des configurations où les dispositifs signalant les événements ont une plage d'adresse limitée (par exemple, 1..255), le maître peut considérer que la recherche se termine lorsqu'aucune réponse n'est donnée à une Group\_Event\_Request qui s'adresse à ce groupe de dispositifs.

### 8.3.2.6 Interruption et poursuite de Event\_Round

Le Event\_Round de priorité élevée ne doit être suspendu que par la Phase Périodique.

Dans le cas où deux types de priorité sont configurés (élevée et basse), un Event\_Round de priorité élevée doit être démarré après chaque lecture d'un événement de basse priorité, suspendant le Event\_Round de basse priorité.

Le Event\_Round de basse priorité doit être repris au moyen d'une General\_Event\_Request {NOT new\_round, answer\_now}.

### 8.3.2.7 Fin de Event\_Round

Lorsque l'algorithme de recherche a été entièrement réalisé et après que tous les événements ont été lus, le maître doit envoyer une General\_Event\_Request {NOT new\_round, answer\_now} pour rechercher les dispositifs omis par le Event\_Round.

Si cette General\_Event\_Request n'obtient pas de réponse, le maître doit mettre fin à Event\_Round.

Il doit sinon reprendre Event\_Round depuis le début.



Si le même événement est signalé plus de deux fois au cours la procédure de fin de Event\_Round, le maître doit mettre fin à cette dernière et signaler une erreur.

NOTE Des dispositifs peuvent être omis dans un tour en raison de la mauvaise interprétation d'une situation, et en particulier:

- une collision non détectée (trame correcte reçue malgré la transmission de plusieurs dispositifs);
- un silence non détecté (une perturbation a provoqué l'apparition d'une collision alors qu'il n'y en a pas);
- une erreur au cours de la transmission de la Event\_Identifier\_Response;
- deux dispositifs ou plus demandant le même événement.

### 8.3.2.8 Event\_Read

Le maître doit envoyer à nouveau la Event\_Identifier\_Response reçue (Trame-Esclave) en tant que Event\_Read\_Request (comme Trame-Maître).

Le maître ne doit pas répéter Event\_Read\_Request même s'il ne reçoit pas une Event\_Read\_Response correcte en réponse, par exemple, dans le cas d'une erreur de transmission.

NOTE Le maître peut envoyer une Event\_Read\_Request juste après avoir reçu une Event\_Identifier\_Response, ou il peut mettre en paquet les Event\_Read\_Requests de plusieurs Event\_Identifier\_Responces reçues, une fois le Event\_Round terminé, au gré de l'application.

### 8.3.3 Algorithme de recherche d'événements recommandé

L'algorithme suivant est recommandé.

#### 8.3.3.1 Premier Event\_Arbitration

Si une collision se produit en réponse à une General\_Event\_Request, le maître exécute un Event\_Arbitration pour isoler un événement.

Le maître envoie à cet effet une Group\_Event\_Request, en s'adressant au Groupe {N-1, 0}.

NOTE La première Group\_Event\_Request s'adresse aux 2048 dispositifs ayant une adresse paire.

#### 8.3.3.2 Event\_Arbitration descendante

En supposant que le maître a envoyé une Group\_Event\_Request au Groupe {N, C}:

- si une seule réponse arrive, le maître renvoie cette réponse;
- si une collision se produit, le maître envoie une Group\_Event\_Request à un groupe dont la taille potentielle équivaut à la moitié de celle du groupe précédent, et dont l'adresse commune est la même:

$$G\_new = G\_old \{M = M - 1, 0 + C\}$$

NOTE 1 Cette Group\_Event\_Request s'adresse à la moitié du groupe précédent dont le bit d'adresse commun de poids fort est "0".

- si un silence suit une Group\_Event\_Request, le maître envoie une Group\_Event\_Request à un groupe dont la taille potentielle équivaut à la moitié de celle du groupe précédent, et dont l'adresse commune est:

$$G\_new = G\_old \{M = M-1, (2^{(12-M)} + C)\}$$

NOTE 2 Cette Group\_Event\_Request suppose que l'événement provient de l'autre groupe, et commence par le demi-groupe dont le bit d'adresse commun de poids fort est "0".

Le maître envoie une Single\_Event\_Request lorsqu'une collision se produit sur le deuxième bit de poids fort de l'Adresse de Dispositif (Group\_Event\_Request avec bit 4=0).

### 8.3.3.3 Event\_Arbitration ascendant

Le maître continue le Event\_Arbitration suivant à un niveau de l'arbre situé en dessous du niveau où la collision la plus récente s'est produite.

Le maître envoie à cet effet la Group\_Event\_Request avec le bit de poids faible de la Group\_Address, fixé à la valeur opposée.

$$G_{\text{new}} = G_{\text{old}} \{M = M/2, 2^{(12-M)} + C\}$$

Si un silence ou une Event\_Identifier\_Response est obtenue en réponse à cette Group\_Event\_Request, le maître envoie une General\_Event\_Request et recommence l'arbitrage à partir du haut.

Si une collision se produit en réponse à la Group\_Event\_Request, le maître augmente la taille de la Group\_Address d'une unité et exécute un Event\_Arbitration sur cette branche:

$$G_{\text{new}} = G_{\text{old}} \{M = M+1, (2^{(12-M)} + C)\}$$

### 8.3.4 Trames de Données de Supervision pour Event\_Arbitration

#### 8.3.4.1 General\_Event\_Request

La General\_Event\_Request doit être constituée de quatre champs de 4 bits chacun, comme le montre la Figure 51:

Les bits 0 à 3 constituent le F\_code = 9;

Les 4 bits suivants constituent le Event\_Mode, codé comme suit:

- EM = '0000'B: answer\_now, NOT new\_round,
- EM = '0001'B: answer\_now, new\_round,
- EM = '0010'B: NOT answer\_now, NOT new\_round,
- EM = '0011'B: NOT answer\_now, new\_round,
- EM = '0100'B à '1111'B: réservé.

Les 4 bits suivants constituent le Event\_Type, codé comme suit:

- ET = '0000'B: Event\_Round de priorité élevée (high\_priority)
- ET = '0001'B: Event\_Round de basse priorité (low\_priority)
- ET = '0010'B à '1111'B: réservé.

Les 4 derniers bits sont réservés.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	1	0	0	QT	NR	0	0	0	HP	0	0	0	0
F_code = 9				Event_Mode				Event_Type				réservé			

**Figure 51 – Format de trame de General\_Event\_Request**

NOTE General\_Event\_Request est la seule Trame-Maître sans adresse, puisqu'elle est diffusée par défaut à tous les dispositifs.

#### 8.3.4.2 Group\_Event\_Request

Le maître doit interroger un groupe de dispositifs appartenant au groupe {M, C} à l'aide d'une Group\_Event\_Request formée de trois champs, comme le montre la Figure 52:

- les bits 0 à 3 sont le F\_code = 13;
- les bits suivants, qui commencent par le bit 4, constitués d'un ensemble de "1" contigus fermé par un "0", forment le masque. L'emplacement du bit "0" dépend de l'étape dans l'arbre de recherche. Le nombre de "1" est égal à M-1;
- les bits d'ordre inférieur restants à droite de zéro jusqu'au bit 15 définissent l'adresse de groupe C.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	1	1	1	1	1	1	0	A	B	C	D	E	F
F_code = 13				masque (négliger)						Group_Address					

**Figure 52 – Trame de Group\_Event\_Request (M = 6, C = ABCDEF)**

EXEMPLE En supposant que tous les dispositifs ont un événement en suspens, si l'adresse est:

1111 1111 1110, tous les dispositifs répondent (= General\_Event\_Request {NOT new\_round, answer\_now});  
 1111 1111 1101, tous les dispositifs ayant une adresse impaire répondent;

1111 1001 1001, tous les dispositifs ayant une adresse terminée par 01 1001 répondent;

0111 0110 1000, deux dispositifs qui diffèrent par leur bit de poids fort répondent.

Dans la Figure 52 ci-dessus, le Groupe est {M = 6, C = ABCDEF} et il s'adresse à 64 dispositifs.

#### 8.3.4.3 Single\_Event\_Request

Le maître doit interroger un dispositif unique au sujet de son événement en envoyant une Single\_Event\_Request, qui est une Trame-Maître dont la structure est la suivante, comme le montre la Figure 53:

- les bits 0 à 3 sont le F\_code = 14;
- les 12 bits suivants définissent une adresse de dispositif individuelle.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	0	A	A	A	A	A	A	A	A	A	A	A	A
F_code = 14				Adresse de Dispositif											

**Figure 53 – Trame de Single\_Event\_Request**

#### 8.3.4.4 Event\_Identifier\_Response

Un esclave ayant un événement en suspens dont la priorité a été établie au début du Event\_Round doit répondre en envoyant une Event\_Identifier\_Response à:

- une General\_Event\_Request {answer\_now};
- une Group\_Event\_Request ayant l'adresse de groupe correspondant à la sienne; ou
- une Single\_Event\_Request dont l'adresse correspond à l'Adresse de Dispositif de 12 bits.

La Event\_Identifier\_Response doit être une Trame-Esclave de 16 bits dont le contenu est interprété comme un F\_code de 4 bits suivi par une adresse de 12 bits, comme le montre la Figure 54.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
F_code				adresse logique ou physique											

**Figure 54 – Trame de la Event\_Identifier\_Response**

Un esclave ne doit pas utiliser les F\_codes suivants dans son Event\_Identifier\_Response:

- F\_code = 8 (Mastership\_Transfer);
- F\_code = 9 (General\_Event\_Request);
- F\_code = 13 (Group\_Event\_Request);
- F\_code = 14 (Single\_Event\_Request).

#### 8.3.4.5 Event\_Read\_Request

Le maître doit envoyer à nouveau une Event\_Identifier\_Response correctement reçue comme Event\_Read\_Request.

La Event\_Read\_Request doit avoir le même format que la Event\_Identifier\_Response, à la seule différence qu'elle a un Délimiteur de Début de Trame-Maître au lieu d'un Délimiteur de Début de Trame-Esclave.

#### 8.3.4.6 Event\_Read\_Response

Le format de la Event\_Read\_Response est celui de toute Trame-Esclave.

NOTE La Event\_Read\_Response sera le plus vraisemblablement une trame de Données de Messagerie.

### 8.4 Devices\_Scan

#### 8.4.1 Device\_Status

Tous les dispositifs doivent mettre en œuvre un port émetteur pour leur Device\_Status, contenant un mot de 16 bits qui indique leur statut, et qui peut être interrogé.

