IEC 62026-3

Edition 3.0 2014-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs) – Part 3: DeviceNet

Appareillage à basse tension – Interfaces appareil de commande-appareil (CDI) – Partie 3: DeviceNet

IEC 62026-3:2014-08(en-fr)

Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2014 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 l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC 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
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.

IEC Catalogue - webstore.iec.ch/catalogue
The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub
The advanced search enables 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
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - 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 14 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary
More than 55 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - 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.

A propos de l'IEC
La Commission Electrotechnique Internationale (IEC) 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 IEC
Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Catalogue IEC - webstore.iec.ch/catalogue
Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

Recherche de publications IEC - www.iec.ch/searchpub
La recherche avancée permet de trouver des publications IEC 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.

IEC Just Published - webstore.iec.ch/justpublished
Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org
Le premier dictionnaire en ligne 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 14 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary
Plus de 55 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

Service Clients - 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.

Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014

IEC 62026-3

Edition 3.0 2014-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs) – Part 3: DeviceNet

Appareillage à basse tension – Interfaces appareil de commande-appareil (CDI) – Partie 3: DeviceNet

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

PRICE CODE
CODE PRIX XF

ICS 29.130.20

ISBN 978-2-8322-1792-4

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.

® Registered trademark of the International Electrotechnical Commission
Marque déposée de la Commission Electrotechnique Internationale

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

CONTENTS

FOREWORD 8
INTRODUCTION 10
1 Scope 11
2 Normative references 11
3 Terms, definitions, symbols and abbreviated terms 12
3.1 Terms and definitions 12
3.2 Symbols and abbreviated terms 18
4 Classification 18
4.1 General 18
4.2 DeviceNet communication model 19
4.3 DeviceNet, CAN and CIP™ 20
5 Characteristics 21
5.1 DeviceNet connections 21
5.1.1 General 21
5.1.2 DeviceNet's use of the CAN identifier field 21
5.1.3 Connection establishment 22
5.2 DeviceNet messaging protocol 23
5.2.1 Explicit messaging 23
5.2.2 Input/output messaging 34
5.2.3 Fragmentation/reassembly 35
5.2.4 Offline connection set 39
5.2.5 Device heartbeat 49
5.2.6 Device shutdown message 50
5.2.7 Duplicate MAC ID detection protocol 52
5.2.8 Quick connect 53
5.3 DeviceNet communication object classes 53
5.3.1 General 53
5.3.2 Identity object class definition (class ID code: 0x01) 53
5.3.3 Message router object class definition (class ID code: 0x02) 54
5.3.4 DeviceNet object class definition (class ID code: 0x03) 54
5.3.5 Connection object class definition (class ID code: 0x05) 54
5.3.6 Acknowledge handler object class definition (class ID code: 0x2B) 55
5.4 Link access state machine 55
5.4.1 General 55
5.4.2 State transition diagram and event matrix 55
5.4.3 Duplicate MAC ID detection 58
5.5 Predefined master/slave connection set 58
5.5.1 General 58
5.5.2 Predefined master/slave connection set messages 59
5.5.3 DeviceNet object class specific services for the master/slave connection set 61
5.5.4 Slave connection object characteristics 68
5.5.5 Master connection object characteristics 73
5.5.6 Bit-strobe command/response messages 73
5.5.7 Poll command/response messages 74
5.5.8 Multicast poll command/response messages 75
5.5.9 Change of state/cyclic connections 75

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

5.5.10 Group 2 only devices 77
5.6 CIP Safety™ on DeviceNet 78
5.6.1 General 78
5.6.2 Use of CAN identifiers for CIP Safety on DeviceNet 78
5.7 Physical layer 78
5.7.1 General 78
5.7.2 Transceiver 80
5.7.3 Grounding 81
5.7.4 Isolation 81
5.7.5 Transmission medium 83
5.7.6 Topology 83
5.7.7 Link power 84
6 Product information 85
7 Normal service, mounting and transport conditions 85
7.1 Normal service conditions 85
7.1.1 General 85
7.1.2 Ambient air temperature 85
7.1.3 Altitude 86
7.1.4 Climatic conditions 86
7.2 Conditions during transport and storage 86
7.3 Mounting 86
8 Constructional and performance requirements 86
8.1 Indicators and configuration switches 86
8.2 DeviceNet cable 87
8.2.1 Overview 87
8.2.2 Cable profile template 87
8.2.3 Thick cable profile 88
8.2.4 Thin cable profile 92
8.2.5 Flat cable profile 95
8.3 Terminating resistors 98
8.4 Connectors 98
8.4.1 General specifications 98
8.4.2 Connector profile template 98
8.4.3 Open connector profile 99
8.4.4 Sealed mini connector profile 102
8.4.5 Sealed micro connector profile 103
8.4.6 Flat trunk connector profile 104
8.5 Device taps and power taps 107
8.5.1 Device taps 107
8.5.2 Power taps 108
8.6 Link powered devices 109
8.7 Miswiring protection 109
8.8 Power supplies 109
8.9 Electromagnetic compatibility (EMC) 110
8.9.1 General 110
8.9.2 Immunity 110
8.9.3 Emissions 111
8.10 Additional functional safety requirements related to EMC 112
9 Tests 112

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

9.1 General 112
9.2 Electrical and EMC testing 112
9.2.1 Test of the DeviceNet power supply 112
9.2.2 Device peak current consumption 113
9.2.3 Power ON behaviour 113
9.2.4 Reverse connection of V+ and V− 114
9.2.5 Disconnection of V− 115
9.2.6 Differential input impedance test 115
9.2.7 Transmit levels 115
9.2.8 Acknowledge delay 116
9.2.9 CDI tests 117
9.2.10 Electromagnetic compatibility testing 118
9.3 Logical testing 120
9.3.1 General 120
9.3.2 Duplicate MAC ID check test 120
9.3.3 UCMM 121
9.3.4 Allocation of predefined master/slave connection set – Explicit messaging connection 122
9.3.5 Allocation of predefined master/slave connection set – I/O messaging connection 122
9.3.6 Logical testing of safety products 123
Bibliography 124

Figure 1 – Typical DeviceNet controller-device interfaces 19
Figure 2 – DeviceNet protocol architecture compared with the OSI reference model 20
Figure 3 – DeviceNet's use of the CAN identifier field 21
Figure 4 – Explicit message CAN data field use 23
Figure 5 – Explicit message data field format 23
Figure 6 – Explicit message header format 24
Figure 7 – Service field format 24
Figure 8 – Open explicit messaging connection request format 26
Figure 9 – Open explicit messaging connection response format 28
Figure 10 – Close connection request format 30
Figure 11 – Close connection response format 30
Figure 12 – Non-fragmented explicit request message format, values 0 – 3 32
Figure 13 – Non-fragmented explicit request message format, value 4 33
Figure 14 – Non-fragmented success response message format 33
Figure 15 – Error response message 34
Figure 16 – Data field of an I/O message 34
Figure 17 – Format of DeviceNet fragmentation protocol 35
Figure 18 – I/O message fragment format 36
Figure 19 – Explicit message fragment format 36
Figure 20 – Acknowledgement message format 38
Figure 21 – Establishing the offline ownership 40
Figure 22 – Multicast nature of the offline ownership 41
Figure 23 – Offline ownership request message 44

AVA Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014

Figure 24 – Offline ownership response message protocol 42
Figure 25 – Communication faulted request message – Multicast protocol 43
Figure 26 – Communication faulted request message – Point-to-point protocol 44
Figure 27 – Identify communication faulted request message – Multicast protocol 45
Figure 28 – Communication faulted identify response message 46
Figure 29 – Identify communication faulted request message – Point-to-point protocol 46
Figure 30 – Who communication faulted request message 47
Figure 31 – Who response message 48
Figure 32 – Change MAC ID communication faulted request message 48
Figure 33 – Device heartbeat message 49
Figure 34 – Device shutdown message 51
Figure 35 – Duplicate MAC ID check CAN identifier field 52
Figure 36 – Duplicate MAC ID check message data field format 52
Figure 37 – Link access state transition diagram 56
Figure 38 – Allocation choice byte contents 61
Figure 39 – Allocate_master/slave_connection_set request message 62
Figure 40 – Success response to allocate_master/slave_connection_set request 63
Figure 41 – Parent explicit messaging connection logic 65
Figure 42 – Release choice byte contents 66
Figure 43 – Release_master/slave_connection set request message 66
Figure 44 – Success response to release_master/slave_connection_set request 67
Figure 45 – Predefined master/slave I/O connection state transition diagram 69
Figure 46 – Predefined master/slave explicit messaging connection state transition diagram 71
Figure 47 – Physical layer block diagram 79
Figure 48 – Device containing a non-isolated physical layer 82
Figure 49 – Device containing an isolated physical layer 83
Figure 50 – DeviceNet medium topology 84
Figure 51 – Thick cable: physical configuration 91
Figure 52 – Thick cable: current available on the DeviceNet power bus 91
Figure 53 – Thin cable: physical configuration 94
Figure 54 – Thin cable: current available on the DeviceNet power bus 94
Figure 55 – Flat cable: physical configuration 97
Figure 56 – Flat cable: current available on the DeviceNet power bus 97
Figure 57 – Open connector pinout 101
Figure 58 – Open connector geometry 101
Figure 59 – Mini connector pinout 103
Figure 60 – Micro connector pinout 104
Figure 61 – Flat trunk connector layout – Part 1 106
Figure 62 – Flat trunk connector layout – Part 2 107
Figure 63 – Power supply rise time test circuit 112
Figure 64 – Current consumption test circuit 113
Figure 65 – Power ON test circuit 114

AVA Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014

Figure 66 – Test circuit for reversal of V+ and V− and also V− interruption 114
Figure 67 – Differential impedance test circuit 115
Figure 68 – Transmit level test setup 116
Figure 69 – Transmit levels 116
Figure 70 – Timing test setup 117
Figure 71 – Timing 117
Figure 72 – CDI test configuration 118
Figure 73 – Test circuit for logical tests 120

Table 1 – Message body format values 27
Table 2 – Group select values 27
Table 3 – Source message ID in open explicit messaging connection request 28
Table 4 – Destination message ID in open explicit messaging connection response 29
Table 5 – UCMM error conditions/codes 31
Table 6 – Fragment type bit values 35
Table 7 – Ack status bit values 38
Table 8 – Offline connection set 40
Table 9 – Addresses reporting based upon mask 44
Table 10 – Device shutdown message shutdown code ranges 51
Table 11 – Device shutdown message “Open” shutdown codes 52
Table 12 – Link access state event matrix *(1 of 2)* 56
Table 13 – Predefined master/slave connection set identifier fields 60
Table 14 – Allocate_master/slave_connection_set request service data field parameters 61
Table 15 – Allocate_master/slave_connection set response parameters 62
Table 16 – Release_master/slave_connection_set request service data field parameters 66
Table 17 – DeviceNet object specific additional error codes 68
Table 18 – Connection instance IDs for predefined master/slave connections 69
Table 19 – Predefined master/slave I/O connection state event matrix *(1 of 2)* 70
Table 20 – Predefined master/slave explicit messaging connection state event matrix 72
Table 21 – Predefined master/slave I/O connection object attribute access 73
Table 22 – General physical layer characteristics 79
Table 23 – Transmitter characteristics 80
Table 24 – Receiver characteristics 81
Table 25 – Load limits 85
Table 26 – Cable profile: data pair specifications 87
Table 27 – Cable profile: DC power pair specifications 87
Table 28 – Cable profile: general specifications 88
Table 29 – Cable profile: topology 88
Table 30 – Thick cable: data pair specifications 89
Table 31 – Thick cable: DC power pair specifications 89
Table 32 – Thick cable: general specifications 90
Table 33 – Thick cable: topology 90

AVA Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014

Table 34 – Thick cable: maximum current available (A) based on network length 91
Table 35 – Thin cable: data pair specifications 92
Table 36 – Thin cable: DC power pair specifications 93
Table 37 – Thin cable: general specifications 93
Table 38 – Thin cable: topology 93
Table 39 – Thin cable: maximum current available (A) based on network length 94
Table 40 – Flat cable: data pair specifications 95
Table 41 – Flat cable: DC power pair specifications 96
Table 42 – Flat cable: general specifications 96
Table 43 – Flat cable: topology 96
Table 44 – Flat cable: maximum current available (A) based on network length 97
Table 45 – Connector profile template 99
Table 46 – Open connector 100
Table 47 – Sealed mini connector 102
Table 48 – Sealed micro connector *(1 of 2)* 103
Table 49 – Flat trunk connector 105
Table 50 – Internal pass-through conductor specifications 108
Table 51 – Internal drop conductor specifications 108
Table 52 – Internal pass-through conductor specifications 108
Table 53 – Internal power drop conductor specifications 109
Table 54 – Voltage regulator specifications 109
Table 55 – DeviceNet power supply specifications 110
Table 56 – Immunity performance criteria 111

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

LOW-VOLTAGE SWITCHGEAR AND CONTROLGEAR – CONTROLLER-DEVICE INTERFACES (CDIs) –

Part 3: DeviceNet

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 62026-3 has been prepared by subcommittee 121A: Low-voltage switchgear and controlgear, of IEC technical committee 121: Switchgear and controlgear and their assemblies for low voltage.

This third edition of IEC 62026-3 cancels and replaces the second edition published in 2008. This third edition constitutes a technical revision.

The main changes with respect to the previous edition are the followings:

– specification of group 4 messages (5.1.2);
– clarifications on messaging protocol (5.2);
– addition of I/O multicast poll messages (5.5.2 and 5.5.8);
– clarifications on slave behaviour (5.5.4 and 5.5.9);
– clarifications on physical layer (transceiver) in 5.7;

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

– miscellaneous corrections/clarifications on cable (8.2);
– clarifications on EMC testing (9.2.10) and logical testing (9.3).

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

FDIS	Report on voting
17B/1814/FDIS	121A/18/RVC

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 the IEC 62026, under the general title *Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs)*, 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.

The contents of the corrigenda of March 2015 and October 2019 have been included in this copy.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

INTRODUCTION

DeviceNet™[1] is intended for use in, but is not limited to, industrial automation applications. These applications may include devices such as limit switches, proximity sensors, electro-pneumatic valves, relays, motor starters, operator interface panels, analogue inputs, analogue outputs and controllers.

[1] DeviceNet™ is a trade name of ODVA, Inc. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the trademark holder or any of its products. Compliance to this standard does not require use of the trade name DeviceNet™. Use of the trade name DeviceNet™ requires permission of ODVA, Inc.

AVA Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014

LOW-VOLTAGE SWITCHGEAR AND CONTROLGEAR – CONTROLLER-DEVICE INTERFACES (CDIs) –

Part 3: DeviceNet

1 Scope

This part of IEC 62026 specifies an interface system between single or multiple controllers, and control circuit devices or switching elements. The interface system uses two conductor pairs within one cable – one of these pairs provides a differential communication medium and the other pair provides power to the devices. This part establishes requirements for the interoperability of components with such interfaces.

This part of IEC 62026 specifies the following particular requirements for DeviceNet:

– requirements for interfaces between controllers and switching elements;
– normal service conditions for devices;
– constructional and performance requirements;
– tests to verify conformance to requirements.

These particular requirements apply in addition to the general requirements of IEC 62026-1.

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 60529:1989, *Degrees of protection provided by enclosures (IP Code)*
IEC 60529:1989/AMD 1:1999
IEC 60529:1989/AMD 2:2013

IEC 60947-5-2:2007, *Low-voltage switchgear and controlgear – Part 5-2: Control circuit devices and switching elements – Proximity switches*
IEC 60947-5-2:2007/AMD 1:2012

IEC 61000-4-2:2008, *Electromagnetic compatibility (EMC) – Part 4-2: Testing and measurement techniques – Electrostatic discharge immunity test*

IEC 61000-4-3:2006, *Electromagnetic compatibility (EMC) – Part 4-3: Testing and measurement techniques – Radiated, radio-frequency, electromagnetic field immunity test*
IEC 61000-4-3:2006/AMD 1:2007
IEC 61000-4-3:2006/AMD 2:2010

IEC 61000-4-4:2012, *Electromagnetic compatibility (EMC) – Part 4-4: Testing and measurement techniques – Electrical fast transient/burst immunity test*

IEC 61000-4-5:2005, *Electromagnetic compatibility (EMC) – Part 4-5: Testing and measurement techniques – Surge immunity test*

AVA Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

IEC 61000-4-6:2013, *Electromagnetic compatibility (EMC) – Part 4-6: Testing and measurement techniques – Immunity to conducted disturbances, induced by radio-frequency fields*

IEC 61158-4-2:2014, *Industrial communication networks – Fieldbus specifications – Part 4-2: Data-link layer protocol specification – Type 2 elements*

IEC 61158-5-2:2014, *Industrial communication networks – Fieldbus specifications – Part 5-2: Application layer service definition – Type 2 elements*

IEC 61158-6-2:2014, *Industrial communication networks – Fieldbus specifications – Part 6-2: Application layer protocol specification – Type 2 elements*

IEC 61784-3-2, *Industrial communication networks – Profiles – Part 3-2: Functional safety fieldbuses – Additional specifications for CPF 2*

IEC 62026-1:2007, *Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs) – Part 1: General rules*

CISPR 11:2009, *Industrial, scientific and medical equipment – Radio-frequency disturbance characteristics – Limits and methods of measurement*
CISPR 11:2009/AMD 1:2010

ISO 11898-1:2003, *Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signalling*

ISO 11898-2:2003, *Road vehicles – Controller area network (CAN) – Part 2: High-speed medium access unit*

ANSI B93.55M-1981 (R1988), *Hydraulic Fluid Power – Solenoid Piloted Industrial Valves – Interface Dimensions for Electrical Connectors*

ASTM D 4566-94[2], *Standard Test Methods for Electrical Performance Properties of Insulations and Jackets for Telecommunications Wire and Cable*

3 Terms, definitions, symbols and abbreviated terms

For the purposes of this document, the terms, definitions, symbols and abbreviations given in IEC 62026-1 as well as the following apply.

3.1 Terms and definitions

3.1.1
acknowledged fragmentation
fragmentation performed on an explicit message, in which the transmission of a fragment from the transmitting object is followed by the transmission of an acknowledgement by the receiving object

Note 1 to entry: The reception of each fragment is acknowledged by the receiving object.

[2] A newer version of this document exists (ASTM D4566-08e1), however the listed revision applies for this standard

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

3.1.2
ack status
field within an acknowledgement/response message format that indicates whether or not an error has been encountered by the receiver of a fragmented message

Note 1 to entry: This applies specifically to the DeviceNet fragmentation protocol.

3.1.3
application objects
set of object classes and their object instances that are available within the node

Note 1 to entry: These objects manage and provide the exchange of data and messages across DeviceNet controller-device interfaces (CDIs) and within the DeviceNet compliant node.

3.1.4
attribute
description of an externally accessible characteristic or feature of an object

Note 1 to entry: Attributes typically provide status information or govern the operation of an object.

3.1.5
bit-strobe
communication using strobing

3.1.6
broadcast
communication from one node to all other nodes

3.1.7
BOI attribute
bus-off interrupt attribute
attribute of the DeviceNet object that defines the behaviour of a device after encountering a bus-off event in the CAN chip

Note 1 to entry: See IEC 61158-4-2:2014, 7.7.4.4 for further details.

3.1.8
CAN
ISO specification that defines a generic physical layer and data link medium access procedure based on non-destructive bit-wise arbitration

Note 1 to entry: See ISO 11898-1 and ISO 11898-2.

Note 2 to entry: CAN is the abbreviation of “controller area network”.

3.1.9
CAN_H
positive half of the differential physical CAN signal

3.1.10
CAN_L
negative half of the differential physical CAN signal

3.1.11
client
(1) object which uses the services of another (server) object to perform a task;

(2) initiator of a message to which a server reacts

Note 1 to entry: See server (3.1.44).

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

[SOURCE: IEC 61158-4-2:2014, 3.4.9, MODIFIED]

3.1.12
common service
CIP service used by DeviceNet objects

Note 1 to entry: See IEC 61158-5-2:2014, 6.2.1.3 and IEC 61158-6-2:2014, 4.1.8.

3.1.13
communication objects
objects that manage and provide run-time exchange of messages across DeviceNet

3.1.14
connection
logical binding between two or more application objects

Note 1 to entry: These application objects can be located at the same node or at different nodes.

3.1.15
CID
connection ID
connection identifier assigned to all transmissions that are associated with a particular connection between multiple nodes

3.1.16
connection object
manages the communication-specific aspects associated with connections between nodes

3.1.17
consumer
end point of a connection that is responsible for receiving data

3.1.18
destination MAC ID
MAC ID of a node that is to receive a message

3.1.19
device tap
physical point of attachment from a DeviceNet device to a trunk cable or a drop cable

3.1.20
device type
identification of a collection of device-dependent information describing a viable combination of options selected for all layers in the communication stack

3.1.21
dominant
one of two complementary logic levels on the physical signal

Note 1 to entry: The dominant level is a logical "0".

3.1.22
duplicate MAC ID detection
DeviceNet-defined protocol that ensures no two nodes on the same link are assigned the same MAC ID

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

3.1.23
explicit messaging
each explicit message commands the performance of a particular task and the return of the results of the task performance to the requester

3.1.24
fragmentation
DeviceNet protocol provided by the connection object that defines a method by which data greater than eight (8) bytes in length may be transmitted

3.1.25
group 2 client
unconnected message manager (UCMM) capable device that has gained ownership of the predefined master/slave connection set within a server such that it may act as the client on those connections

3.1.26
group 2 only client
device that is acting as a group 2 client to a group 2 only server

Note 1 to entry: The group 2 only client provides the UCMM functionality for group 2 only servers that it has allocated.

3.1.27
group 2 server
unconnected message manager (UCMM) capable device that has been configured to act as the server for the predefined master/slave identifier connections

3.1.28
group 2 only server
slave device that is UCMM incapable and uses the predefined master/slave connection set to establish communications

Note 1 to entry: A group 2 only device can transmit and receive only those identifiers defined by the predefined master/slave connection set.

3.1.29
I/O connection
connection between a producer and one or more consumers for the purpose of exchanging application-specific, time-critical I/O data

3.1.30
I/O data
information which is transferred between I/O points and the controllers which use and set the values

3.1.31
I/O messaging
exchange of data in a previously defined format

3.1.32
isolated device
device in which some of its components are not referenced to the V- of the physical layer

Note 1 to entry: See non-isolated device (3.1.37).

3.1.33
master
node which gathers and distributes I/O using the predefined master/slave connection set

AVA Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

3.1.34
MAC ID
link address of a DeviceNet node

Note 1 to entry: MAC ID is the abbreviation of “medium access control ID”.

3.1.35
multicast connection
logical connection from one object to multiple other objects

Note 1 to entry: A multicast connection allows data to be transferred in a single transaction from a producer to multiple consumers sharing the same connection.

3.1.36
node
DeviceNet entity which is identified at the data link level by a unique MAC ID

Note 1 to entry: Multiple DeviceNet nodes can be implemented in one device but they appear as logically distinct nodes on the DeviceNet link.

3.1.37
non-isolated device
device in which all components are referenced to the V- of the physical layer

Note 1 to entry: See isolated device (3.1.32).

3.1.38
object
(1) abstract representation of a device’s capabilities;

Note 1 to entry: Objects can be composed of any or all of the following components:

a) Data (information which changes with time);

b) Configuration (parameters for behaviour);

c) Procedures (actions that can be performed using data and configuration)

(2) collection of related data (in the form of variables) and procedures for operating on that data.

3.1.39
point-to-point connection
connection that exists between two objects only

Note 1 to entry: Explicit messaging connections are always point-to-point. I/O connections can be either point-to-point or multicast, see multicast connection (3.1.35).

3.1.40
predefined master/slave connection set
utilization of explicit messaging and I/O messaging connections to create and configure predefined connection objects within each connection end-point

Note 1 to entry: Uses the general rules as a basis for the definition of a set of connections which facilitate communications typically seen in a master/slave relationship.

3.1.41
producer
end point of a connection that is responsible for sending data

3.1.42
recessive
one of two complementary logic levels on the physical signal

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Note 1 to entry: The recessive level is a logical "1".

3.1.43
serial number
unique 32-bit integer assigned by each manufacturer to every DeviceNet device

Note 1 to entry: The number is stored within the device as an attribute of the identity object and is unique with respect to the manufacturer.

3.1.44
server
object which provides services to another (client) object

Note 1 to entry: See client (3.1.11).

[SOURCE: IEC 61158-4-2:2014, 3.4.62, MODIFIED]

3.1.45
service
operation or function that an object performs upon request from another object

3.1.46
slave
node that receives data from and returns data to its master using the predefined master/slave connection set and a communication method set by the master

3.1.47
source MAC ID
MAC ID of a node that is transmitting a message

3.1.48
trigger
service used by an application to initiate the production of data

3.1.49
UCMM capable device
device that supports the UCMM

Note 1 to entry: See 3.1.52.

3.1.50
UCMM incapable device
device that does not support the UCMM

Note 1 to entry: See 3.1.52.

3.1.51
unconnected explicit message
explicit message between nodes that have not yet established a connection between each other

3.1.52
UCMM
unconnected message manager
function within a node that receives and processes unconnected explicit messages

3.1.53
USINT
8-bit integer

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Note 1 to entry: USINT is the abbreviation of “unsigned short integer”.

3.1.54
UINT
16-bit integer

Note 1 to entry: UINT is the abbreviation of “unsigned integer”.