##### 8.4.1.1 Device\_Status\_Request

Le maître doit interroger le Device\_Status d'un dispositif en envoyant une Device\_Status\_Request, qui est une Supervisory\_Data\_Request avec un F\_code = 15, et qui contient l'adresse de 12 bits du dispositif, comme le montre la Figure 55.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	1	A	A	A	A	A	A	A	A	A	A	A	A
F_code = 15				Adresse de Dispositif											

Figure 55 – Device\_Status\_Request

##### 8.4.1.2 Device\_Status\_Response

###### 8.4.1.2.1 Format général

Un esclave doit répondre à une Device\_Status\_Request en envoyant une Device\_Status\_Response, qui est une Supervisory\_Data\_Response formée des champs suivants, comme le montre la Figure 56.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SP	BA	GW	MD	T	T	T	T	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capacités				class_specific				common_flags							

Figure 56 – Device\_Status\_Response

#### 8.4.1.2.2 Champ des capacités

Les quatre premiers bits doivent définir les capacités de base d'un dispositif:

- a) SP bit 0 mis à "0" indique les capacités Device\_Status et Process\_Data  
bit 0 mis à "1" indique un dispositif spécial, les 3 bits suivants ne sont pas spécifiés;
- b) BA bit 1 mis à "1" indique la capacité Administrateur de Bus;
- c) GW bit 2 mis à "1" indique la capacité Passerelle;
- d) MD bit 3 mis à "1" indique la capacité Données de Messagerie.

#### 8.4.1.2.3 Champ spécifique

Ce champ spécifique de 4 bits dépend du champ des capacités. Il caractérise quatre champs spécifiques:

- a) dispositif de Classe 1;
- b) dispositif doté de la capacité Données de Messagerie;
- c) dispositif doté de la capacité Administrateur de Bus;
- d) dispositif doté de la capacité Passerelle.

##### 8.4.1.2.3.1 Champ spécifique du dispositif de Classe 1

Un dispositif de Classe 1 doit répondre en ayant son champ des "capacités" mis à "0000", comme le montre la Figure 57.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capacités				class_specific				common_flags							

Figure 57 – Device\_Status d'un dispositif de Classe 1

Le champ spécifique doit être "0000".

Deux bits de Device\_Status sont fixes:

- SER = 0 (pas de réservation possible pour le service);
- FRC = 0 (pas de forçage possible).

##### 8.4.1.2.3.2 Champ spécifique de Données de Messagerie

Un dispositif capable de communiquer des Données de Messagerie conformément aux Protocoles en Temps Réel du Réseau Embarqué de Train doit répondre avec le Bit 3 mis à "1", comme le montre la Figure 58.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	x	x	1	x	X	x	x	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capacités				class_specific				common_flags							

Figure 58 – Device\_Status d'un dispositif de Classes 2/3/4/5

##### 8.4.1.2.3.3 Champ spécifique de l'Administrateur de Bus

Un dispositif doté de la capacité Administrateur de Bus doit répondre par le Bit 1 mis à "1", et positionner à 1 les quatre bits dans le champ "class\_specific", comme le montre la Figure 59:

- AX1: Bit 4, second bit de poids faible de la Actualisation\_Key de la Liste Périodique;

- AX0: Bit 5, bit de poids faible de la Actualisation\_Key de la Liste Périodique;
- ACT: Bit 6 est mis à "1" si le dispositif est actualisé et en état de devenir maître;
- MAS: Bit 7 est mis à "1" si le dispositif est actuellement maître.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	x	x	AX1	AX0	ACT	MAS	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capacités				Class_specific				common_flags							

**Figure 59 – Device\_Status d'un dispositif ayant la capacité Administrateur de Bus**

#### 8.4.1.2.3.4 Champ spécifique de Passerelle

Un dispositif doté de la capacité Passerelle doit répondre avec son Bit 2 mis à "1".

Si le dispositif est également doté de la capacité Administrateur de Bus, le champ "specific" (« spécifique ») doit être celui d'un Administrateur de Bus, comme le montre la Figure 60.

Si le dispositif est une passerelle sans la capacité Administrateur de Bus, le dispositif doit définir le champ "class\_specific" comme le montre la Figure 60:

- STD: le Bit 4 mis à "1" indique une perturbation statique (bus distant en panne);
- DYD: le Bit 5 mis à "1" indique une perturbation dynamique (par exemple, inauguration du train);
- RV6 le Bit 6 est "0" (réservé); et
- RV7 le Bit 7 est "0" (réservé).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	x	1	x	STD	DYD	RV6	RV7	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capacités				class_specific				common_flags							

**Figure 60 – Device\_Status d'un dispositif ayant la capacité Passerelle**

NOTE Les bits STD et DYD peuvent être lus directement sur une passerelle ayant la capacité Administrateur de Bus.

#### 8.4.1.2.4 Champ commun

Ces huit bits sont communs à tous les dispositifs:

LAT: Line\_A\_Trusted:

doit être affirmé si la Trame-Maître de ce télégramme a été reçue sur la Line\_A, et nié si elle a été reçue sur la Line\_B;

RLD: Redundant\_Line\_Disturbed:

doit être mis à 1 si la Observed\_Line est perturbée (5.2.5);

SSD: Some\_System\_Disturbance:

doit être mis à 1 lorsque la supervision du niveau de rafraîchissement d'un Port quelconque s'est déclenchée, et mis à 0 lorsque tous les Ports configurés ont fonctionné dans des conditions normales;

SDD: Some\_Device\_Disturbance:

doit être mis à 1 par un mauvais fonctionnement du dispositif (par exemple: erreur de somme de contrôle de la ROM ou de la RAM) ou par une défaillance extérieure au dispositif (par exemple: capteurs endommagés, ...) et mis à 0 une fois la défaillance éliminée;

ERD: Extended\_Reply\_Delay:

doit être affirmé si  $T\_ignore > T\_reply\_def$ , et nié dans le cas contraire (6.2.4.2);

FRC: dispositif forcé:

doit être affirmé si un Port quelconque a été forcé, à une valeur imposée et nié si aucun des Ports raccordés au MVB n'a été forcé;

DNR: Device\_Not\_Ready:

doit être affirmé si le dispositif n'est pas opérationnel (par exemple, application ne fonctionnant pas) mais capable de fonctionner normalement sur le bus, et nié si le dispositif est opérationnel;

SER: Système Réserve:

doit être mis à 1 si le dispositif a été réservé pour un usage exclusif et mis à 0 lorsque cet usage exclusif est suspendu ou arrive à expiration.

#### 8.4.2 Protocole du Device\_Status

Lorsqu'il reçoit une Trame-Maître avec le  $F\_code = 15$  (Device\_Status\_Request) à sa propre adresse, un dispositif doit répondre en envoyant sa Device\_Status\_Response.

NOTE 1 La destination des trames Device\_Status est normalement le maître, mais il peut également s'agir d'un dispositif de supervision qui fonctionne par invitation.

NOTE 2 La lecture du Device\_Status peut causer la remise à "0" du bit RLD des dispositifs de Classe 1 et de certains autres dispositifs, voir 5.2.5.

#### 8.4.3 Protocole du Devices\_Scan

NOTE Un maître configuré dispose d'une liste des dispositifs configurés par Network\_Management, la Known\_Devices\_List. La liste des dispositifs trouvés, la Devices\_List peut différer de la Known\_Devices\_List en raison d'une défaillance d'un dispositif ou d'une erreur de configuration.

Pour surveiller le bus, le maître doit régulièrement interroger le Device\_Status de tous les dispositifs et constituer la Devices\_List qui contient l'adresse et le Device\_Status de chaque dispositif trouvé.

Le maître doit établir la liste des dispositifs ayant la capacité Administrateur de Bus, qu'il trouve dans la Bus\_Administrators\_List.

Le maître doit scruter les dispositifs connus dans sa Known\_Devices\_List, ainsi que les dispositifs inconnus, pour vérifier l'insertion de nouveaux dispositifs.

La fréquence de scrutin relative aux dispositifs connus et inconnus est un paramètre de configuration.

Il est recommandé que le Devices\_Scan soit réalisé à la fréquence de 64 dispositifs toutes les 512,0 ms.

Un dispositif qui cesse de répondre pendant trois interrogations consécutives doit être retiré de la Devices\_List.

### 9 Transfert de maîtrise

NOTE Le Transfert de Maîtrise décrit les protocoles qui sélectionnent un maître parmi plusieurs administrateurs de bus, et garantit le transfert de maîtrise à la fin d'un tour ou à l'apparition d'une défaillance.



## 9.1 Opération de transfert de maîtrise

### 9.1.1 Configuration de l'administrateur de bus

#### 9.1.1.1 Bus\_Configuration

Les informations dont un administrateur de bus a besoin pour un fonctionnement normal (T\_reply, Liste Périodique, etc.), doivent être contenues dans la structure de données Bus\_Configuration.

La Bus\_Configuration doit être identifiée par un Actualisation\_Index qui identifie sa version et qui doit être incrémenté de un à chaque nouvelle version chargée.

La Bus\_Configuration peut de plus être identifiée par une Actualisation\_Key de 15 bits, qui identifie sa version de la configuration et qui (si elle est utilisée) doit être incrémentée de un pour chaque nouvelle version chargée sur l'administrateur de bus. Les deux bits de poids faible de l'Actualisation\_Key peuvent être utilisés comme Actualisation\_Index.

Un administrateur de bus disposant d'une Bus\_Configuration (10.4.3) est dit "configuré" (« configuré »). Il doit mettre le bit ACT de son Device\_Status à "1".

Un administrateur de bus "unconfigured" (« non configuré ») exécute uniquement le Event\_Round et le Devices\_Scan. Il doit mettre le bit ACT de son Device\_Status à "0".

Un maître doit supposer qu'un autre administrateur de bus est "actualised" (« actualisé ») si l'Actualisation\_Index de ce dernier est égal à son propre indice ou est plus élevé d'une unité.

Un administrateur de bus doit retourner à l'état "unconfigured" (« non configuré ») si l'Actualisation\_Index a été modifié quatre fois de suite sans que l'administrateur de bus ne soit devenu maître.

NOTE La génération de l'Actualisation\_Key relève de la Gestion de Réseau.

#### 9.1.1.2 Bus\_Administrators\_List

Chaque administrateur de bus doit détenir une liste circulaire de tous les autres administrateurs de bus, appelée Bus\_Administrators\_List, ordonnée lorsque le jeton circule, et de préférence classée dans l'ordre ascendant des adresses de dispositif.

NOTE La Bus\_Administrators\_List peut faire partie intégrante de la Bus\_Configuration, ou peut être générée en ligne par le Devices\_Scan.

## 9.2 Spécifications relatives au transfert de maîtrise

NOTE Les états de l'administrateur de bus sont décrits dans la syntaxe SDL de la Figure 61.

### 9.2.1 Etats

#### 9.2.1.1 État "STANDBY\_MASTER"

Dans cet état, l'administrateur de bus n'exerce pas la maîtrise. Il doit passer à l'état "REGULAR\_MASTER":

- s'il ne reçoit pas de Trames-maîtres au cours d'une temporisation T\_standby, ou
- s'il reçoit une offre de maîtrise du maître actuel à laquelle il peut répondre (ce qui signifie que l'administrateur de bus est "actualised" (« actualisé »)).



### 9.2.1.2 État "REGULAR\_MASTER"

Dans cet état, l'administrateur de bus doit exercer la maîtrise (il exécute le Scrutin Périodique, le Scrutin d'Événements et le Devices\_Scan).

S'il détecte une Trame-Maître qu'il n'a pas envoyée lui-même, l'administrateur de bus doit supposer qu'une collision entre maîtres s'est produite et revenir à l'état "STANDBY\_MASTER".

Après que son tour soit passé ou après une commande "resign" (démission), l'administrateur de bus doit passer à l'état "FIND\_NEXT".

### 9.2.1.3 État "FIND\_NEXT"

Dans cet état, l'administrateur de bus doit aller chercher l'adresse du maître proposé dans sa Bus Administrators\_List et scruter le Device\_Status de celui-ci.

Le maître proposé est un administrateur de bus valide si son Device\_Status indique que:

- a) il est administrateur de bus (BA = 1);
- b) il est configuré (ACT = 1);
- c) il a un Actualisation\_Index égal ou supérieur d'une unité à celui du maître actuel, (Modulo 4).

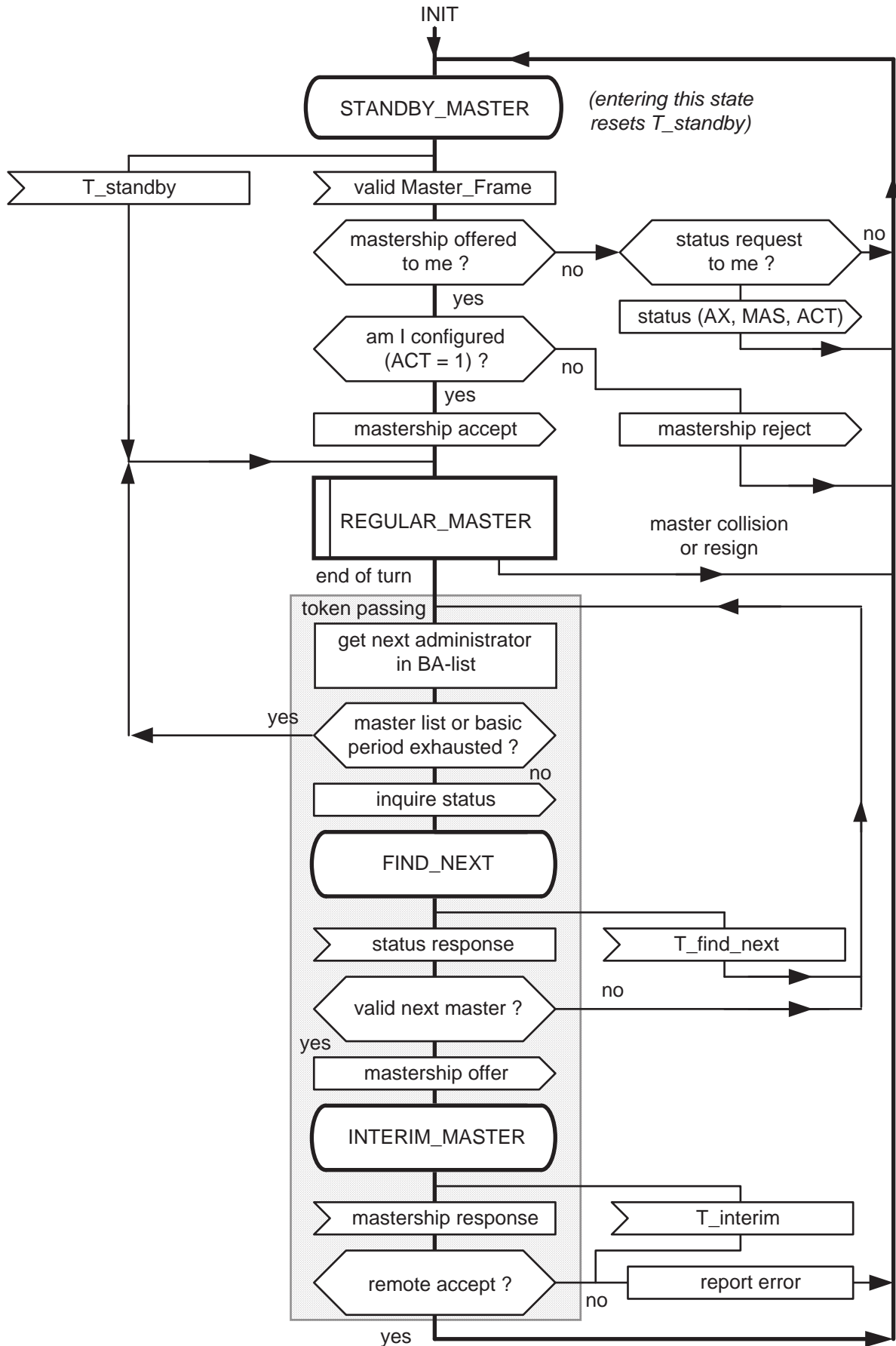
L'administrateur de bus doit envoyer une Mastership\_Transfer\_Request à lui-même puis à l'administrateur de bus valide et passer à l'état "INTERIM\_MASTER".

Sinon, l'administrateur de bus doit retourner à l'état "REGULAR\_MASTER" et il doit vérifier l'administrateur de bus suivant dans sa Bus Administrators\_List au prochain tour.

### 9.2.1.4 État "INTERIM\_MASTER"

Dans cet état, l'administrateur de bus doit attendre la Mastership\_Transfer\_Response provenant de l'administrateur de bus suivant.

Lorsqu'il reçoit une réponse (acceptation ou rejet) ou à l'expiration de la temporisation, l'administrateur de bus doit passer à l'état "STANDBY\_MASTER". Si la réponse de l'autre administrateur de bus est "reject" (« rejet »), l'administrateur de bus concerné doit signaler une erreur.



**Légende**

Anglais	Français
entering this state, resets	entrée dans cet état, met à "0"
valid	valide
mastership offered to me?	La maîtrise m'est proposée ?
yes	oui
no	non
status request to me?	demande d'état (à moi)?
status	état
Am I configured ?	suis-je configuration ?
mastership accept	acceptation de la maîtrise
mastership reject	rejet de la maîtrise
master collision or resign	collision ou renonciation de maître
end of turn	fin de tour
token passing	passage du jeton
get next administrator in BA-list	passe au prochain administrateur dans la liste BA
master list or basic period exhausted?	liste maître ou période de base écoulee ?
inquire status	demande d'état
status response	réponse d'état
mastership offer	offre de maîtrise
mastership response	réponse de maîtrise
remote accept?	acceptation à distance ?
report error	consignation d'erreur
valid next master ?	maître suivant valide ?

**Figure 61 – Etats de transfert de maîtrise****9.2.2 Temporisations pour le transfert de maîtrise****9.2.2.1 Durée d'un tour**

Un tour doit durer un multiple de 1024 ms et ne doit pas dépasser  $256 \times 1024$  ms.

**9.2.2.2 Temporisations de "STANDBY\_MASTER"**

T\_standby doit être égal à:

- $(T_{\text{alive}} \times 2 \times (1 + \text{rank\_in\_ba\_list}))$  pour un administrateur de bus configuré; ou
- $(T_{\text{alive}} \times 2 \times (\text{ba\_adr} + 15))$  pour un administrateur de bus non configuré; ou
- infini pour un administrateur de bus désactivé.

où:

T\_alive est l'intervalle maximum entre les Trames-Maîtres;  
 rank\_in\_ba\_list est le classement de ce dispositif dans la Bus\_Administrators\_List;  
 ba\_adr est l'Adresse de Dispositif de l'administrateur de bus.

NOTE Attribuer les Adresses de Dispositif basses aux administrateurs de bus permet d'accélérer la récupération.

**9.2.2.3 Temporisations de "FIND\_NEXT"**

T\_find\_next doit être égal à T\_reply.

### 9.2.2.4 Temporisations de "INTERIM\_MASTER"

T\_interim doit être égal à T\_reply.

## 9.3 Trames de données de supervision pour le transfert de maîtrise

### 9.3.1 Télégramme de Device\_Status

Un maître doit vérifier si le maître proposé est actif et configuré correctement en lisant le Device\_Status du maître proposé grâce à une Device\_Status\_Request, comme le montre la Figure 62.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
F_code = 15				Adresse de Dispositif du maître proposé											

**Figure 62 – Device\_Status\_Request (envoyée par le maître actuel)**

Le maître proposé doit répondre en envoyant une Device\_Status\_Response, comme le montre la Figure 63.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x	1	x	x	AX1	AX0	ACT	MAS	LAT	RLD	SSD	SDD	ERD	FRC	DNR	SER
capacités				class_specific				common_flags							

**Figure 63 – Device\_Status\_Response (envoyée par le maître proposé)**

### 9.3.2 Télégramme de transfert de maîtrise

Le maître actuel doit envoyer une Mastership\_Transfer\_Request contenant, comme le montre la Figure 64:

- le F\_code = 8;
- l'adresse du maître proposé ou son adresse propre (auto-transfert).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
F_code = 8				Adresse de Dispositif du maître proposé											

**Figure 64 – Mastership\_Transfer\_Request (envoyée par le maître actuel)**

Le maître en poste ou le maître proposé doit répondre en envoyant une Mastership\_Transfer\_Response, qui est une Supervisory\_Data\_Response constituée, comme le montre la Figure 65:

- du Bit 0 = ACP,
  - si ACP = 1, indique que l'administrateur de bus accepte la maîtrise;
  - si ACP = 0, indique que l'administrateur de bus rejette la maîtrise;
- des Bits 1 à 15 qui contiennent la Actualisation\_Key du maître proposé (ou "0" si elle n'est pas utilisée).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ACP	Actualisation_Key														

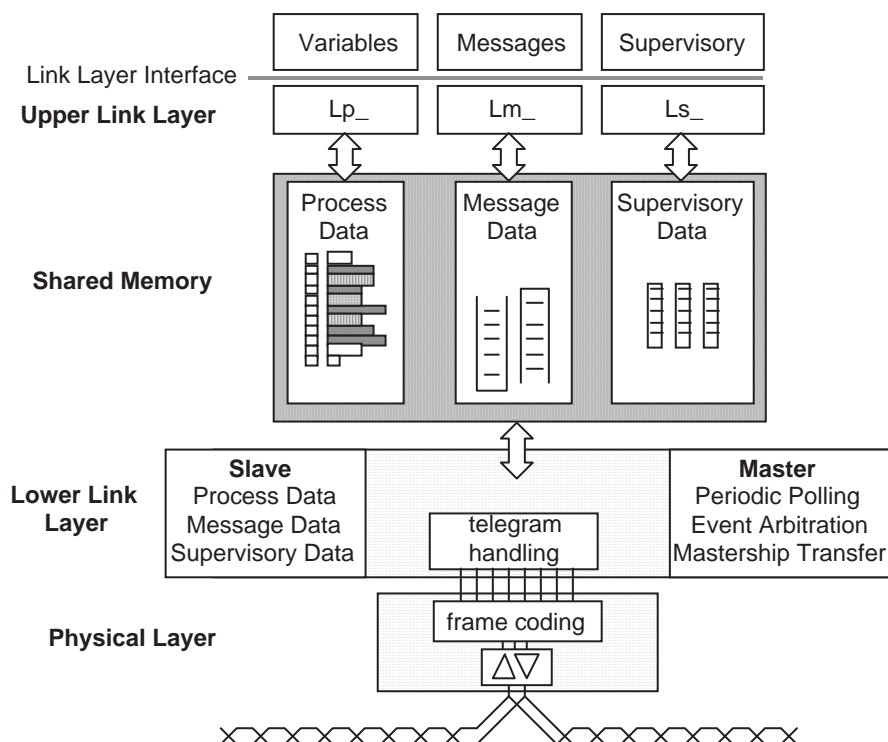
**Figure 65 – Mastership\_Transfer\_Response (envoyée par le maître suivant proposé)**

## 10 Interface de Couche de Liaison

### 10.1 Organisation de la Couche de Liaison

L'Interface de Couche de Liaison du MVB doit offrir trois services, comme le montre la Figure 66:

- a) l'Interface de Liaison de Données de Processus (LPI), qui fournit un ensemble de Ports dans lesquels sont stockées les Données de Processus. Cette interface est spécifiée à l'Article 6 de la CEI 61375-2-1 et seuls les paramètres spécifiques au MVB sont spécifiés dans la présente norme;
- b) l'Interface de Liaison de Données de Messagerie (LMI), qui fournit un ensemble de queues (queues d'attente) pour les Données de Messagerie, utilisée par les couches supérieures (réseau, transport, session). Cette interface est spécifiée à l'Article 6 de la CEI 61375-2-1 et seuls les paramètres spécifiques au MVB sont spécifiés dans la présente norme;
- c) l'Interface de Supervision de Liaison (LSI), qui fournit les services de configuration de la couche de liaison et de supervision du bus. Cette interface est spécifique au MVB et est spécifiée en 10.4.