3.2 Symbols and abbreviated terms

BOI	bus-off interrupt
CAN	controller area network
CDI	controller-device interface, see IEC 62026-1
CID	connection identifier
CIP™	Common Industrial Protocol
CRC	cyclic redundancy check
DCR	DC resistance
EUT	equipment under test
LED	Light-emitting diode
MAC ID	medium access control identifier
UCMM	unconnected message manager
UINT	unsigned integer
USINT	unsigned short integer

4 Classification

4.1 General

DeviceNet interfaces controlling devices to control circuit devices or switching elements. DeviceNet uses two conductor pairs within one cable – one of these pairs provides a differential communication medium, and the other pair provides power to the devices. The maximum current supported is 8 A at 24 V d.c. Data is transmitted at bit rates of 125 kbit/s, 250 kbit/s or 500 kbit/s with maximum cable lengths of 500 m, 250 m and 100 m respectively. A maximum of 8 bytes of data may be transmitted without fragmentation. A maximum of 64 nodes may be connected using a linear topology with a trunk line and drop lines (see Figure 1). DeviceNet supports the transmission of I/O data, diagnostics, messaging and programming/configuration. Data exchange may be event driven (change of state), cyclic, polled, strobed, or multicast polled.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

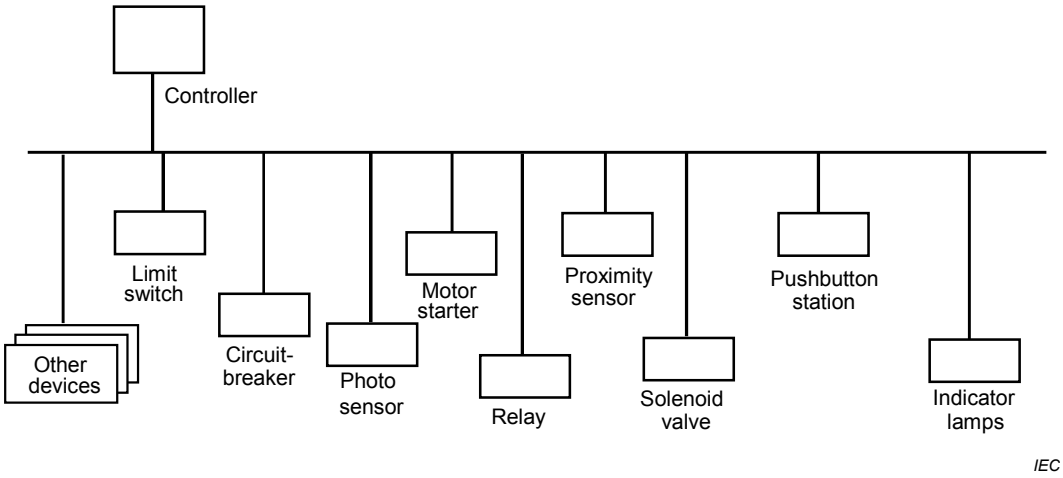


Figure 1 – Typical DeviceNet controller-device interfaces

This part of IEC 62026 defines a connection based scheme to facilitate all application communications. A DeviceNet connection provides a communication path between multiple end-points. The end-points of a connection are applications that need to share data. Transmissions associated with a particular connection are assigned an identification value when a connection is established. This identification value is called the connection ID (CID).

Connection objects model the communication characteristics of a particular application-to-application(s) relationship.

DeviceNet's connection based scheme defines a dynamic means by which the following two types of connections may be established:

– **I/O connections**: Provide dedicated, special purpose communication paths between a producing application and one or more consuming applications. Application specific I/O data moves through these paths.

  I/O messages are exchanged across I/O connections. An I/O message consists of a CID and associated I/O data. The connection end-points shall have knowledge of the intended use or meaning of the I/O message.

  This part of IEC 62026 does not define any particular use for I/O messaging. There are a wide variety of functions that may be accomplished using I/O messaging. The meaning and/or intended use of all I/O messages shall be made known to the system either by the particular type of product transmitting an I/O message, or based upon configuration performed using explicit messaging.

– **Explicit messaging connections:** Provide generic, multi-purpose communication paths between two devices. Explicit messages provide the typical request/response oriented communications.

  Explicit messages are exchanged across explicit messaging connections. Explicit messages are used to command the performance of a particular task and to report the results of performing the task.

  DeviceNet defines an explicit messaging protocol that states the meaning of the message. An explicit message consists of a CID and associated messaging protocol information.

The rules that govern the dynamic establishment of these connections are used as a foundation upon which a predefined set of connections is defined.

4.2 DeviceNet communication model

The abstract object oriented communication model of a DeviceNet node includes the following:

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

- **unconnected message manager (UCMM):** processes DeviceNet unconnected explicit messages;
- **identity object:** identifies and provides general information about the device;
- **connection class:** allocates and manages internal resources associated with both I/O and explicit messaging connections;
- **connection object:** manages the communication specific aspects associated with a particular application-to-application relationship;
- **DeviceNet object:** provides the configuration and status of a physical DeviceNet CDI;
- **message router:** forwards an explicit request message to the appropriate object;
- **application objects:** implement the intended purpose of the product.

4.3 DeviceNet, CAN and CIP™

DeviceNet lower layers are based on ISO 11898-1 and ISO 11898-2 and use the Controller Area Network (CAN) technology. DeviceNet upper layers use a subset of the Common Industrial Protocol (CIP™ [3]) protocol and services specified in the IEC 61158 series.

The relationships between DeviceNet, CAN, CIP and the OSI reference model (ISO/IEC 7498-1:1994) are shown in Figure 2.

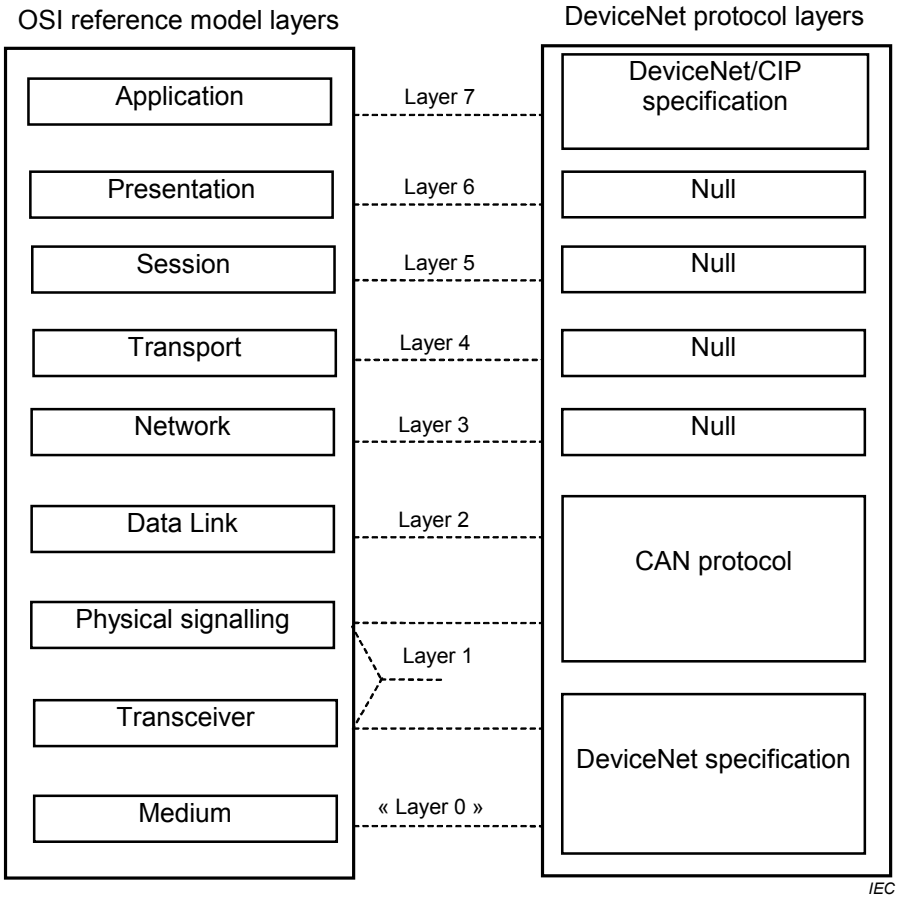


Figure 2 – DeviceNet protocol architecture compared with the OSI reference model

3 CIP™ is a trade name of ODVA, Inc. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by IEC of the trademark holder or any of its products. Compliance to this profile does not require use of the trade name CIP™. Use of the trade name CIP™ requires permission of ODVA, Inc.

AVA Documents Document Sharing Hub
Full document details: IEC 62026-3 2014

5 Characteristics

5.1 DeviceNet connections

5.1.1 General

DeviceNet is a connection based controller-device interface. DeviceNet connections are used to provide logical paths between multiple applications. When a connection is established, the transmissions associated with that connection are assigned a connection ID (CID). If the connection involves a bi-directional exchange, then two CID values shall be assigned to the connection.

DeviceNet uses the CAN identifier field and defines the steps involved in the dynamic establishment of I/O and explicit messaging connections.

5.1.2 DeviceNet's use of the CAN identifier field

DeviceNet uses the 11 bit CAN Identifier field ("Standard Frame Format") as described in ISO 11898-1.

The overall CAN identifiers available within DeviceNet are subdivided into four separate message groups: group 1, group 2, group 3 and group 4.

For connection based messages, the CID is placed within the CAN identifier field. Figure 3 also describes the components of a DeviceNet CID.

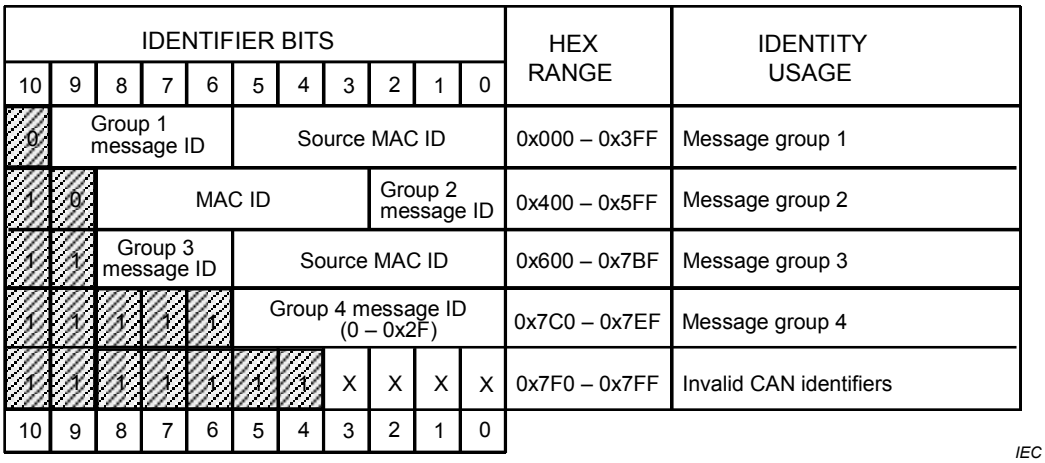


Figure 3 – DeviceNet's use of the CAN identifier field

As shown in Figure 3 the CAN identifier field on DeviceNet contains the following:

- **message ID:** Identifies a message within a message group inside a particular node. When a connection is established, the nodes utilise a message ID in combination with a MAC ID to generate a CID. The resulting CID is specified in the CAN identifier field associated with related transmissions;
- **source MAC ID:** Groups 1 and 3 require the specification of a source MAC ID within the CAN identifier field;
- **MAC ID:** Message group 2 allows the specification of either source or destination within the MAC ID portion of the CAN identifier field.

Both explicit messaging and I/O connections may be established in message groups 1, 2 and 3.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Group 2 message ID values 6 and 7 are used as follows:

– group 2 message ID 6 is used for the configuration of the communications utilised in a master/slave application (see 5.5);

– group 2 message ID 7 is used in the detection of nodes that have been assigned identical MAC IDs (see 5.3.6).

Group 3 message ID values 5, 6 and 7 are used as follows:

– group 3 message ID 5 is used when sending responses associated with unconnected explicit messaging requests, as well as device heartbeat and device shutdown messages;

– group 3 message ID 6 is used when sending unconnected explicit messaging requests;

– group 3 message ID 7 is invalid and shall not be used.

Group 4 message ID values are used as follows:

– group 4 message IDs 0x00 through 0x2B are reserved for future use and shall not be used;

– group 4 message ID 0x2C is used for communication faulted response messages;

– group 4 message ID 0x2D is used for communication faulted request messages;

– group 4 message ID 0x2E is used for offline ownership response messages;

– group 4 message ID 0x2F is used for offline ownership request messages.

5.1.3 Connection establishment

5.1.3.1 Explicit messaging connections and the UCMM

Message group 3 defines the identifier codings to support unconnected explicit messaging. Unconnected explicit messages establish and manage explicit messaging connections. Unconnected request messages are specified by transmitting a group 3 message whose message ID component is set to 6. The only valid services that can be transmitted as unconnected explicit request messages are:

– open explicit messaging connection request;

– close connection request.