#### Légende

Anglais	Français
Link Layer Interface	Interface de Couche de Liaison
Supervisory	De supervision
Upper Link Layer	Couche de Liaison Supérieure
Process Data	Données de Processus
Message Data	Données de Messagerie
Supervisory Data	Données de Supervision
Shared Memory	Mémoire Partagée
Lower Link Layer	Couche de Liaison Inférieure
Slave	Esclave
telegram handling	traitement des télégrammes
Master	Maître
Periodic Polling	Scrutin Périodique
Event Arbitration	Arbitrage des Evénements
Mastership Transfer	Transfert de Maîtrise
Physical Layer	Couche Physique
frame coding	codage des trames

**Figure 66 – Organisation de la Couche de Liaison**

## 10.2 Interface de Liaison de Données de Processus

L'Interface de Données de Processus est spécifiée à l'Article 6 de la CEI 61375-2-1. Les paramètres spécifiques au MVB sont décrits ci-après.

La longueur d'un Dataset doit être limitée à 256 bits.

La Port\_Address d'un dataset doit comporter 12 bits. Elle doit être la même que l'adresse utilisée pour transmettre le dataset par le bus.

Le nombre de Datasets par Traffic\_Store doit être limité à 4095.

Le dataset dont l'adresse est 4095 doit être réservé pour la synchronisation et la répartition temporelle.

Lorsqu'il est nécessaire d'avoir une synchronisation exacte, la fin de la Trame-Maître doit servir de point de référence.

NOTE Un Dataset est un groupe de variables transmises dans la même trame de Données de Processus.

### **10.3 Interface de Liaison de Données de Messagerie**

#### **10.3.1 Généralités**

L'Interface de Liaison de Données de Messagerie fournit des services relatifs à l'envoi de trames de Données de Messagerie et à la récupération des Données de Messagerie reçues.

L'Interface de Liaison de Données de Messagerie fournit le service de base sur lequel reposent les services de messagerie.

L'Interface de Liaison de Données de Messagerie (LMI) est spécifiée à l'Article 6 de la CEI 61375-2-1. Les paramètres spécifiques au MVB sont décrits ci-après.

Cette interface n'est pas mise en œuvre dans les dispositifs qui n'ont pas la capacité Données de Messagerie.

#### **10.3.2 Priorité**

L'Interface de Liaison de Données de Messagerie repose sur une queue d'émission, qui peut être raccordée à l'un ou l'autre des deux niveaux de priorité (ET = 0 et ET = 1) de l'accès apériodique au support, et sur une queue de réception, qui n'a pas de priorité.

L'attribution des priorités dépend de l'application.

#### **10.3.3 Taille des paquets**

Le champ "SZ" d'un paquet vide doit être mis à 0.

La longueur maximale que doit indiquer le champ "SZ" est de 27 (32 octets – 5 octets d'en-tête de liaison, y compris 1 octet pour le champ "SZ").

#### **10.3.4 Protocol\_Type**

La valeur du Protocol\_Type pour les Protocoles en Temps Réel doit être '1000'B. D'autres valeurs pour le Protocol\_Type sont réservées pour un usage futur.

#### **10.3.5 Protocole de Transport de Messages**

Un dispositif MVB doit annoncer une taille de paquet de 27 octets lorsqu'il ouvre une connexion.

Lorsqu'il répond à une ouverture de connexion, un dispositif du MVB doit spécifier une taille de paquet égale à 27 octets.

## 10.4 Interface de Supervision de Liaison

### 10.4.1 Généralités

L'Interface de Supervision de Liaison est spécifique au MVB.

Elle fournit des services généraux pour la configuration et l'inspection de la Couche de Liaison et la signalisation d'événements.

L'Interface de Supervision de Liaison est regroupée en 6 services:

- a) MVB\_Status: indique l'état de la couche de liaison;
- b) MVB\_Control: contrôle le fonctionnement du dispositif;
- c) MVB\_Devices: fournit le Device\_Status des autres dispositifs du même bus;
- d) MVB\_Administrator: met en place l'administrateur de bus;
- e) MVB\_Report: signalisation d'événements.

Il est admis d'accéder à ces objets par des procédures d'interface ou par câble. Quelle que soit la méthode employée, un dispositif doit pouvoir présenter la fonctionnalité décrite.

Les procédures de cette interface ont le préfixe ls\_v\_ (supervision de liaison, bus de véhicule).

Le format des paramètres des procédures d'interface suivantes n'est pas indiqué. Toutefois, le document Train Network\_Management (Article 8 de la CEI 61375-2-1) propose un format recommandé comme format de paramètres.

### 10.4.2 Procédures de l'Interface de Supervision de Liaison

Les services sont définis par des structures de données qui peuvent être lues et écrites à l'aide d'une procédure de la forme suivante:

Définition	Procédure d'accès à la couche de liaison MVB
Syntaxe	LS_RESULT                      ls_v_<service_name> (<parameters>)

Les résultats possibles figurent au Tableau 9.

**Tableau 9 – Codage LS\_RESULT**

Constante	Code	Signification
LS_OK	0	succès
LS_BUSY	1	essayer à nouveau ultérieurement
LS_CALL_SEQUENCE	2	séquence de commande erronée

Seuls les paramètres des procédures sont définis ici.

### 10.4.3 MVB\_Status

L'objet MVB\_Status décrit les paramètres statiques et dynamiques de la couche de liaison. Cette structure est destinée à être transmise sans modifications.



Chaque dispositif doit avoir un objet MVB\_Status doté des attributs figurant au Tableau 10.

**Tableau 10 – Objet MVB\_Status**

Attribut	Type	Signification
device_address	WORD16	adresse du dispositif sur le bus. Quatre bits de poids fort = '0000'B.
mvb_hardware_name	STRING32	identifie le contrôleur MVB, le matériel et sa version.
mvb_software_name	STRING32	identifie la version de logiciel MVB, de préférence avec le format suivant: xxxx-Vz.z-dd.mm.yy.
device_status	Device_Status	copie du Device_Status à 16 bits.
réservé	WORD8 (=0)	utilisé pour l'alignement.
t_ignore	UNSIGNED8	valeur de la temporisation configurée pour le délai entre une Trame-Maître et une Trame-Esclave sur ce dispositif en multiples de 1 µs. si t_ignore = 0, la valeur est par défaut T_reply_def = 42,7 µs.
lineA_errors	UNSIGNED32	compteur de 32 bits incrémenté à chaque fois que le signal Line_Disturbed de la Line_A est mis à "1"; ce compteur ne revient pas à zéro lorsqu'il atteint sa valeur la plus élevée.
lineB_errors	UNSIGNED32	compteur de 32 bits incrémenté à chaque fois que le signal Line_Disturbed de la Line_B est mis à "1"; ce compteur ne revient pas à zéro lorsqu'il atteint sa valeur la plus élevée.

#### 10.4.4 MVB\_Control

L'objet MVB\_Control contrôle les paramètres statiques et dynamiques de la couche de liaison. Cette structure est destinée à être transmise sans modifications.

Chaque dispositif doit avoir un objet MVB\_Control comprenant les paramètres figurant au Tableau 11.

**Tableau 11 – Objet MVB\_Control**

Attribut		Type	Signification
device_address		WORD16	adresse du dispositif sur le bus, avec les quatre bits de poids fort = 0 (si modifiable par logiciel)
t_ignore		UNSIGNED16	valeur de la temporisation T_ignore à utiliser, en multiples de 1 µs valeur par défaut (t_ignore = 0): T_reply_def = 42,7 µs valeurs recommandées: 22,0 µs, 64,0 µs et 86,0 µs valeur maximale: 255,0 µs.
command		BITSET8	commande à exécuter par la Couche de Liaison du MVB:
	aon	(0)	(Administrateur de Bus uniquement) active l'administrateur de bus
	aof	(1)	(Administrateur de Bus uniquement) désactive l'administrateur de bus
	spl	(2)	(Administrateur de Bus uniquement) passage à la nouvelle Liste Périodique (=1). Provoque le passage de l'administrateur de bus à la nouvelle Liste Périodique la prochaine fois que l'administrateur de bus devient maître
	tms	(3)	(Administrateur de Bus uniquement) transfert de la maîtrise à l'administrateur de bus suivant dans la Bus_Administrators_List, à la fin du macro-cycle suivant la demande. Conserve la maîtrise si l'autre administrateur de bus ne l'accepte pas.
	sla	(4)	oblige à recevoir à partir de la Line_A, même si cette ligne est perturbée (sla = slb = 0: pas d'action)
	slb	(5)	oblige à recevoir à partir de la Line_B, même si cette ligne est perturbée (sla = slb = 1: fonctionnement normal redondant)
	cla	(6)	réinitialise le compteur d'erreurs de trames de la Line_A (cla = clb = 0: pas d'action)
	clb	(7)	réinitialise le compteur d'erreurs de trames de la Line_B (cla = clb = 1: remet les deux compteurs à "0")

NOTE Pour les dispositifs qui ne sont pas des administrateurs de bus, les lignes redondantes sont commutées lorsque l'administrateur de bus lit le Device\_Status.

#### 10.4.5 MVB\_Devices

L'objet MVB\_Devices comprend notamment la liste des dispositifs et leur Device\_Status dans l'ordre croissant des adresses, indiquant pour chacun:

- l'Adresse de Dispositif;
- le Device\_Status lu lors du dernier Devices\_Scan.

Chaque administrateur de bus doit avoir un objet MVB\_Devices doté des attributs figurant au Tableau 12.

**Tableau 12 – Objet MVB\_Devices**

Attribut		Type	Signification
nr_devices		UNSIGNED16	nombre de dispositifs dans la Devices_List.
devices_list		ARRAY [nr_devices] OF	liste des dispositifs trouvés, comprenant pour chacun:
	device_address	WORD16	adresse du dispositif formée de 12 bits (quatre bits de poids fort = 0)
	device_status	Device_Status	copie du Device_Status du dispositif formé de 16 bits.

#### 10.4.6 MVB\_Administrator

L'objet MVB\_Administrator comprend en particulier:

- les paramètres de configuration (Période de Base, T\_reply, stratégie d'Événement, etc.);
- la Known\_Devices\_List et l'algorithme Devices\_Scan;
- la Liste Périodique, constituée des Cycle\_Lists et des Split\_Lists;
- la Bus\_Administrators\_List.

Chaque administrateur de bus doit avoir un objet MVB\_Administrator doté des attributs figurant au Tableau 13.

**Tableau 13 – Objet MVB\_Administrator**

Attribut	Type	Signification
checkword0	UNSIGNED16	Somme de contrôle sur cette structure – dépendant de l'application. Ce mot est situé au point de décalage 0 de cette structure.
actualisation_key	UNSIGNED16	Numéro de version de la configuration, sous forme d'un nombre à 16 bits.  Le bit de poids fort mis à "1" indique que les 15 bits de poids faible doivent être transmis dans la Mastership_Transfer_Response.  Les deux bits de poids faible sont copiés sur AX0 et AX1.
t_reply_max	UNSIGNED16	délai de réponse maximum prévisible entre une Trame-Maître et une Trame-Esclave, en multiples de 1 µs  valeur par défaut (t_reply = 0) = 42,7 µs, valeur maximale = 255,0 µs.  si le mode Extended Reply Delay est utilisé, cette valeur doit être supérieure ou égale à la valeur la plus élevée de la temporisation T_ignore de tout dispositif dans cette configuration.
macro_cycles	UNSIGNED16	Nombre de périodes de base à l'issue desquelles le trafic périodique se répète (par exemple 512 pour une période de base de 2,0 ms T_bp).
event_poll_strategy	ENUM16	'0000'H pas d'interrogation d'événements '4000'H interrogation d'événements à priorité élevée uniquement '8000'H interrogation d'événements à basse priorité uniquement 'C000'H interrogation d'événements à priorité élevée et basse.
basic_period	UNSIGNED16	T_bp en microsecondes (valeurs recommandées: 1000, 1250, 1042, 2000, 2043, 2500).
macrocycles_per_turn	UNSIGNED16	Nombre de Macro_Cycles après lesquels le maître doit proposer la maîtrise à l'administrateur de bus suivant figurant sur sa liste.
devices_scan_strategy	ENUM16	0: scrutin de toutes les adresses de dispositifs connues 1: scrutin de toutes les adresses de dispositif
reserved2	WORD16	non utilisé
reserved3	WORD16	non utilisé
reserved4	WORD16	non utilisé
reserved5	WORD16	non utilisé
known_devices_list_offset (offset_KDL)	UNSIGNED16	Décalage (à partir du checkword0) où est située la première adresse de la liste des dispositifs.  La taille de la liste est égale à offset_RSL – offset_KDL.
reserved_list_offset (offset_RSL)	UNSIGNED16	non utilisé (offset_RSL= offset_PL)
periodic_list_offset (offset_PDL)	UNSIGNED16	décalage (à partir du checkword0) où sont situés le premier F_code et l'adresse du premier cycle. La taille de la liste est égale à offset_BAL – offset_PDL.

**Tableau 13 (suite)**

Attributs		Type	Signification
bus_administrators_list_offset (offset_BAL)		UNSIGNED16	décalage (à partir du checkword0) où est située la première adresse de la Bus_Administrators_List.
devices_scan_list_offset (offset_DSL)		UNSIGNED16	décalage (à partir du checkword0) où est située la liste d'attribution du balayage des dispositifs (non utilisé)
end_list_offset (offset_END)		UNSIGNED16	décalage fictif qui indique la taille de la devices_scan_list et montre le prochain emplacement après le dernier dispositif de cette liste.
known_devices_list		ARRAY [nr_devices] OF	liste des dispositifs connus, (nr_devices = offset_KDL – offset_RSL), comprenant chacun:
	device_address	WORD16	adresse du dispositif formée de 12 bits (quatre bits de poids fort =0)
cycle_lists_offsets		ARRAY [11] OF	ensemble de 11 éléments comprenant pour chacun des 11 cycles possibles:
	cycle_n_offset	UNSIGNED16	décalage (à partir du début de la Liste Périodique) où est située la première adresse du cycle
split_list_offsets		ARRAY [5] OF	ensemble de 5 éléments, comprenant chacun:
	split_n_m_offset	UNSIGNED16	décalage (à partir du début de la Liste Périodique) où est située la première adresse du tableau split_n_m
cycle_lists		ARRAY [11] OF	ensemble de 11 listes, comprenant chacune
	cycle_n	ARRAY [cycle_size] OF	ensemble de valeurs [cycle_size], (cycle_size est le décalage entre cette liste et la suivante) comprenant chacune:
	f_code	ENUM4	F_code (valeur quelconque) de la Trame-Maître à envoyer
	address	UNSIGNED12	adresse dans la Trame-Maître
split_lists		ARRAY [5] OF	ensemble de cinq listes (n= 4, n=16, n=64, n= 256, n= 1024) et m = n/2, comprenant chacune:
	split_n_m	ARRAY [n] OF	ensemble de [n] valeurs, comprenant chacune:
	lower_count (n)	UNSIGNED8	nombre de trames du cycle inférieur (n) à mettre dans chaque Période de Base de l'octet de poids fort (cette liste apparaît deux fois, sa taille étant la moitié de celle de la liste 'upper_count').
	upper_count (m)	UNSIGNED8	nombre de trames du cycle supérieur (m) à mettre dans chaque Période de Base
bus_administrators_list		ARRAY [nr_admins] OF	liste des administrateurs de bus connus, (nr_admins = devices_scan_list_offset – bus_administrators_list_offset), comprenant chacun:
	f_code	UNSIGNED4	F_code = 8 (prêt pour le transfert de maîtrise)
	device_address	UNSIGNED12	adresse du dispositif formée de 12 bits.

#### 10.4.7 MVB\_Report

Définition	L'application reçoit des rapports en appelant une fonction donnée du type:
Syntaxe	<pre>typedef void      (* ls_v_report)                   (ls_report)</pre>

La valeur des rapports doit être conforme aux codes du Tableau 14.

**Tableau 14 – Codage du LS\_V\_REPORT**

Constante	Code	Signification
LS_V_FRAME_COUNT	1	le nombre de trames reçues a atteint 65526.
LS_V_HW_ERROR	2	une perturbation s'est produite
LS_V_CHANGE	3	(Administrateur de Bus uniquement), modification de la liste des dispositifs
LS_V_REPEATED_EVENT	4	(Administrateur de Bus uniquement), un dispositif signale toujours le même événement
LS_V_SYNCHRO	5	synchronisation reçue
LS_V_MASTER_CHANGE	6	ce dispositif a transféré la maîtrise.

## 11 Protocoles en Temps Réel

Dans l'architecture TCN, les bus WTB et MVB partagent les mêmes Protocoles en Temps Réel, qui offrent deux services de communication:

- a) le service de Variables de Processus, offert par une base de données en temps réel distribuée, mise à jour périodiquement par diffusion;
- b) le service de messagerie, offrant le transfert de messages à la demande, sous forme de:
  - messages point à point, et/ou
  - messages distribués ou publipostés.

Ces services sont définis en détail à l'Article 6 de la CEI 61375-2-1.

## 12 Fonction Passerelle

Vacant.

## 13 Gestion de Réseau

### 13.1 Teneur du présent article

Le Gestionnaire de Réseau de Train spécifie différents services pour aider à la mise en service, aux essais, à l'exploitation et la maintenance d'un Réseau Embarqué de Train, tel que:

- a) l'identification et le contrôle des stations;
- b) la gestion des couches de liaison du Bus de Train et du Bus de Véhicule;
- c) la distribution du routage et de la topographie;
- d) la lecture à distance et le forçage de variables;
- e) le téléchargement.

Les services communs WTB et MVB sont spécifiés à l'Article 8 de la CEI 61375-2-1.

Le présent article définit les services spécifiques au MVB.

## 13.2 Objets gérés par le MVB

### 13.2.1 Objets de liaison MVB

#### 13.2.1.1 Objet MVB\_Status

Chaque Station attachée à un MVB doit mettre en œuvre un objet MVB\_Status en lecture seule.

L'objet MVB\_Status doit identifier la version du matériel et du logiciel, définir les paramètres dynamiques et statiques, révéler les erreurs et les statistiques de la MVB Link\_Layer et contenir le Device\_Status spécifié dans la CEI 61375-3-1 (« Bus de Véhicule Multifonctions »).

NOTE L'Agent accède à cet objet par l'interface de supervision de la couche de liaison du MVB, appelée ls\_v\_xxx.

#### 13.2.1.2 Objet MVB\_Control

Chaque Station avec une connexion MVB prenant en charge la capacité Bus\_Administrator doit mettre en œuvre un objet de MVB\_Control en écriture seule.

Par cet objet, un Gestionnaire peut commander à un Bus\_Administrator:

- a) de lâcher le contrôle du bus de sorte qu'un autre Bus\_Administrator puisse prendre la relève, ou
- b) d'utiliser une liste de scrutin préalablement chargée.

#### 13.2.1.3 Objet MVB\_Configuration

Chaque Station avec une connexion MVB prenant en charge la capacité Bus\_Administrator doit mettre en œuvre l'objet Bus\_Configuration en écriture seule, contenant tous les paramètres exigés pour que le Bus\_Administrator fonctionne.

L'objet Bus\_Configuration doit contenir en particulier les sous-objets suivants traités comme un tout et couverts par un contrôle de version commune:

- a) les paramètres d'initialisation du Bus\_Administrator;
- b) la Known\_Devices\_List indiquant les dispositifs dont il convient qu'ils soient normalement présents;
- c) la Periodic\_List définissant la stratégie de scrutin des Données de Processus;
- d) la Bus\_Administrators\_List définissant l'algorithme de transfert de maîtrise;
- e) la Devices\_List définissant comment les dispositifs connus et inconnus sont explorés.

NOTE 1 La modification de la Periodic\_List en ligne peut mener à une opération non-déterministe, voire causer une perte de trafic si la configuration échoue (par exemple, trop grand nombre de variables incluses).

NOTE 2 Le protocole de transfert de maîtrise assure qu'une fois qu'un Bus\_Administrator démarre en utilisant la nouvelle liste de configuration, tous les autres feront de même.

#### 13.2.1.4 Objet MVB\_Devices

Chaque Station avec une connexion MVB prenant en charge la capacité Bus\_Administrator doit mettre en œuvre un objet MVB\_Devices en lecture seule, contenant la liste des dispositifs présents avec leur Device\_Status.