The messages listed above are never transmitted as connection-based messages (see connection-based explicit messaging in 5.2.1.6).

Responses to unconnected explicit requests are transmitted as unconnected response messages. Unconnected response messages are specified by transmitting a group 3 message whose message ID component is set to 5. The only valid services that can be transmitted as unconnected explicit response messages are:

– open explicit messaging connection response;

– close connection response;

– error response;

– device heartbeat message;

– device shutdown message.

5.1.3.2 I/O connections

I/O connections are dynamically established by interfacing with the connection object class across a previously established explicit messaging connection. A connection object instance shall be created and configured at each node.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

5.2 DeviceNet messaging protocol

5.2.1 Explicit messaging

5.2.1.1 General

This subclause describes the explicit messaging protocol and presents details associated with the dynamic establishment of explicit messaging connections (see Figure 4).

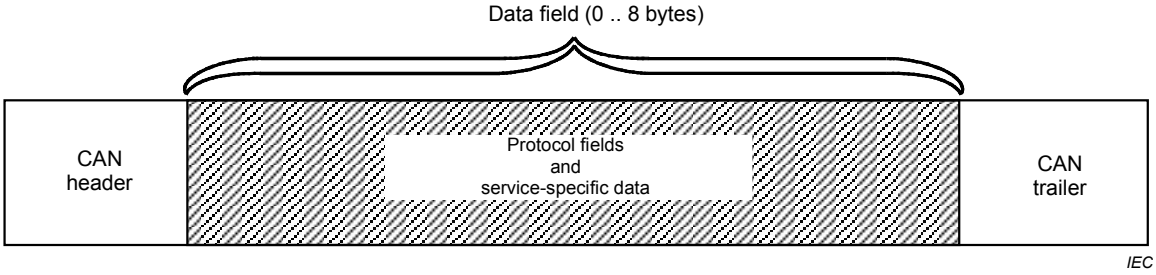


Figure 4 – Explicit message CAN data field use

Figure 5 provides the format of the CAN data field associated with explicit messages:

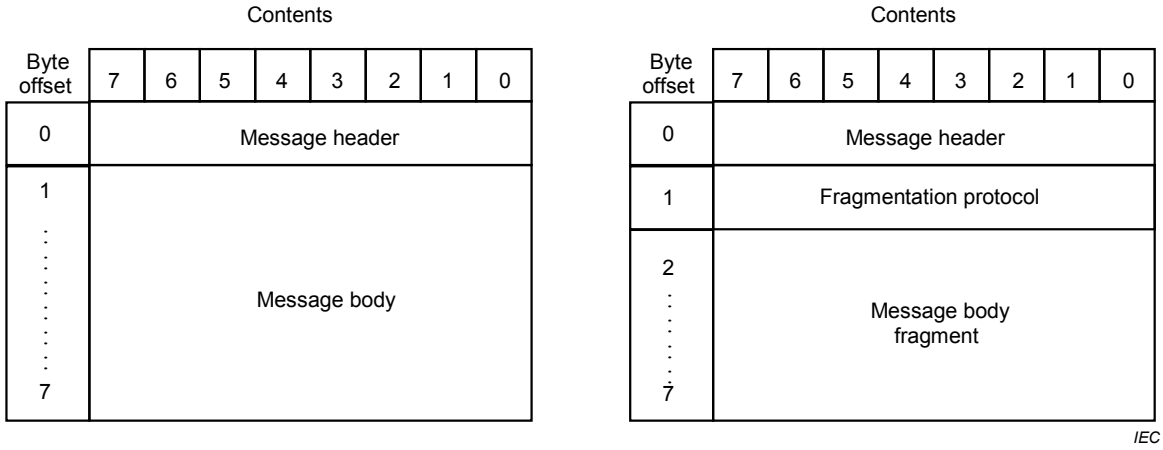


Figure 5 – Explicit message data field format

The data field of a transmission that contains the complete explicit message includes:

– a message header;
– the entire message body.

If the explicit message is greater than eight bytes in length, it shall be transmitted in a fragmented manner. The fragmentation/re-assembly function is provided by the connection object. A piece of a fragmented explicit message includes:

– a message header;
– the fragmentation protocol (see 5.2.3.2);
– a message body fragment.

5.2.1.2 Message header

The message header is located within byte offset zero in the CAN data field of an explicit message and shall be formatted as shown in Figure 6:

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

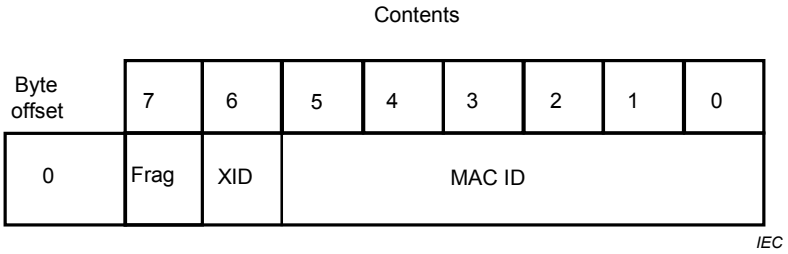


Figure 6 – Explicit message header format

Message header contents:

– **frag** (fragment bit): Indicates whether or not this transmission is a piece of a fragmented explicit message.

0 = non-fragmented

1 = fragmented

– **XID (transaction ID):** This field is utilised by an application to match a response with the associated request. This field is simply echoed by the server in a response message. The server module does not utilise this field to perform any type of duplicate message detection logic;

– **MAC ID:** Contains either the source or destination MAC ID.

When an explicit message is received, the MAC ID field within the message header is examined. If the destination MAC ID is specified in the CID, then the source MAC ID of the other end-point shall be specified in the message header. If the source MAC ID is specified in the CID, then the receiving module's MAC ID shall be specified in the message header. If neither of these conditions are true, then the message shall be discarded.

5.2.1.3 Message body

The message body contains a service field and service specific arguments.

The first argument specified within the message body is the service field which identifies the particular request or response being delivered. Figure 7 illustrates the format of the service field.

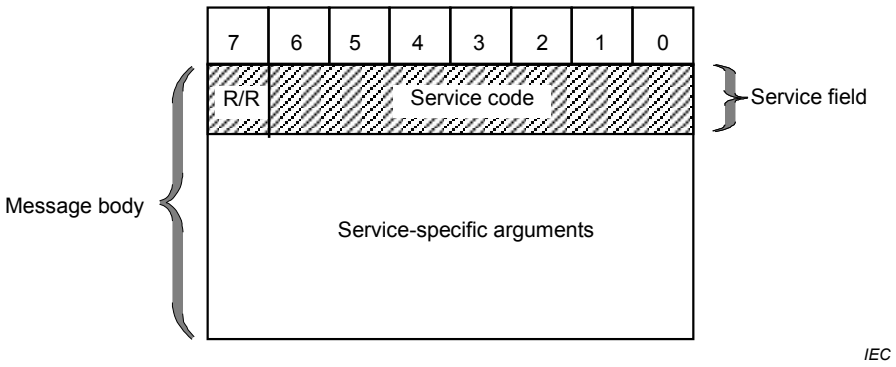


Figure 7 – Service field format

Service field contents:

– **service code:** the value specified within the least significant 7 bits of the service field byte that indicates the type of service being transmitted;

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

– **R/R:** the most significant bit in the service field. Its value determines whether a message is a request or a response.

  0 = request

  1 = response

DeviceNet defines a set of common services. DeviceNet common services are the open set whose parameters and required behaviours are defined in IEC 61158-5-2:2014, 6.2.1.3 and IEC 61158-6-2:2014, 4.1.8.

Information following the service field is specific to the particular type of service being transmitted.

5.2.1.4 Fragmentation protocol

If the transmission is a piece of a fragmented explicit message, then the data field shall contain the message header, the fragmentation protocol and a message body fragment. The fragmentation protocol facilitates the fragmentation and reassembly of explicit messages which have a message body greater than 8 bytes (see 5.2.3).

5.2.1.5 UCMM services

5.2.1.5.1 General

The unconnected message manager (UCMM) provides for the dynamic establishment of explicit messaging connections. This subclause presents a detailed description of the service specific arguments associated with the open explicit messaging connection and close connection services provided by the UCMM. See 9.3.3 for test specifications regarding establishment of explicit messaging connections.

The UCMM processes two services which manage the allocation and deallocation of explicit messaging connections:

– **open explicit messaging connection:** service code = 0x4B. Used to establish an explicit messaging connection;

– **close connection:** service code = 0x4C. Used to delete a connection object and deallocate all associated resources.

These services are accessed using the unconnected explicit request and response CAN identifier fields within message group 3 as specified in 5.1.2. When processing an unconnected explicit request the UCMM may need to return an error indication to the requestor and, as such, the error response explicit message may be transmitted with the unconnected explicit response CAN identifier.

5.2.1.5.2 Open explicit messaging connection request

This service requests the establishment of a logical connection between two nodes across which explicit messages will be transmitted. This service is transmitted as an unconnected request message (message group 3, message ID 6) (see Figure 8).

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

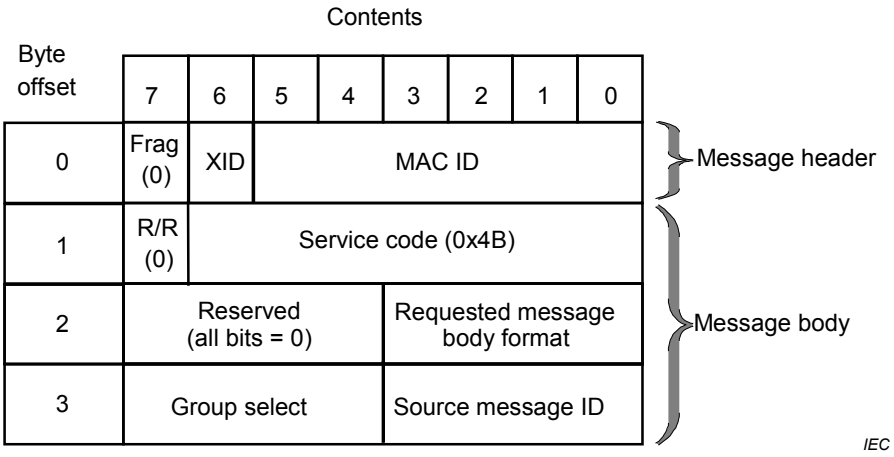


Figure 8 – Open explicit messaging connection request format

Open explicit messaging connection request contents:

– **frag (0)/transaction ID/MAC ID**: See 5.2.1.2. The destination MAC ID is always specified in the message header associated with an open explicit messaging connection request/response;

– **R/R bit (0):** Indicates this is a request message;

– **service code (0x4B):** Identifies this message as an open explicit messaging connection service;

– **reserved bits:** These bits shall be ignored by the receiver and shall be set to zero by the transmitter;

– **requested message body format:** The field used by the client to request a particular message body format for subsequent explicit messages transmitted over this connection. A server shall support at least one of the DeviceNet message body formats (values 0 – 3).

The server node responding to this open explicit messaging request defines the actual message body format to be used over this connection. See Table 1 for message body format values. Servers shall do one of the following:

- refuse the request and return a supported format value in the open explicit messaging connection response; the returned format shall be one of the DeviceNet message body formats (values 0 – 3);
- accept this request by echoing the same format value in the open explicit messaging connection response;

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Table 1 – Message body format values

Value	Meaning
0	DeviceNet (8/8). Class ID = 8 bit integer, Instance ID = 8 bit integer
1	DeviceNet (8/16). Class ID = 8 bit integer, Instance ID = 16 bit integer
2	DeviceNet (16/16). Class ID = 16 bit integer, Instance ID = 16 bit integer
3	DeviceNet (16/8). Class ID = 16 bit integer, Instance ID = 8 bit integer
4	CIP path. The addressing size is variable and is provided as a packed EPATH on each request [a]
5 – 0xF	Reserved
NOTE Messages transmitted across this connection are formatted as described in 5.2.1.6.	
[a] The CIP Path is specified as an 8-bit path length (USINT) followed by the path (packed EPATH). See IEC 61158-6-2:2014, 4.1.9 for information on EPATH. The server shall support, as a minimum, the most efficient encoding for any logical segment within the EPATH (for example 8-bit class encoding for class code 0x01).	

– **group select:** The field that indicates the message group across which messages associated with this connection are to be exchanged. The following values are shown in Table 2:

Table 2 – Group select values

Value	Meaning
0	Message group 1
1	Message group 2[a]
2	Reserved
3	Message group 3
4 – 0xE	Reserved
0xF	Reserved for node ping[b]
[a] The message group 2 identifier allows for specification of either the source or destination MAC ID. For explicit messaging connections established across message group 2, the client places the MAC ID of the server in the connection ID when transmitting messages across this connection. The server places its own MAC ID in the connection ID when transmitting messages across this connection. This process requires the server to allocate two separate message IDs from its group 2 pool. [b] This group select value can be used to force a UCMM capable target node to respond without creating any resources within the node. When a request with this value is received, the target shall respond with a general error code of "invalid parameter" (0x20) and an additional error code of 0x01.	

The client selects the message group across which the transmissions associated with this explicit messaging connection will take place. If the server cannot satisfy the request, then it shall reject the request and return an error response.

– **source message ID:** The use of this field depends on the value within the group select field (see Table 3):

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Table 3 – Source message ID in open explicit messaging connection request

If group select equals:	Then the source message ID:
0 or 3	Specifies the message ID the client has allocated from its group 1 or 3 message ID pool. The client shall use this message ID in combination with its own MAC ID to generate the connection ID specified when it transmits a message across this connection[a]
1	Is ignored/set to the value zero (0)[b]

[a] The client places this value within the message ID component of the message group 1 or 3 Identifier.

[b] Explicit messaging connections established across message group 2 require the server to allocate two group 2 message IDs and return them in the open explicit messaging connection response message. The client shall use one of these message IDs to generate the connection ID it specifies when transmitting a message across this connection. The other shall be used by the server to generate the connection ID it specifies when transmitting a message across this connection.

The UCMM within the server validates the open explicit messaging connection request arguments. If they are valid, the UCMM invokes the create service of the connection class to obtain a connection object instance (see IEC 61158-5-2:2014, 6.2.3). The resulting connection object is automatically configured to be an explicit messaging connection object.

If the server supports multiple message body formats and the client has requested one of those formats, then the server shall echo the requested message body format in the open explicit response message. If the server does not support multiple message body formats, then the server shall specify its default format in the open explicit messaging connection response.

If no errors are detected, then an open explicit messaging connection success response shall be returned. If an error is detected, then an error response message shall be returned.

5.2.1.5.3 Open explicit messaging connection success response

This service is used to respond to a successful open explicit messaging connection request message (see Figure 9).

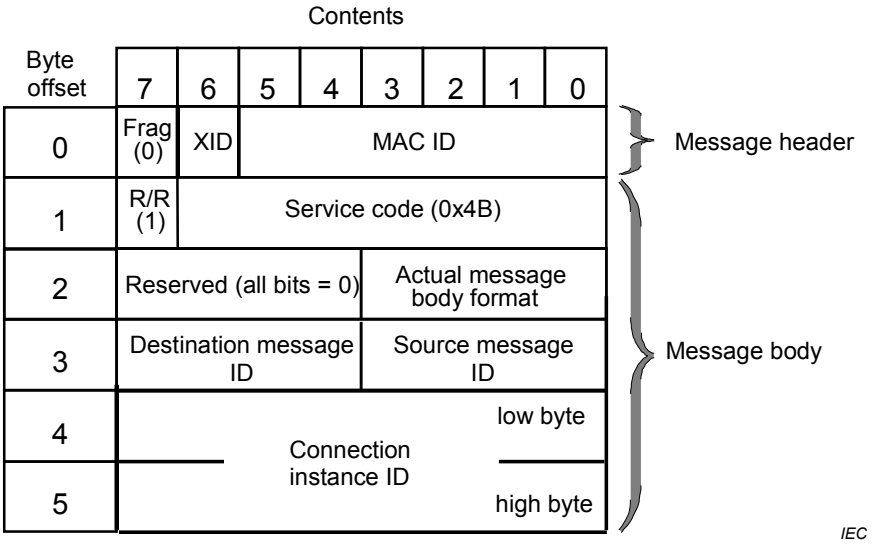


Figure 9 – Open explicit messaging connection response format

Open explicit messaging connection response contents:

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

- **frag (0)/transaction ID/MAC ID:** See 5.2.1.2. The destination MAC ID is always specified in the message header associated with an open explicit messaging connection request/response;
- **R/R bit (1):** Indicates that this is a response message;
- **service code (0x4B):** Identifies this message as an open explicit messaging connection service;
- **reserved bits:** These shall be set to zero by the transmitter;
- **actual message body format:** The field used by the server to define the format of the message body associated with subsequent explicit messages transmitted over this connection (as described in Table 1);
- **destination message ID:** The use of this field depends on the message group across which the connection takes place (as described in Table 4);

Table 4 – Destination message ID in open explicit messaging connection response

If group select within the open request was set to:	Then the destination message ID in the open response:
0 or 3	Is ignored and shall be set to the value zero (0)
1	Is used by the client in combination with the server's MAC ID to generate the connection ID it specifies when transmitting across this connection

- **source message ID:** The message ID value that the server has allocated. The server allocates a message ID from its group 1, 2, or 3 message ID pool that is used in conjunction with its own MAC ID (source MAC ID) to generate the connection ID that is specified when it transmits a message across this connection;
- **connection instance ID:** The server creates an explicit messaging connection object when it successfully services an open request. This field holds the instance ID value (16 bit integer field) assigned to that explicit messaging connection object.

5.2.1.5.4 Close connection request

This service is used to terminate a connection (either I/O or messaging) within one of the end points. The reception of the close connection request message by the UCMM results in the invocation of the connection class's delete service (see IEC 61158-5-2:2014, 6.2.3). A close connection request message is transmitted as an unconnected request message.

The responder verifies that the specified connection instance exists. If the connection instance exists, and it can be deleted, then it shall be deleted. All resources associated with the connection instance are freed. If the request is successfully serviced, then a close connection response shall be returned. If the request is not successful, an error response shall be returned.

Close connection request contents (see Figure 10):

- **frag (0)/transaction ID/MAC ID:** See 5.2.1.2;
- **R/R bit (0):** Indicates that this is a request message;
- **service code (0x4C):** Identifies that this is a close connection service;
- **connection instance ID:** The field that specifies the connection instance to be deleted. The format for the connection instance ID within this message is always specified as a 16 bit integer.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

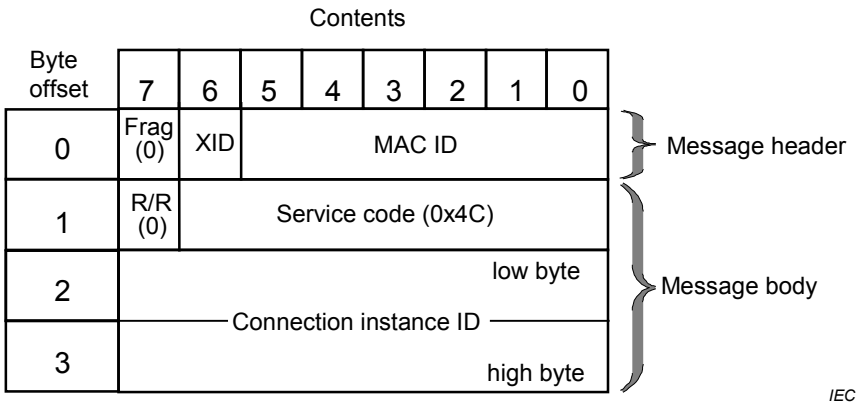


Figure 10 – Close connection request format

5.2.1.5.5 Close connection response

This service is used to respond to a successful close connection request message.

Close connection response format contents (see Figure 11):

- **frag (0)/transaction ID/MAC ID:** See 5.2.1.2;
- **R/R bit (1):** Indicates that this is a response message;
- **service code (0x4C):** Identifies this message as a close connection service.

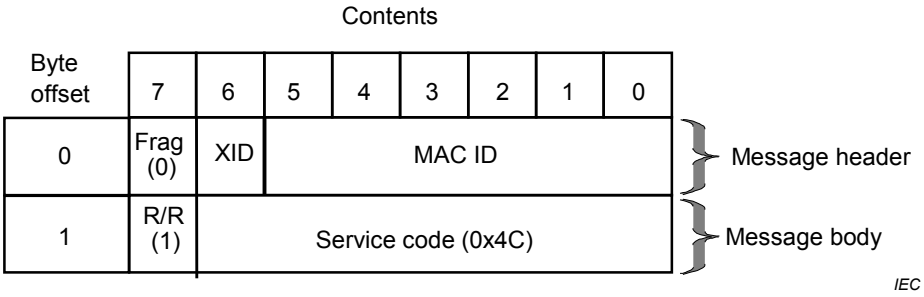


Figure 11 – Close connection response format

5.2.1.5.6 Error response

Table 5 shows a standard set of UCMM related error conditions and the error code (both general error code and additional error code) information to be used in an associated error response message. The format of an error response message is described in 5.2.1.6.5.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Table 5 – UCMM error conditions/codes

Error condition	General error name	General error code	Additional error code
Service code not “open” or “close”	Service not supported	0x08	0xFF
Group select resource error	Resource unavailable	0x02	0x01
Group select out of range	Invalid parameter	0x20	0x01
No server connections available	Resource unavailable	0x02	0x02
No server message IDs available	Resource unavailable	0x02	0x03
Client source message ID invalid	Invalid parameter	0x20	0x02
Duplicate client source message ID	Resource unavailable	0x02	0x04
Connection instance ID invalid	Object does not exist	0x16	0xFF

Error condition descriptions:

– **service not “open” or “close”:** A service received across the UCMM port is not of type “open” or “close” and is therefore not supported by UCMM;

– **group select resource error:** The group select argument indicates utilising a message group which is not supported by the device;

– **group select out of range:** The group select field contains an invalid value;

– **no server connections available:** The maximum number of connections supported by this server has already been reached;

– **no server message IDs available:** The server has allocated all available message IDs within the message group requested by the client;

– **client source message ID invalid:** The source message ID received with an open explicit message connection request is invalid for the specified message group;

– **duplicate client source message ID:** The source message ID and source MAC ID received within an open explicit messaging connection request are already in use for a group 1 or group 3 explicit messaging connection;

– **connection instance ID invalid:** The connection instance ID received with the close connection request does not exist.

5.2.1.6 Connection based explicit messaging

5.2.1.6.1 General

A connection based explicit message is a message transmitted over an explicit messaging connection. A connection based explicit message shall conform to the formats described in this subclause.

5.2.1.6.2 Explicit request message contents (message body format values 0 – 3)

Figure 12 shows the format for the message associated with a non-fragmented explicit request for message body format values 0 – 3:

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

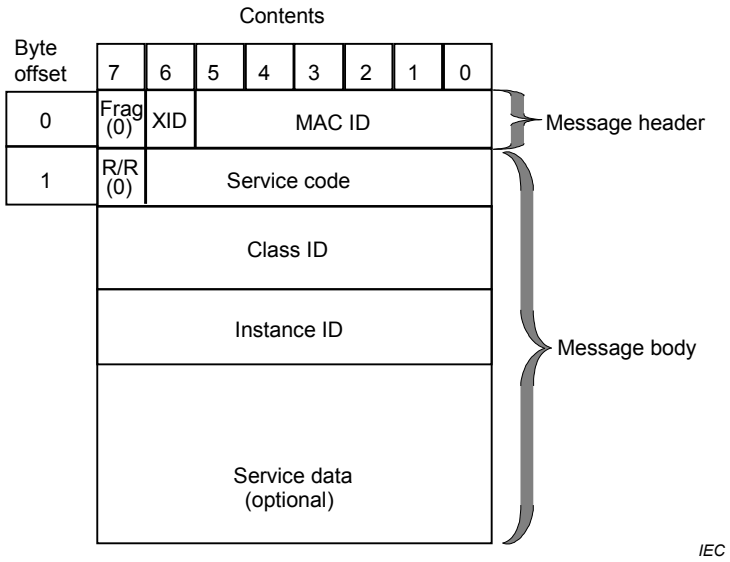


Figure 12 – Non-fragmented explicit request message format, values 0 – 3

Non-fragmented explicit request message contents:

- **frag (0)/transaction ID/MAC ID:** See 5.2.1.2;
- **R/R bit (0):** Indicates that this is a request message;
- **service code:** Defines the service being requested;
- **class ID:** Defines the object class towards which this request is directed. The class ID is specified within either an 8 bit or 16 bit integer field based on the actual message body format value returned in the open explicit messaging connection response;
- **instance ID:** Defines the particular instance within the object class towards which this request is directed. The instance ID is specified within either an 8 bit or 16 bit integer field based on the actual message body format value returned in the open explicit messaging connection response. DeviceNet uses the value zero to denote that the request is directed towards the class itself rather than a specific instance within the class;
- **service data:** Carries request specific data. Formats for the DeviceNet common services are described in IEC 61158-6-2:2014, 4.1.8. Class-specific and object-specific service definitions include the format of this field.