NOTE 1 Le Bus\_Administrator élabore cette liste en explorant périodiquement les dispositifs.

NOTE 2 Cet objet peut également être exécuté par une Station qui n'est pas un Bus\_Administrator, par l'observation du trafic du bus.

### 13.3 Services MVB et messages de gestion

#### 13.3.1 Services de liaison du MVB

##### 13.3.1.1 Read\_MVB\_Status

###### 13.3.1.1.1 Description

Ce service lit le MVB\_Status, qui décrit l'état d'une couche de liaison MVB à laquelle une Station est connectée.

###### 13.3.1.1.2 Call\_Read\_Mvb\_Status

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 10							
bus_id								reserved1							

```

Call_Read_Mvb_Status:=  RECORD
{
  bus_id                UNSIGNED8 (0..15),  -- indicatif de la couche de
                                          liaison
  reserved1             WORD8 (=0)          -- réservé
}

```

###### 13.3.1.1.3 Reply\_Read\_Mvb\_Status

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 10							
bus_id								reserved1							
reserved2															
device_address															
mvb_hardware_name: STRING32															
								CHARACTER8 ou '00'H							
mvb_software_name: STRING32															
								CHARACTER8 ou '00'H							
device_status															
reserved3								t_ignore							
lineA_errors															
lineB_errors															

```

Reply_Read_Mvb_Status:= RECORD
{
  bus_id                UNSIGNED8 (0..15),  -- indicatif de la liaison
  reserved1             WORD8 (=0),         -- réservé
  reserved2             WORD16 (=0),        -- réservé
  device_address        WORD16,             -- adresse du dispositif sur le
                                          bus
  mvb_hardware_name     STRING32,          -- descripteur du contrôleur
                                          MVB, y compris la version du
                                          matériel
  mvb_software_name     STRING32,          -- descripteur de la version
                                          logicielle MVB, au format:
                                          xxxx-Vz.z-dd.mm.yy
}

```

```

device_status      Device_Status,      -- copie du Device_Status du
                                         MVB formé de 16 bits
reserved3          WORD8 (=0),         -- réservé
t_ignore           UNSIGNED8,          -- valeur de la temporisation
                                         configurée pour le délai
                                         entre une Master_Frame et
                                         une Slave_Frame sur ce
                                         dispositif en multiples de
                                         1,0 µs,
                                         Valeur par défaut (t_ignore
                                         = 0) = 42,7 µs.
lineA_errors       UNSIGNED32,         -- compteur de 32 bits
                                         incrémenté chaque fois
                                         qu'une trame défectueuse de
                                         tout type est reçue sur la
                                         Line_A; ce compteur ne
                                         revient pas à zéro lorsqu'il
                                         atteint sa valeur la plus
                                         élevée.
lineB_errors       UNSIGNED32         -- compteur de 32 bits
                                         incrémenté chaque fois
                                         qu'une trame défectueuse de
                                         tout type est reçue sur la
                                         Line_B; ce compteur ne
                                         revient pas à zéro lorsqu'il
                                         atteint sa valeur la plus
                                         élevée.
}

```

Le Device\_Status est défini par la structure suivante:

```

Device_Status ::= RECORD
{
  capabilities      BITSET4            -- capacités de base d'un
                                         dispositif MVB
                                         (tous les quatre bits à 0 =
                                         classe 1)

  {
    sp              (0),               -- spécial - dispositif de
                                         classe 1
    ba              (1),               -- MVB Bus_Administrator
    gw              (2),               -- passerelle ou noeud du Bus
                                         de Train
    md              (3)                -- capacité de messagerie
  },
  class_specific    ONE_OF             -- Classe 1 (C1),
                                         Administrateur de Bus (BA)
                                         ou passerelle (GW)

  {
    [C1]            WORD4,             -- pour classe 1, = 0
    [BA]            RECORD             -- pour l'administrateur de bus
    {
      act_index     UNSIGNED2          -- deux bits de poids faible de
                                         l'index d'actualisation
      act           BOOLEAN1           -- 1 = actualisé
      mas           BOOLEAN1           -- 1 = maître actuel
    }
    [GW]            RECORD             -- pour une passerelle qui
                                         n'est pas en même temps
                                         administrateur de bus.

    {
      std           BOOLEAN1           -- 1 = perturbation statique
      dyd           BOOLEAN1           -- 1 = perturbation dynamique
      rsv1          BOOLEAN1           -- réservé, =0
      rsv2          BOOLEAN1           -- réservé, =0
    }
  }
}

```



```

    }
  }
  common_flags      BITSET8
  {
    lat              (0)          -- non utilisé = 0
    old              (1)          -- perturbation de la ligne
                                observée
    ssd              (2)          -- Certaine Perturbation du
                                Système (dérangement du
                                processus contrôlé, par
                                exemple perte
                                d'alimentation)
    sdd              (3)          -- Certaine Perturbation du
                                Dispositif
                                (dysfonctionnement du
                                dispositif, par exemple
                                erreur de la somme de
                                contrôle)
    erd              (4)          -- délai de réponse prolongé
                                activé
    frc              (5)          -- dispositif forcé (port forcé
                                dans toute Traffic_Store
                                (Mémoire Partagée) de ce
                                dispositif)
    dnr              (6)          -- Dispositif Non Prêt (par
                                exemple, non initialisé)
    ser              (7)          -- Réservé pour le Service
                                (cette Station est
                                actuellement réservée par un
                                Gestionnaire).
  }
}

```

NOTE Le MVB\_Status est défini en 8.4.3, qui est la référence de la définition des termes en cas de litige.

### 13.3.1.2 Write\_MVB\_Control

#### 13.3.1.2.1 Description

Ce service positionne les paramètres du Bus\_Administrator, et en particulier:

- met en marche et arrête un Bus\_Administrator,
- le fait commuter sur la nouvelle Periodic\_List,
- l'oblige à transférer la maîtrise.

Dans les dispositifs n'ayant pas la capacité Bus\_Administrator, ce service permet une réception sélective uniquement.

#### 13.3.1.2.2 Call\_Write\_Mvb\_Control

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 11							
bus_id								reserved1							
device_address															
reserved2								t_ignore							
reserved3								aon	aof	spl	tms	sla	slb	cla	clb

Call\_Write\_Mvb\_Control ::= RECORD

```

{
bus_id                UNSIGNED8 (0..15), -- indicatif de la liaison
reserved1             WORD8 (=0),         -- réservé
device address        WORD16,             -- L'Adresse de Dispositif doit
                                         être mise à 1
                                         ou 0 si non utilisée.

reserved2             WORD8 (=0)         -- réservé, =0
t_ignore              UNSIGNED8,         -- valeur de la temporisation
                                         T_ignore en multiples de 1,0
                                         µs;
                                         valeur par défaut (t_ignore
                                         = 0): 42,7 µs;
                                         valeurs recommandées:
                                         22,0 µs, 64,0 µs, 86,0 µs,
                                         valeur maximale: 255,0 µs.

reserved3             WORD8 (=0),         -- réservé
command               BITSET8,           -- commande à exécuter
{
    aon                (0)              -- (Administrateur de Bus
                                         uniquement)
                                         active Bus_Administrator

    aof (1)             -- (Administrateur de Bus
                                         uniquement)
                                         désactive l'Administrateur
                                         de Bus
                                         aon = aof = 1: pas d'action

    spl                (2)              -- (Administrateur de Bus
                                         uniquement)
                                         commute sur la nouvelle
                                         Liste Périodique lorsque
                                         l'Administrateur de Bus
                                         devient maître

    tms (3)             -- (Administrateur de Bus
                                         uniquement)
                                         transfère la maîtrise à
                                         l'Administrateur de Bus
                                         suivant dans la
                                         Bus_Administrators_List
                                         (Liste des Administrateurs
                                         de Bus), à la fin du macro-
                                         cycle suivant la demande,
                                         conserve la maîtrise si
                                         l'autre Administrateur de
                                         Bus ne l'accepte pas.

    sla (4)             -- (sla = slb = 0: pas
                                         d'action)
                                         oblige à recevoir à partir
                                         de la Line_A, même si celle-
                                         ci est perturbée;

    slb (5)             -- (sla = slb = 1: redondance
                                         normale)
                                         oblige à recevoir à partir
                                         de la Line_B, même si celle-
                                         ci est perturbée

    cla (6)             -- (cla = clb = 0: pas
                                         d'action)
                                         réinitialise le compteur
                                         d'erreurs de trames de la
                                         Line_A.

    clb (7)             -- (cla = clb = 1: remet les
                                         deux compteurs à zéro)
                                         réinitialise le compteur

```

d'erreurs de trames de la  
Line\_B;

}  
}

NOTE MVB\_Control est défini en 10.4.4, qui est la référence en cas de litige.

13.3.1.2.3 Reply\_Write\_Mvb\_Control

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 11							

Reply\_Write\_Mvb\_Control ::= RECORD

{ }

--

aucun paramètre

13.3.1.3 Read\_MVB\_Devices

13.3.1.3.1 Description

Récupère la liste des dispositifs que le Bus\_Administrator a trouvés sur le bus, avec leur Device\_Status.

13.3.1.3.2 Call\_Read\_Mvb\_Devices

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 12							
bus_id								reserved1							

Call\_Read\_Mvb\_Devices ::= RECORD

{

bus\_id                      UNSIGNED8 (0..15),    -- indicatif de la liaison

reserved1                  WORD8 (=0)            -- réservé

}

13.3.1.3.3 Reply\_Read\_Mvb\_Devices

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 12							
bus_id								reserved1, =0							
device_address															
nr_devices															
devices_list: ARRAY [nr_devices] OF															
device_address (n)															
device_status (n)															

```

Reply_Read_Mvb_Devices ::= RECORD
{
    bus_id                UNSIGNED8 (0..15),    -- indicatif de la liaison
    reserved1             WORD8 (=0),           -- réservé
    device address        WORD16,               -- Adresse de dispositif sur ce
                                                bus
    nr_devices            UNSIGNED16,           -- nombre de dispositifs dans
                                                cette liste
    devices_list          ARRAY [nr_devices] OF
    {
        device status     Device Status        -- copie du Device_Status du
                                                dispositif
    }
}

```

### 13.3.1.4 Write\_MVB\_Administrator

#### 13.3.1.4.1 Description

Pour permettre un accès rapide aux structures des données, la liste de configuration du Bus\_Administrator fonctionne avec des décalages et des index. Ceci s'applique particulièrement aux quatre listes suivantes:

- a) la Known\_Devices\_List;
- b) la Periodic\_List;
- c) la Bus\_Administrators\_List;
- d) la Devices\_List.

Par exemple, la Periodic\_List est identifiée par un décalage situé au mot 24. La taille de la Periodic\_List est implicitement donnée par la différence entre ce décalage (periodic\_list\_offset) et le décalage de la structure suivante (bus\_administrators\_list\_offset) situé au mot 26.