5.2.1.6.3 Explicit request message contents (message body format value 4)

Figure 13 shows the format for the message associated with a non-fragmented explicit request for message body format value 4:

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

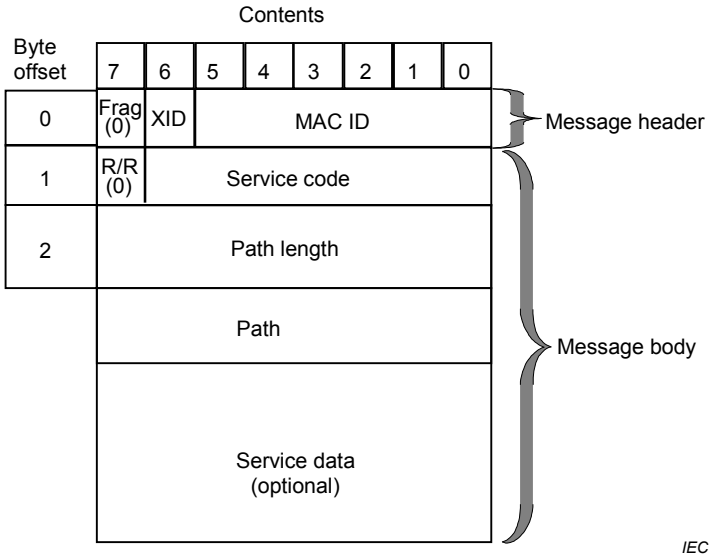


Figure 13 – Non-fragmented explicit request message format, value 4

Non-fragmented explicit request message contents:

- **frag (0)/transaction ID/MAC ID:** See 5.2.1.2;
- **R/R bit (0):** Indicates that this is a request message;
- **service code:** Defines the service being requested;
- **path length:** This 8-bit integer value (USINT) provides the length of the message request path in bytes;
- **path:** Defines the path of the message request (packed EPATH);
- **service data:** Carries request specific data. Formats for the DeviceNet common services are described in IEC 61158-6-2:2014, 4.1.8. Class-specific and object-specific service definitions include the format of this field.

5.2.1.6.4 Success response explicit message

Figure 14 shows the format for the message associated with a non-fragmented success response:

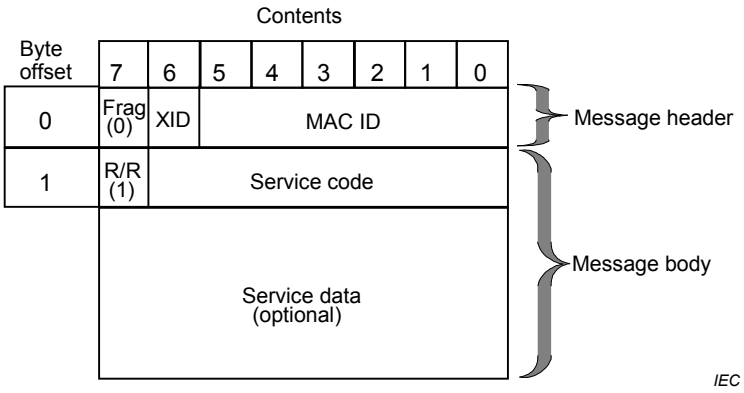


Figure 14 – Non-fragmented success response message format

Non-fragmented success response message contents:

- **frag (0)/transaction ID/MAC ID:** See 5.2.1.2;

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

– **R/R bit (1):** Indicates that this is a response message;

– **service code:** Contains the service code sent in the request message;

– **service data**: Carries request-specific data.

5.2.1.6.5 Error response message

The error response message is returned when an error is encountered while attempting to service a previously received explicit request message. The error response may be sent as either a connection based (request was received across an explicit messaging connection) or unconnected (request was an unconnected explicit request message) response message. Figure 15 shows the format of an error response message.

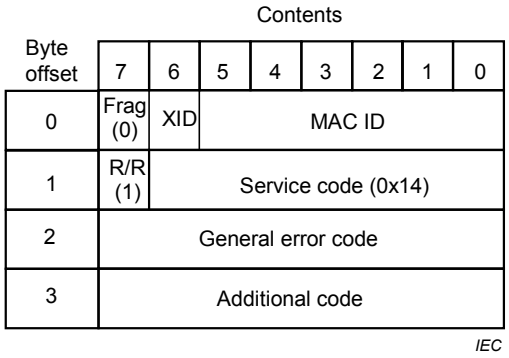


Figure 15 – Error response message

Error response message contents:

– **frag (0)/transaction ID/MAC ID:** See 5.2.1.2;

– **R/R bit (1):** Indicates that this is a response message;

– **service code (0x14):** Identifies this message as an error response;

– **general error code:** Identifies the encountered error. See IEC 61158-5-2:2014, 6.2.1.3.3 and IEC 61158-6-2:2014, 4.1.11 for a list of general error codes;

– **additional code:** Contains an object-specific or service-specific value that further describes the error condition. If the responding object has no additional information to specify, then the value 0xFF shall be placed within this field.

5.2.2 Input/output messaging

DeviceNet defines a fragmentation protocol for transfer of an I/O message greater than eight bytes in length. This is the only protocol information carried within the data field of an I/O message (see Figure 16).

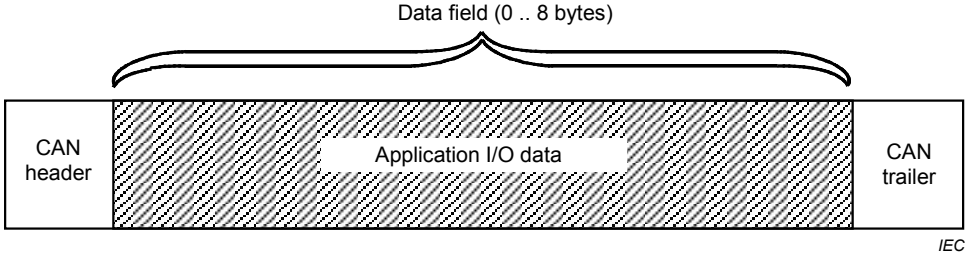


Figure 16 – Data field of an I/O message

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

5.2.3 Fragmentation/reassembly

5.2.3.1 General

This subclause defines the means by which a message whose length is greater than eight bytes is fragmented and reassembled. The fragmentation/reassembly function is provided by the DeviceNet connection object.

The logic that triggers a fragmented transmission is:

– explicit messaging connection object instances examine the length of each message to be transmitted. If the message is greater than eight bytes in length, then the fragmentation protocol shall be used;

– I/O connection object instances examine the produced_connection_size attribute of the connection object (see 5.3.2). If the produced_connection_size attribute is greater than eight, then the fragmentation protocol shall be used.

Two types of fragmentation are defined:

– **acknowledged:** performed when fragmenting an explicit message;

– **unacknowledged:** performed when fragmenting an I/O message.

5.2.3.2 Fragmentation protocol

The fragmentation protocol is located within a single byte in the CAN data field, and is formatted as shown in Figure 17:

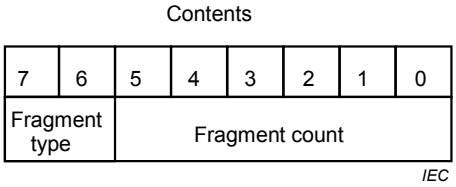


Figure 17 – Format of DeviceNet fragmentation protocol

Fragmentation protocol contents:

– **fragment type:** Indicates whether this is the first fragment, one of the middle fragments, or the last fragment (see Table 6).

Table 6 – Fragment type bit values

Value	Meaning
0	First fragment. The fragment count field shall contain the value 0 or 0x3F [a]
1	Middle fragment [b]
2	Last fragment [c]
3	Fragment acknowledge [d]

[a] If the fragment count contains the value zero (0), then this is the first in a series of fragments. If the fragment count field contains the value 0x3F, then this is also the last transmission in the series.

[b] This fragment is neither the first nor the last fragment in the series.

[c] Marks this as the last fragment.

[d] The value the receiver of a fragmented message uses to acknowledge the reception of a fragment.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

– **fragment count:** Marks each separate fragment such that the receiver may determine whether or not a fragment has been missed. If the fragment type is first fragment, then this field takes on a special meaning (as described in Table 6). The fragment count is increased by one for each successive fragment in a series and resets to zero at overflow (fragment count = (fragment count + 1) mod 64).

For I/O message fragmentation, the fragmentation protocol information is placed within byte offset 0 (see Figure 18).

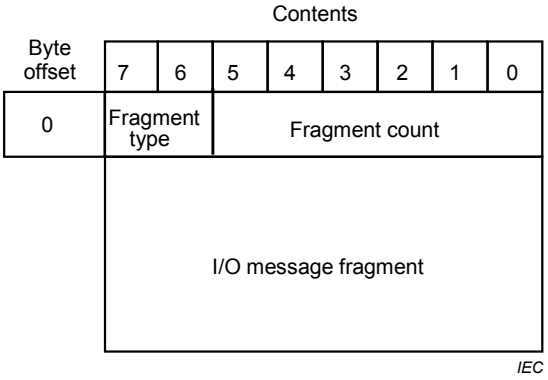


Figure 18 – I/O message fragment format

For explicit message fragmentation, the fragmentation protocol information is placed within byte offset 1 (see Figure 19).

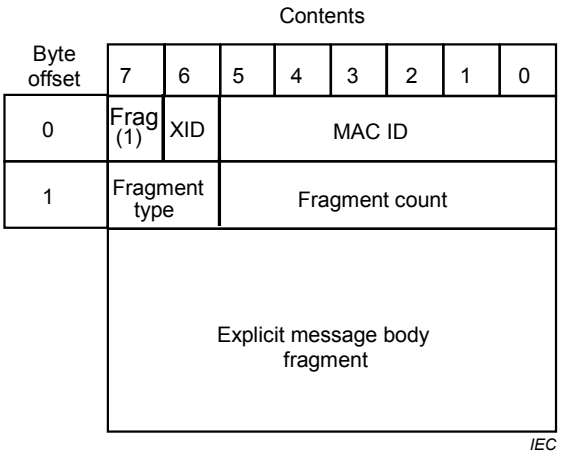


Figure 19 – Explicit message fragment format

The Frag bit within the message header in Figure 19 is set to 1 to indicate that this is a piece of the explicit message, not the entire message. The value 1 here also indicates that the next byte contains the fragmentation protocol.

The receiver of a fragmented series of transmissions parses the fragmented message as defined in this subclause. This procedure applies to both I/O and explicit message fragmentation.

If the first transmission is expected by the connection and the fragment type is equal to first fragment:

– if the fragment count is 0x3F, then this is the only transmission in the series, and the connection processes the message and awaits the beginning of a new series;

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

– if the fragment count is 0, then this is the first in a series of transmissions and the connection stores the fragment and saves the fragment count.

If the first transmission is expected by the connection and either the fragment type is not equal to first fragment or the fragment count is not equal to 0 or 0x3F, the connection discards the transmission and awaits the beginning of a new message.

If the first transmission is not expected by the connection, then the connection verifies:

– that the fragment type is not first fragment; and

– that the fragment count is numerically one (1) greater than the previous value received.

If one of these checks fails, then an error has been detected. If both checks pass, then the fragment is appended to the previously received fragment(s), and the fragment type is parsed to determine whether or not more fragments are to be expected.

If more fragments are forthcoming, then the connection saves the received fragment count and waits for the next fragment. If this is the last transmission in the series and an error has not been detected, then the connection processes the message and resets to looking for the beginning of a new series.

If an explicit message is being fragmented, then the receiver shall generate and transmit an acknowledgement after the reception of each fragment.

If an error is detected, then error recovery specific to whether this is a fragmented I/O or explicit message takes place.

If the detection of a missed fragment was triggered by the reception of the first fragment in the next series, then any processing associated with the current series is immediately discontinued, the fragments stored in memory are discarded, and processing immediately begins on the new series.

5.2.3.3 Unacknowledged fragmentation

Fragmentation of an I/O message is performed in an unacknowledged fashion. Unacknowledged fragmentation consists of the back-to-back transmission of the fragments from the transmitting module. The receiving module(s) returns no acknowledgements.

When an I/O connection's send_message service is invoked, it examines its connection_size attribute to determine whether or not a fragmented series of messages is to be transmitted. If the connection_size attribute is greater than eight (8), then the fragmentation protocol is placed within the I/O.

If the connection_size attribute is less than or equal to eight (8) bytes, then the raw data is transmitted without the presence of the fragmentation protocol.

If the application requests to transmit a piece of data whose length is greater than the connection_size attribute, then an internal error is indicated and the transmission does not occur. If the receiving I/O connection object detects a missed fragment by determining that the received fragment count is not equal to the previously received fragment count plus one (1), then the following error recovery steps are taken:

– all subsequent fragments in this series are dropped and the application is not informed of an I/O message reception;

– the connection object begins looking for the beginning of a new fragmented series of transmissions and discards the remaining fragments in this series.

Documents
Document Sharing Hub

Full document details: IEC 62026-3 2014

5.2.3.4 Acknowledged fragmentation

Acknowledged fragmentation is used for fragmented explicit messages. Acknowledged fragmentation consists of the transmission of a fragment from the transmitting node followed by the transmission of an acknowledgement by the receiving node. The receiving node acknowledges the reception of each fragment.

Figure 20 shows the format for the acknowledgement message transmitted by the receiver after each explicit message fragment is received.

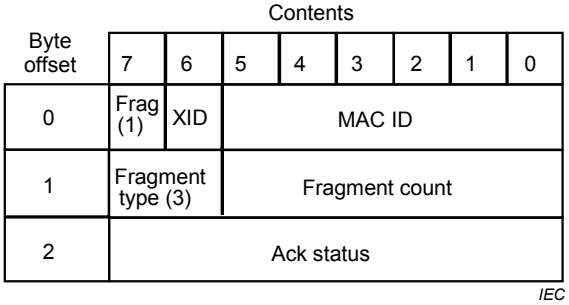


Figure 20 – Acknowledgement message format

Acknowledgement message contents:

– **fragment type:** Indicates that this is a fragment acknowledge by placing the value 3 within this field;

– **fragment count:** Echoes the last fragment count value received;

– **ack status:** Indicates whether or not an error has been detected by the receiver of the fragmented message. The ack status bit values are defined in Table 7:

Table 7 – Ack status bit values

Value	Meaning
0	Success. No errors have been detected and the fragmented transmission shall continue
1	Too much data. The maximum amount of data the receiver can receive across this connection has been exceeded
2 – 255	Reserved

The transmitting node functions as specified below:

– the connection object formulates the message header by setting the Frag bit to one (1), the XID (transaction ID) field to the value specified by the application, and initializes the MAC ID field in the same manner it would a non-fragmented explicit message. The message headers associated with each separate fragment are identical;

– the connection object then places the appropriate fragmentation protocol information into the message. The connection object stores the fragment count that was inserted into the message;

– the connection object then takes the next piece of the message body and places it into the message;

– the message is transmitted and a wait for acknowledgement timer is started. The amount of time to wait is application specific;

– if the wait for acknowledgement timer expires, then the connection object automatically retries the last transmission. The connection object retries once. If the timer expires once again (two consecutive wait for acknowledgement time-outs), then the application is

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

informed that an error has been detected and the requested transmission cannot take place;

– if an acknowledgement is received, then the connection object determines whether or not the fragment count in the acknowledgement is equal to the last fragment count it transmitted. If they are equal, then the fragment was successfully delivered and acknowledged, and normal processing continues. If the values are not equal, then the connection object continues to wait for an acknowledgement with a matching fragment count.

The initial state associated with the receiving module entails waiting for either the first fragment in a fragmented transmission or waiting for the reception of a complete explicit message. The receiving side functions as specified below:

– If the message header indicates that this is a fragmented portion of an explicit message, then the connection object examines the fragmentation protocol to determine its validity. If the connection has yet to receive the first transmission in the series (in the initial state) and the fragment type field is not equal to the first fragment, then the fragment is dropped and no acknowledgement is returned;

– If the fragment type indicates that this is the first fragment, then the fragment count shall equal zero (0) or 0x3F. If the fragment count is 0x3F, then this is the only transmission in the series and the connection does the following:

  • processes the message;

  • resets to looking for the beginning of a new series.

  If the fragment count is zero (0), then this is the first in a series of transmissions and the connection stores the fragment and saves the fragment count. If the fragment type indicates first fragment and the fragment count is not zero (0) or 0x3F, then the fragment is dropped and no acknowledgement is returned;

– If the fragment count is one (1) greater than the previously received count and the fragment type does not indicate that this is the first fragment, then the next fragment has been received. The fragment is appended to the previously received fragment(s) and an acknowledgement is returned. The fragment count associated with this fragment is stored;

– If the fragment count is neither one (1) greater nor equal to the previously received count, then the fragment is discarded and no acknowledgement is returned. The receiver resets to the initial state;

– When the final fragment is received and the acknowledgement is transmitted, the connection object continues processing the message as if it were non-fragmented.

5.2.4 Offline connection set

5.2.4.1 General

This sublcause describes the offline connection set messaging protocol and presents details associated with the establishment of offline connection set ownership. Support of the offline connection set is optional for all types of devices. The group 4 offline connection set messages are used by client tool(s) to recover nodes in the communication faulted state. Using the offline connection set messages, a client (tool) shall be able to:

a) visually identify the faulted node(s) to which it is communicating with by flashing an LED;

b) send fault recovery messages to the faulted node; and when possible

c) recover the faulted node without having to unplug it from the subnet.

At any point in time, only one (1) client node shall communicate with nodes in the communication faulted state, connected to a single subnet. Ownership of communication faulted nodes is gained via a dialog between clients (tools) using the offline ownership request/response messages.

Table 8 shows the group 4 identifiers associated with the offline connection set.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Table 8 – Offline connection set

Identifier bits (Connection ID)											Identity usage
Group ID					Message ID						
10	9	8	7	6	5	4	3	2	1	0	Group 4 messages
1	1	1	1	1	0x2C						Communication faulted response message
1	1	1	1	1	0x2D						Communication faulted request message
1	1	1	1	1	0x2E						Offline ownership response message
1	1	1	1	1	0x2F						Offline ownership request message

Only clients wishing to support the offline connection set shall produce messages using group 4 message ID 0x2F and consume response messages with group 4 message ID 0x2E. Once ownership is gained, the client shall produce all messages destined for communication faulted node(s) using group 4 message ID 0x2D.

A client may NOT produce a “Communication faulted request message” until it has gained ownership of the “Offline connection set”.

Once a client (tool) establishes offline connection set ownership, it is able to transmit communication faulted request messages with group 4 message ID 0x2D and receive communication faulted response messages with group 4 message ID 0x2C.

While in the communication faulted state, a node supporting this feature is only required to consume a single connection ID; group 4 message ID 0x2D. A faulted node shall produce its communication fault response messages on group 4 message ID 0x2C.

Offline connection set messages are low priority and may be subject to delays due to other network traffic.

5.2.4.2 Offline ownership

Figure 21 and Figure 22 illustrate the steps involved for a client (tool) to gain ownership of the offline connection set.

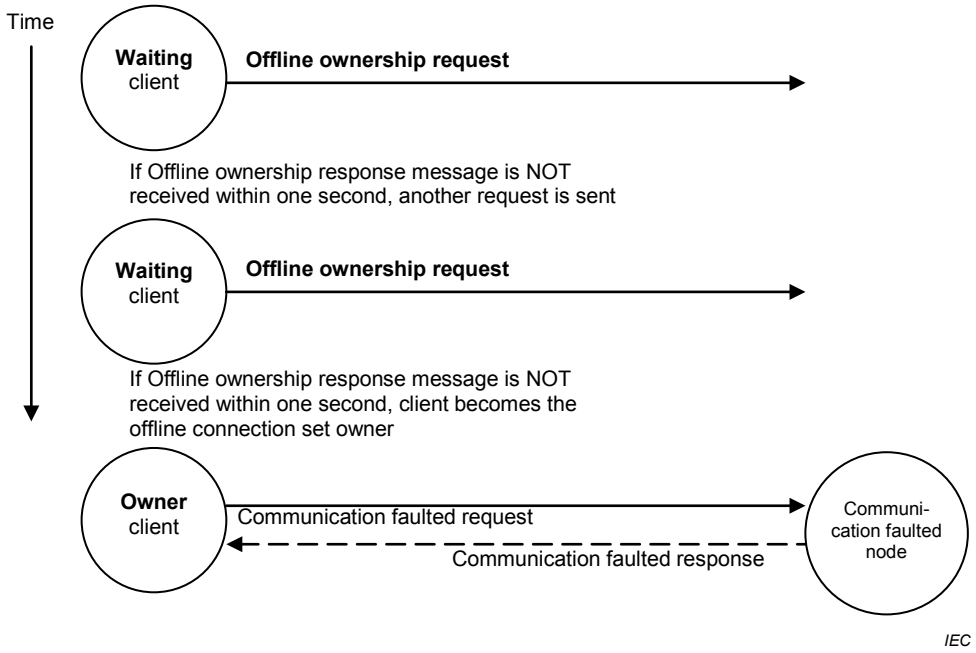


Figure 21 – Establishing the offline ownership

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

To gain control of the offline connection set a client (tool) shall produce its *offline ownership request message*. Upon successful transmission, the client (tool) shall wait for an *offline ownership response message* for a period of 1 s. If a response is not received, it shall produce a second *offline ownership request message* and shall wait another second. If a response is not received, it shall become the owner of the offline request message. If an *offline ownership response message* is received during either wait period, it shall not become the owner of the offline connection set and shall wait to become the owner.

Only one client tool shall have ownership of the offline connection set at any point in time.

The scenario where one client (tool) has previously claimed ownership of the offline connection set, and additional client nodes are arbitrating for ownership, the current owner shall reply to any *offline ownership request messages* with an *offline ownership response message*.

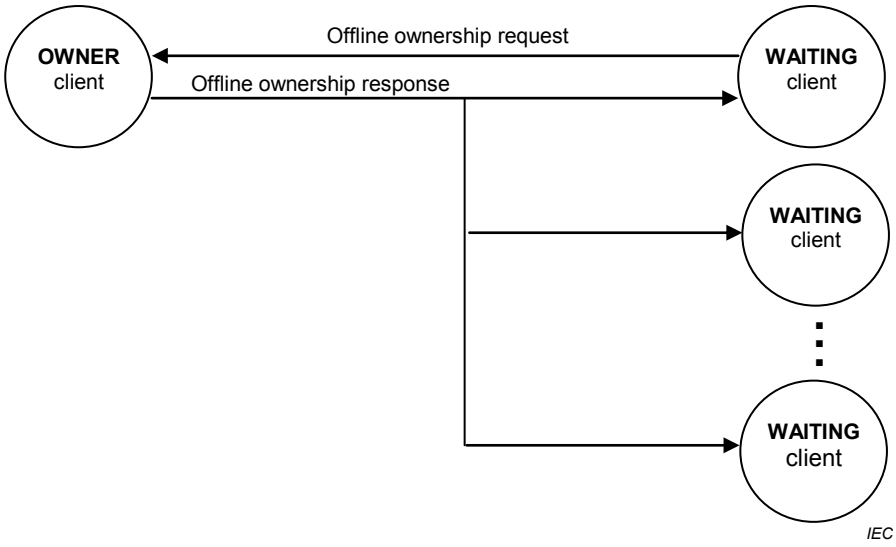


Figure 22 – Multicast nature of the offline ownership

A waiting client shall not attempt to send an offline ownership request message for a minimum of two seconds after receiving an offline ownership response message.

5.2.4.3 Offline ownership messages

5.2.4.3.1 General

The two messages used to manage the offline connection set are only produced and consumed by client nodes supporting this functionality. These messages are only serviced while a client is participating in a recovery activity, otherwise they are ignored by a client.

5.2.4.3.2 Offline ownership request message (client only)

To gain ownership of the offline connection set, a client tool shall produce an *offline ownership request message* with group 4 message ID 0x2F, see Figure 23.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

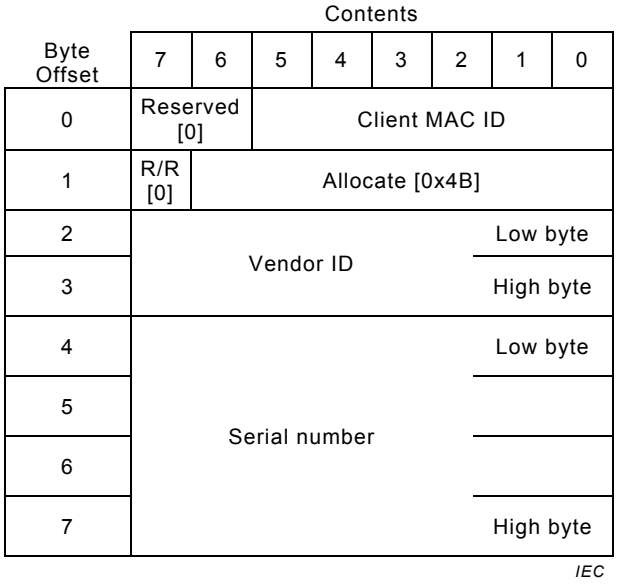


Figure 23 – Offline ownership request message

Upon successful transmission, the client (tool) shall wait for an *offline ownership response message* for at least 1 s. If a response is not received, it shall produce a second *offline ownership request message* and shall wait at least 1 s. If a response is not received, it shall become the owner of the offline connection set. If an *offline ownership response message* is received during either time period, it shall not become the owner of the offline connection set and shall wait to become the owner.

Once a client is the owner of the offline connection set, if it receives an *offline ownership request message*, it shall produce an *offline ownership response message* within 1 s.

If a client is waiting to become the owner of the offline connection set, it shall not produce an *offline ownership request message* at a rate faster than once every 2 s. This 2 s delay shall be reset upon the receipt of any message using an *offline ownership request or response connection ID*.