La Periodic\_List est elle-même divisée comme suit:

- les 11 Listes de Cycle, qui énumèrent pour chaque Cycle les trames-maîtres (F\_code et adresse) dans l'ordre dans lequel il convient de les envoyer. Tous les F\_codes sont admis. Chaque Liste de Cycle est nommée selon sa période individuelle, qui est un multiple de  $2^n$  de la période de base, avec  $n = (1..10)$  pour le Cycle\_1, le Cycle\_2, le Cycle\_4,... jusqu'au Cycle\_1024;
- les cinq Split\_Lists (Listes de Scission) qui définissent la répartition réelle des cycles en sous-cycles sur les différentes périodes et qui contiennent le nombre de trames-maîtres à envoyer dans chaque période de base. Un sous-cycle peut ne pas avoir plus de 32 membres. Dans la pratique, il n'est pas possible d'avoir plus de 15 indicatifs dans un sous-cycle.

Pour des raisons d'efficacité, la Split\_List (Liste de Scission) de deux cycles est incluse dans la même structure de données.

EXEMPLE 1 La liste Split\_8\_16 contient dans le premier octet, la Split\_List du Cycle\_8 et, dans le deuxième, la Split\_List du Cycle\_16. Dans la mesure où la Split\_List du Cycle\_8 englobe uniquement 8 périodes, alors que la Split\_List du Cycle\_16 englobe 16 périodes, la Split\_List du Cycle\_8 est répétée deux fois.

Le décalage de la Cycle\_List (Liste de Cycle) et de la Split\_List (Liste de Scission) elle-même n'est pas relatif au commencement du message, mais relatif au premier mot de la Periodic\_List, appelé 'periodic\_list\_offset'.

La taille de chaque liste est donnée par la différence entre son décalage et le décalage de la liste suivante.

EXEMPLE 2 La taille de Cycle\_1 est donnée par la différence: Cycle\_2\_Offset – Cycle\_1\_Offset, la taille de Cycle\_32 étant donnée par Cycle\_64\_Offset – Cycle\_32\_Offset.

Le Tableau 15 donne un exemple d'une liste de MVB\_Administrator.

NOTE En cas de litige entre la spécification suivante et la spécification MVB définie en 4.6, cette dernière s'applique.

### 13.3.1.4.2 Call\_Write\_Mvb\_Administrator

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 13							
bus_id								reserved1							
checksum0															
configuration_version															
t_reply_max															
macro_cycles															
event_poll_strategy															
basic_period															
macrocycles_per_turn															
device_scan_strategy															
reserved2															
reserved3															
reserved4															
reserved5															
known_devices_list_offset															
reserved_list_offset															
periodic_list_offset															
bus_administrators_list_offset															
device_scan_list_offset															
end_list_offset															
known_devices_list [nr_devices]															
device_address (n)															
(offset_P0) cycle_lists_offsets ARRAY [11] OF															
cycle_n_offset (counted from offset_P0)															
split_lists_offsets ARRAY [5] OF															
split_m_n_offset (counted from offset_P0)															
cycle_lists [11]															
cycle_n ARRAY [cycle_size = cycle_n+1_offset – cycle_n_offset] OF															
f_code (n)				address (n)											
split_2_4 ARRAY [4] OF															
cycle_2								cycle_4							
split_8_16 ARRAY [32] OF															
cycle_8								cycle_16							
split_32_64 ARRAY [64] OF															
cycle_32								cycle_64							
split_128_256 ARRAY [256] OF															
cycle_128								cycle_256							
split_512_1024 ARRAY [1024] OF															
cycle_512								cycle_1024							
bus_administrators_list [nr_admins]															
Void (Vide)				device_address (n)											

```

UpperLowerCount ::= RECORD
    -- type auxiliaire pour les
    -- Split_Lists (listes de
    -- scission)
    {
        lower_count      UNSIGNED8
            -- nombre de trames du cycle le
            -- plus court à mettre dans
            -- chaque période de base
            -- (cette liste apparaît deux
            -- fois, sa taille étant la
            -- moitié de celle de la
            -- liste 'upper_count').
        upper_count      UNSIGNED8,
            -- nombre de trames du cycle le
            -- plus long à mettre dans
            -- chaque période de base
    }

Call_Write_Mvb_Administrator ::= RECORD
    {
        bus_id            UNSIGNED8 (0..15),
            -- indicatif de la liaison
        reserved1         WORD8
            -- réservé, =0
        checkword0        UNSIGNED16,
            -- somme de contrôle de cette
            -- structure (dépendant de
            -- l'utilisateur)
            -- checkword0 est situé au
            -- décalage 0.
        actualisation_key  UNSIGNED16,
            -- numéro de version de la
            -- configuration, sous forme
            -- d'un nombre à 16 bits. Le
            -- bit de poids fort mis à 1
            -- indique que les 15 bits de
            -- poids faible doivent être
            -- transmis dans la
            -- Mastership_Transfer_Response
            -- (Réponse de Transfert de
            -- Maîtrise).
            -- Les deux bits de poids
            -- faible sont copiés sur AX0
            -- et AX1.
        t_reply_max       UNSIGNED16,
            -- valeur maximale du délai de
            -- réponse prévisible pour tout
            -- dispositif sur le bus, en
            -- multiples de 1,0 µs;
            -- valeur par défaut (t_reply =
            -- 0) = 42,7 µs,
            -- valeur maximale = 256 µs.)
        macro_cycles      UNSIGNED16,
            -- durée d'un macrocycle après
            -- lequel le maître peut
            -- déclencher la commande si
            -- demandé, en un nombre de
            -- périodes de base.

        event_poll_strategy ENUM16
            {
                NOEVENTS      ('0000'H),
                    -- pas d'interrogation
                    -- d'événements
                HIPRIONLY     ('4000'H),
                    -- événements à priorité élevée
                    -- uniquement
                LOPRIONLY     ('8000'H),
                    -- événements à basse priorité
                    -- uniquement
                HILOPRIO      ('C000'H)
                    -- événements à priorité élevée
                    -- et basse.
            },
    }

```

```

basic period            UNSIGNED16,      -- période de base en
                                         microsecondes;
                                         valeurs admises: 0000 (pas
                                         d'interrogation), 1 000 µs.

macrocycles_per_turn    UNSIGNED16,      -- nombre de macrocycles au
                                         cours desquels le maître
                                         conserve le contrôle
                                         (multiple de 1 024 ms).

device_scan_strategy     ENUM16,
{
    SCANKNOWNONLY (0), --          scrutin de tous les dispositifs
                                   connus
    SCANALL (1)        --          scrutin de toutes les adresses
                                   de dispositif
},
reserved2                WORD16 (=0),
reserved3                WORD16 (=0),
reserved4                WORD16 (=0),
reserved5                WORD16 (=0),
known_device_list_offset  UNSIGNED16,      -- décalage (à partir du
                                         keyword0) où est située la
                                         première adresse de la liste
                                         de dispositifs;
                                         (La taille de la liste est
                                         la différence entre ce
                                         décalage et le
                                         reserved_list_offset
                                         (décalage de la liste
                                         réservée.))

reserved_list_offset      UNSIGNED16,      -- égal au « décalage de la
                                         liste périodique » (taille
                                         de la liste = 0)

periodic_list_offset      UNSIGNED16,      -- décalage (à partir du
                                         keyword0) où sont situés
                                         le premier F code +
                                         l'adresse du premier cycle;
                                         (la taille de la liste est
                                         la différence entre ce
                                         décalage et le décalage de
                                         la liste des administrateurs
                                         de bus.)

bus_administrators_list   UNSIGNED16,      -- décalage (à partir du
                                         keyword0) où est située la
                                         première adresse de la Liste
                                         des Administrateurs de Bus.

device_scan_list_offset   UNSIGNED16,      -- décalage (à partir du
                                         keyword0), où est située
                                         la liste d'attribution du
                                         balayage des dispositifs
                                         (non utilisé)

end_list_offset           UNSIGNED16,      -- décalage fictif qui indique
                                         la taille de la Scanlist et
                                         montre le prochain
                                         emplacement après le
                                         dernier.

known_devices_list        ARRAY [nr_devices] OF -- liste des dispositifs
                                         connus
{
    -- (dispositifs nr = décalage
    -- de la liste des dispositifs
    -- connus )
    reserved_list_offset
    (décalage de liste réservée)

```

```

device_address      WORD16      -- Adresse de dispositif
                                (Device_Address) du
                                dispositif
    },
cycle_lists_offsets RECORD      -- la liste périodique commence
                                ici
    {
        cycle_1_offset      UNSIGNED16      -- comprenant pour chacun des
                                -- 11 cycles possibles:
                                décalage
                                UNSIGNED16      -- (comptage à partir des
                                cycle_lists_indices)
        cycle_4_offset      UNSIGNED16      -- où la première adresse du
                                cycle
                                -- est située
        cycle_8_offset      UNSIGNED16
        cycle_16_offset     UNSIGNED16
        cycle_32_offset     UNSIGNED16
        cycle_64_offset     UNSIGNED16
        cycle_128_offset    UNSIGNED16
        cycle_256_offset    UNSIGNED16
        cycle_512_offset    UNSIGNED16
        cycle_1024_offset   UNSIGNED16
    }
split_list_offsets   RECORD      -- décalage (comptage à partir
                                des indices de listes de
                                cycles)
    {
                                -- où la première adresse du
                                tableau Split_n_m
                                -- est située
        split_2_4_offset    UNSIGNED16
        split_8_16_offset   UNSIGNED16
        split_32_64_offset  UNSIGNED16      --
        split_128_256_offset UNSIGNED16      --
        split_512_1024_offset UNSIGNED16      --
    }
cycle_lists          RECORD      -- onze listes, une pour chaque
                                cycle
                                -- taille de cycle = décalage
                                entre cette liste et la
                                liste suivante

```



```

cycle_1          ARRAY [cycle_size] OF FcodeAdr
cycle_2          ARRAY [cycle_size] OF FcodeAdr
cycle_4          ARRAY [cycle_size] OF FcodeAdr
cycle_8          ARRAY [cycle_size] OF FcodeAdr
cycle_16         ARRAY [cycle_size] OF FcodeAdr
cycle_32         ARRAY [cycle_size] OF FcodeAdr
cycle_64         ARRAY [cycle_size] OF FcodeAdr
cycle_128        ARRAY [cycle_size] OF FcodeAdr
cycle_256        ARRAY [cycle_size] OF FcodeAdr
cycle_512        ARRAY [cycle_size] OF FcodeAdr
cycle_1024       ARRAY [cycle_size] OF FcodeAdr
}
split_lists      RECORD
{
split_2_4        ARRAY [ 4] OF UpperLowerCount    -- voir
                                                         définition de ce Type.
split_8_16       ARRAY [16] OF UpperLowerCount
split_32_64      ARRAY [64] OF UpperLowerCount
split_128_256    ARRAY [256] OF UpperLowerCount
split_512_1024   ARRAY [1024] OF UpperLowerCount
}
bus_administrators_list  ARRAY [nr_admins] OF -- Liste des
                                                         Administrateurs de Bus
                                                         connus
{
                                                         -- avec:
                                                         (nr_admins = décalage de la
                                                         liste de balayage des
                                                         dispositifs - décalage de la
                                                         liste des administrateurs de
                                                         bus)
f_code           UNSIGNED4          -- réservé, = '0100'B
device_address   UNSIGNED12         -- adresse du dispositif formée
                                                         de 12 bits
}
}

```

**Tableau 15 – Exemple de mvb\_administrator\_list**

Décalage	Valeur	Nom	Contenu
000	D4EC	Known_Devices_List	checkword (non utilisé)
002	0001		actualisation_key (les deux bits de poids faible de Actualisation_Index)
004	07D0		T_reply_max = 2000,0 µs
006	0000		macro_cycle (non utilisé, soit 1024 ms)
008	C000		stratégie d'interrogation d'événements = événements de priorité haute et basse
00A	03E8		basic_period = 1 ms
00C	1000		durée d'un tour (4096 × 1 ms)
00E	0001		stratégie de balayage des dispositifs: 1= tous les dispositifs
010	0000		réservé
012	0000		réservé
014	0000		réservé
016	0000		réservé
018	0024		décalage par rapport à Known_Devices_List ('024'H = 36)
01A	0036		non utilisé (Known_Devices_List size = '036'H – '024'H = 9 mots)
01C	0036		décalage par rapport à Periodic_List (taille = 'B36'H – '036'H = 1664)
01E	0B36		décalage par rapport à Bus_Administrators_List
020	0040		décalage par rapport à Unused_List (vide)
022	0040		décalage par rapport à End_List
024	0020		dispositif '002'H (premier d'une liste de 9 dispositifs)
026	0022		dispositif '004'H bus 0
028	0024		dispositif '024'H bus 0
02A	0028		dispositif '028'H bus 0
02C	002C		dispositif '02C'H bus 0
02E	0034		dispositif '034'H bus 0
030	0036		dispositif '036'H bus 0
032	00C8		dispositif '0C8'H bus 0
034	00DE		dispositif '0DE'H (dernier dispositif)
036	0020	Periodic_List	Cycle_1_offset (emplacement = '36'H + '20'H = '56'H, taille = 0)
038	0020	Split_List_indices:	Cycle_2_offset (emplacement = '056'H, taille = 0)
03A	0020		Cycle_4_offset (emplacement = '056'H, taille = 0)
03C	0020		Cycle_8_offset (emplacement = '056'H, taille = 0)
03E	0020		Cycle_16_offset (emplacement = '056'H, taille = '024'H)
040	0044		Cycle_32_offset (emplacement = '7A'H, taille = '0A'H)
042	004E		Cycle_64_offset
044	0052		Cycle_128_offset
046	0056		Cycle_256_offset
048	0056		Cycle_512_offset
04A	0058		Cycle_1024_offset
04C	0058		Split_2_4_offset (emplacement = '36'H + '58'H = '8E'H)
04E	0060		Split_8_16_offset (emplacement = '36'H + '60'H = '96'H)
050	0080		Split_32_64_offset (emplacement = '36'H + '80'H = 'B6'H)
052	0100		Split_128_256_offset (emplacement = '36'H + '100'H = '136'H)
054	0300		Split_512_1024_offset

Tableau 15 (suite)

Décalage	Valeur	Nom	Contenu
056	00C8	Cycle_16	1 <sup>er</sup> F_code + adresse ('56'H = '36'H + '20'H)
058	00C9	(Cycle_1, Cycle_2,	2 <sup>ème</sup> F_code + adresse
05A	03E8	Cycle_4, Cycle_8,	3 <sup>ème</sup> F_code + adresse
05C	03EA	Cycle_1024 sont vides)	4 <sup>ème</sup> F_code + adresse
05E	0450		5 <sup>ème</sup> F_code + adresse
060	0452		6 <sup>ème</sup> F_code + adresse
062	0454		7 <sup>ème</sup> F_code + adresse
064	0456		8 <sup>ème</sup> F_code + adresse
066	0458		9 <sup>ème</sup> F_code + adresse
068	045A		10 <sup>ème</sup> F_code + adresse
06A	045C		11 <sup>ème</sup> F_code + adresse
06C	045E		12 <sup>ème</sup> F_code + adresse
06E	1464		13 <sup>ème</sup> F_code + adresse
070	1466		14 <sup>ème</sup> F_code + adresse
072	1468		15 <sup>ème</sup> F_code + adresse
074	146A		16 <sup>ème</sup> F_code + adresse
076	3460		17 <sup>ème</sup> F_code + adresse
078	3462		18 <sup>ème</sup> F_code + adresse (et dernier)
07A	015F	Cycle_32	1 <sup>er</sup> F_code + adresse ('56'H = '36'H + '44'H)
07C	01F1		2 <sup>ème</sup> F_code + adresse
07E	0321		3 <sup>ème</sup> F_code + adresse
080	415E		4 <sup>ème</sup> F_code + adresse
082	4320		5 <sup>ème</sup> F_code + adresse (et dernier)
084	019D	Cycle_64	1 <sup>er</sup> F_code + adresse ('84'H = '36'H + '4E'H)
086	11C2		2 <sup>ème</sup> F_code + adresse (et dernier)
088	40F9	Cycle_128	1 <sup>er</sup> F_code + adresse de Cycle_128
08A	418F		2 <sup>ème</sup> F_code + adresse de Cycle_128 (et dernier)
08C	20FE	Cycle_512	1 <sup>er</sup> F_code + adresse de Cycle_512 (et unique)
08E	0000	Split_2_4	non utilisé (cycles vides)
090	0000	(4 mots)	non utilisé (cycles vides)
092	0000		non utilisé (cycles vides)
094	0000		non utilisé (cycles vides)
096	0600	Split_8_16	1 <sup>ère</sup> période: 6 de Cycle_16, 0 de Cycle_8 (vide)
098	0600	(16 mots)	2 <sup>ème</sup> période: 6 de Cycle_16, 0 de Cycle_8 (vide)
09A	0400		3 <sup>ème</sup> période: 4 de Cycle_16, 0 de Cycle_8 (vide)
09C	0200		4 <sup>ème</sup> période: 6 de Cycle_16, 0 de Cycle_8 (vide)
...	...		....
0B4	0000		16 <sup>ème</sup> période: 0 de Cycle_16, 0 de Cycle_8 (vide)
0B6	0000	Split_32_64	1 <sup>ère</sup> période: 0
0B8	0000	(64 mots)	2 <sup>ème</sup> période: 0
0BA	0001		3 <sup>ème</sup> période: 1 de Cycle_32
0BC	0002		4 <sup>ème</sup> période: 2 de Cycle_32
0BE	0201		5 <sup>ème</sup> période: 2 de Cycle_64, 1 de Cycle_32
0C0	0001		6 <sup>ème</sup> période: 1 de Cycle_32
0C2	0000		7 <sup>ème</sup> période: 0 de Cycle_64, 1 de Cycle_32
...	...		...

**Tableau 15** (suite)

Décalage	Valeur	Nom	Contenu
134	0000	Split_128_256 (256 mots)	64 <sup>ème</sup> période: 0
136	0000		1 <sup>ère</sup> période: 0
...	...		2 <sup>ème</sup> période
...	...	Split_512_1024 (1024 mots)	...
334	0000		256 <sup>ème</sup> période:
336	0000		1 <sup>ère</sup> période
338	0000	Bus_Administrators_List	2 <sup>ème</sup> période
...	...		...
340	0001		6 <sup>ème</sup> période: 1 de Cycle_512
...	...	(Liste Non Utilisée, Fin)	...
B34	0000		1024 <sup>ème</sup> période
B36	0002		premier dispositif de la Bus_Administrators_List
B38 (B3A)	0004		second dispositif de la Bus_Administrators_List factice

#### 13.3.1.4.3 Reply\_Write\_Mvb\_Administrator

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
tnm_key								sif_code = 13							

```

Reply_Write_Mvb_Administrator ::= RECORD
    {}
-- aucun paramètre

```

## Bibliographie

CEI 60332-1-2, *Essai de propagation verticale de la flamme sur conducteur ou câble isolé – Procédure pour flamme à prémélange de 1 kW*

CEI 60870 (toutes les parties), *Matériels et systèmes de téléconduite*

CEI 60874-10-1, *Connecteurs pour fibres et câbles optiques – Partie 10-1: Spécification particulière pour connecteur à fibre optique de type BFOC/2,5 pour fibres multimodales type A1 (retirée)*

CEI 60874-10-2, *Connecteurs pour fibres et câbles optiques – Partie 10-2: Spécification particulière pour connecteur à fibre optique de type BFOC/2,5 pour fibres monomodales type B1 (retirée)*

CEI 60874-10-3, *Connecteurs pour fibres et câbles optiques – Partie 10-3: Spécification particulière pour connecteur à fibre optique de type BFOC/2,5 pour fibres monomodales et multimodales (retirée)*

CEI 61158-2, *Communications de données numériques pour le mesurage et le contrôle – Norme de bus de terrain pour utilisation dans les systèmes de contrôle industriels – Partie 2: Spécification de la couche physique et définition du service*

ISO/CEI 7498-1, *Technologies de l'information – Modèle de référence de base pour l'interconnexion de systèmes ouverts (OSI): Le modèle de base*

ISO/CEI 7498-4, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base – Partie 4: Cadre général de gestion*

ISO/CEI 8073, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) - Protocole assurant le service de transport en mode connexion*

ISO/CEI 8473 (toutes les parties), *Technologies de l'information – Protocole assurant le service réseau en mode sans connexion*

ISO/CEI 8802-3, *Technologies de l'information – Télécommunications et échange d'informations entre systèmes – Réseaux locaux et métropolitains – Exigences spécifiques – Partie 3: Accès multiples par supervision du signal et détection de collision (CSMA/CD) et spécifications pour la couche physique*

ISO/CEI 9072 (toutes les parties), *Systèmes de traitement de l'information – Communication de texte – Opérations à distance*

ISO/CEI 9506-1, *Systèmes d'automatisation industrielle – Spécification de messagerie industrielle – Partie 1: Définition du service*

ISO/CEI 9506-2, *Systèmes d'automatisation industrielle – Spécification de messagerie industrielle – Partie 2: Spécification de protocole*

ISO/CEI 10040:1998, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Aperçu général de la gestion-système*

ISO/CEI 10164 (toutes les parties), *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Gestion-systèmes*

ISO/CEI 10165 (toutes les parties), *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Services d'information de gestion – Structure des informations de gestion*

UIC 558, *Ligne de télécommande et d'information – Caractéristiques techniques unifiées pour l'équipement des voitures RIC*

UIC 568, *Sonorisation et téléphone des voitures RIC – Caractéristiques techniques pour l'équipement des voitures RIC unifiées*

UIC 648, *Accouplements pour les conduites électriques et pneumatiques sur les faces frontales des locomotives et des voitures pilotes*

---



INTERNATIONAL  
ELECTROTECHNICAL  
COMMISSION

3, rue de Varembé  
PO Box 131  
CH-1211 Geneva 20  
Switzerland

Tel: + 41 22 919 02 11  
Fax: + 41 22 919 03 00  
[info@iec.ch](mailto:info@iec.ch)  
[www.iec.ch](http://www.iec.ch)