5.2.4.3.3 Offline ownership response message (client only)

The format of the *offline ownership response message* is the same as the *offline ownership request message*, except it is produced at group 4 message ID 0x2E, see Figure 24, and the R/R bit is set (1).

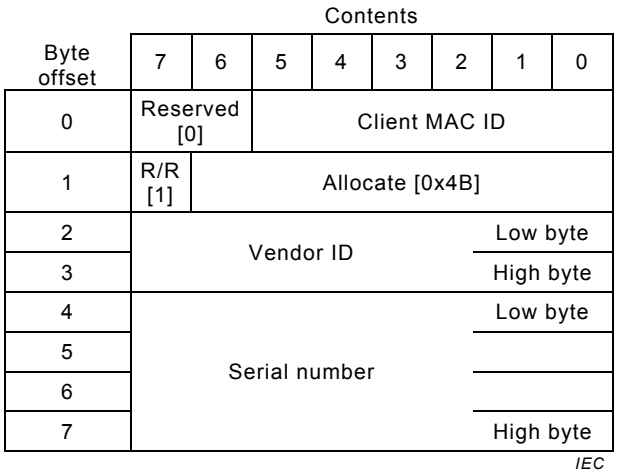


Figure 24 – Offline ownership response message protocol

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Once a client (tool) is in the owner of the offline connection set, it may produce *communication faulted request messages* to all faulted nodes. Any node supporting the offline connection set shall service the *communication faulted request message* only while in the communication faulted state.

5.2.4.4 Communication faulted messages

5.2.4.4.1 General

All nodes in the communication faulted state, which support the faulted node recovery mechanism, shall receive the *communication faulted request messages* produced at group 4 message ID 0x2D. When appropriate, communication faulted nodes shall produce a *communication faulted response message* at group 4 message ID 0x2C.

Communication faulted request messages are consumed by all faulted nodes supporting this functionality. Depending upon the request message, any number of the faulted nodes may reply to a single request.

5.2.4.4.2 Communication faulted message protocols

To support the *communication faulted message* protocols, a node shall support the “Network Status” LED or “Combined Module/Network Status” LED, or have a suitable device-specific method of indicating the network and module status externally (such as a display screen or a text readout). It shall also have a network settable MAC ID.

The protocol of all the *communication faulted request messages* fall into two forms, multicast protocol (see Figure 25) and point-to-point protocol (see Figure 26).

	Contents							
Byte offset	7	6	5	4	3	2	1	0
0	Reserved [0]	Match	Value					
1	R/R [0]	Service						

IEC

Figure 25 – Communication faulted request message – Multicast protocol

The multicast protocol is used by a client to perform requests to all communication faulted node(s) to acquire their serial number(s) and vendor ID(s). Once this information is known by the client, a point-to-point protocol may be used.

When a communication faulted node receives a multicast *communication faulted request message*, it shall perform the following checks to determine whether the message should be accepted and serviced:

– Multicast match protocol:

  IF match bit is set (1);

  AND value equals its MAC ID;

  AND the length of the message is two (2) bytes;

  THEN accept and service the *communication faulted request message.*

– Multicast mask protocol:

  IF match bit is reset (0);

  AND value (logically AND’d with its MAC ID) equals its MAC ID;

  AND the length of the message is two (2) bytes;

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

THEN accept and service the *communication faulted request message.*

The multicast mask protocol (logical AND) is used for dialogs where a range of MAC IDs are requested to respond. Table 9 indicates some, but not all, of the 64 possible mask values.

Table 9 – Addresses reporting based upon mask

5	4	3	2	1	0	Faulted nodes with
1	1	1	1	1	1	MAC ID < 64
0	1	1	1	1	1	MAC ID < 32
0	0	1	1	1	1	MAC ID < 16
0	0	0	1	1	1	MAC ID < 8
0	0	0	0	1	1	MAC ID < 4
0	0	0	0	0	1	MAC ID < 2

The multicast mask protocol is used to reduce the number of messages, which shall be sent on a subnet to determine if, and at what MAC ID, a communication faulted node resides.

The point-to-point protocol follows.

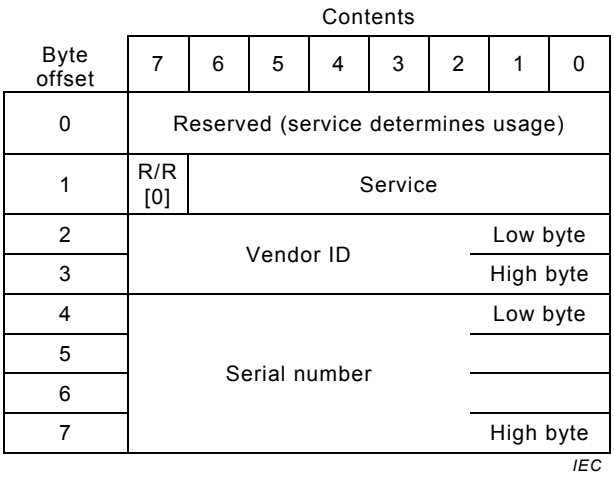


Figure 26 – Communication faulted request message – Point-to-point protocol

When a communication faulted node receives a point-to-point *communication faulted request message*, it shall perform the following checks to determine whether the message should be accepted and serviced:

Point-to-point match protocol:

IF the length of the message is equal to eight (8) bytes;

AND the serial number and vendor ID match its own;

THEN accept and service the *communication faulted request message.*

The following services are currently defined for the *communication faulted request messages*:

Identify *communication faulted request message* – Multicast protocol:

Used when the client is attempting to identify the existence of any communication faulted nodes, which support the offline connection set feature, on a subnet.

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

Identify *communication faulted request message* – Point-to-point protocol:

Used when the client detects multiple communication faulted nodes on a subnet, and wishes to visually identify a specific communication faulted node.

Who *communication faulted request message*:

Used when the client has detected communication faulted node(s) on a subnet, and wishes to acquire their serial number and vendor ID.

Change MAC ID *communication faulted request message*:

Used when the client has detected a specific communication faulted node and wishes to change its MAC ID and have it attempt to come On-line at a specified MAC ID.

A device that implements the *communication faulted message* protocols shall implement support for all four of the above communication faulted message protocols.

5.2.4.5 Identify communication faulted messages

5.2.4.5.1 General

The client may solicit an identify *communication faulted response message* from:

- ALL communication faulted nodes; or
- communication faulted nodes at one or more specific MAC IDs; or
- a communication faulted node with a specific vendor ID and serial number.

5.2.4.5.2 Identify request message – Multicast protocol

The *identify communication faulted request message* – multicast protocol shall be produced at group 4 message ID 0x2D, see Figure 27. The client produces this message when it is attempting to identify the existence of any nodes in the communications faulted state.

If the length of the *identify communication faulted request message* is two (2) bytes, any communication faulted node(s) servicing this message shall respond but not flash their LEDs.

	Contents							
Byte offset	7	6	5	4	3	2	1	0
0	Reserved [0]	Match	Value					
1	R/R [0]	Identify [0x4C]						

IEC

Figure 27 – Identify communication faulted request message – Multicast protocol

5.2.4.5.3 Identify response message – Multicast protocol

The format of the *identify communication faulted response message* is the same as the received *identify communication faulted request message* (with the match and value information the same as the information in the received request), except it is produced at group 4 message ID 0x2C, see Figure 28 and the R/R bit is set (1). The response message shall be initiated within 100 ms, upon the receipt of the request message.

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014

Byte offset	Contents 7	6	5	4	3	2	1	0
0	Reserved [0]	Match	Value					
1	R/R [1]	Identify [0x4C]						

IEC

Figure 28 – Communication faulted identify response message

If more than one node responds to the request, collisions will not occur on the subnet. This is a result of all communication faulted nodes producing the same message (CAN ID and data field). However, when more than one node replies, multiple messages may be received by the client.

5.2.4.5.4 Identify request message – Point-to-point protocol

The *identify communication faulted request message* – point-to-point protocol shall be produced with group 4 message ID 0x2D, see Figure 29. If a client (tool) detects communication faulted node(s) on a subnet, and the user wishes to visually identify a communication faulted node at a specific MAC ID, then the client (tool) shall produce the *identify communication faulted request message* – point-to-point protocol.

Byte offset	Contents 7	6	5	4	3	2	1	0
0	Reserved [0]	Match [0]	Value [0x3F]					
1	R/R [0]	Identify [0x4C]						
2	Vendor ID							Low byte
3								High byte
4	Serial number							Low byte
5								
6								
7								High byte

IEC

Figure 29 – Identify communication faulted request message – Point-to-point protocol

A client shall always set “match” to 0 and “value” to the 0x3F. A server may ignore the contents of byte 0 (checking vendor ID and serial number are sufficient for uniqueness).

IF a node accepts an *identify communication faulted request message* – point-to-point protocol AND is in the communication faulted state, it shall respond as follows:

– Flash its bi-colour network status LED or combined module/network status LED. The flash rate shall alternate the red LED ON for 250 ms, followed by the green LED turning ON for 250 ms. If the device does not have a bi-colour network status LED or combined module/network status LED, a suitable device-specific externally viewable indication shall be provided (e.g. via a display screen) to show that the *identify message* has been accepted;

– Produce an identify *communication faulted response message*, within 100 ms.

IF a node has accepted an *identify communication faulted request message* – point-to-point protocol AND is flashing its LED, it shall stop flashing its LED under the following conditions

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

Similarly, if the device does not have a bi-colour network status LED or combined module/network status LED and the device-specific indication used to indicate that an *identify message* had been accepted is ON, it shall be turned OFF under these conditions:

– Another *identify communication faulted request message* – point-to-point protocol is not accepted within 500 ms.
– Another *identify communication faulted request message* – point-to-point protocol is received and NOT accepted (and another serial number or vendor ID).

If a node uses a network status LED or combined module/network status LED for the *identify message* indication, then, after it stops flashing the LED, the node shall turn off the LED for a minimum of 500 ms, but less than 1 s, prior to returning to its normal operation.

5.2.4.5.5 Identify response message – Point-to-point protocol

The *identify communication faulted response message* – point-to-point protocol is the same as the *identify communication faulted response message* – multicast protocol response.

5.2.4.5.6 Who communication faulted request message

The *who communication faulted request message* shall be produced at group 4 message ID 0x2D, see Figure 30. The *who communication faulted request message* is used to acquire the serial number and vendor ID of a communication faulted node.

	Contents							
Byte offset	7	6	5	4	3	2	1	0
0	Reserved [0]	Match	Value					
1	R/R [0]	Identify who [0x4B]						
2	Time delay byte offset [0-6]							

IEC

Figure 30 – Who communication faulted request message

IF the consuming node:

– is in the communication faulted state; and
– detects the *who communication faulted request message*; and
– meets the match and value criteria.

THEN the communication faulted node:

– shall wait a duration derived from the time delay byte offset; and
– produce a single *who response message* at the group 4 message ID 0x2C; and
– remain in the communication faulted state.

The client shall consume the duplicate MAC ID request message produced at the *communication faulted response* message ID 0x2C, thus acquiring the communication faulted node's serial number and vendor ID.

The accuracy of the wait period shall be dependent upon the timer resolution of the internal timer within the communication faulted node.

The interval the faulted node waits before responding is determined as follows:

– get the value at the time delay byte offset within the *who communication faulted request message* (0-6 are valid values);

AVA Documents Document Sharing Hub

Full document details: IEC 62026-3 2014

– use this value as an offset into the *who response message* shown in Figure 31; then
– multiply the byte value at that offset times 50 ms; and
– delay the response for that period of time.

5.2.4.5.7 Who communication response

The *who response message*, produced following the specified delay, is the same protocol as a duplicate MAC ID request message, except it is produced at a group 4 message ID 0x2C, see Figure 31.

	Contents							
Byte offset	7	6	5	4	3	2	1	0
0	R/R [0]	Physical port number						
1	Vendor ID							Low byte
2								High byte
3	Serial number							Low byte
4								
5								
6								High byte

IEC

Figure 31 – Who response message

5.2.4.5.8 Change MAC ID communication faulted request message

The purpose of this message is to modify the MAC ID of a node in the communication faulted state. This message shall be produced using the group 4 message ID 0x2D, see Figure 32.

	Contents							
Byte offset	7	6	5	4	3	2	1	0
0	Reserved [0]		Value [New MAC ID]					
1	R/R [0]	Change MAC ID [0x4D]						
2	Vendor ID							Low byte
3								High byte
4	Serial number							Low byte
5								
6								
7								High byte

IEC

Figure 32 – Change MAC ID communication faulted request message

Nodes which support the *change_MAC ID communication faulted request message* feature shall not produce a *communication faulted response message*.

IF the consuming node:

– is in the communication faulted state; AND
– detects the *change_MAC ID communication faulted request message*; AND

Documents
Document Sharing Hub
Full document details: IEC 62026-3 2014