

BS EN 62026-3:2015



BSI Standards Publication

# Low-voltage switchgear and controlgear — Controller-device interfaces (CDIs)

Part 3: DeviceNet

**bsi.**

...making excellence a habit.  
**AVA** Documents™  
Document Sharing Hub

## National foreword

This British Standard is the UK implementation of EN 62026-3:2015. It is identical to IEC 62026-3:2014, incorporating corrigendum March 2015. It supersedes BS EN 62026-3:2009 which is withdrawn.

The start and finish of text introduced or altered by corrigendum is indicated in the text by tags. Text altered by IEC corrigendum March 2015 is indicated in the text by AC1 AC1.

The UK participation in its preparation was entrusted by Technical Committee PEL/121, Switchgear and Controlgear and their assemblies for low voltage, to Subcommittee PEL/121/1, Low voltage switchgear and controlgear.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2015  
Published by BSI Standards Limited 2015

ISBN 978 0 580 78536 8

ICS 29.130.20; 35.200

**Compliance with a British Standard cannot confer immunity from legal obligations.**

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 May 2015.

## Amendments/corrigenda issued since publication

| Date | Text affected |
|------|---------------|
|------|---------------|

---

EUROPEAN STANDARD  
NORME EUROPÉENNE  
EUROPÄISCHE NORM

**EN 62026-3**

April 2015

ICS 29.130.20

Supersedes EN 62026-3:2009

English Version

**Low-voltage switchgear and controlgear - Controller-device  
interfaces (CDIs) - Part 3: DeviceNet  
(IEC 62026-3:2014 + COR1:2015)**

Appareillage à basse tension - Interfaces appareil de  
commande-appareil (CDI) - Partie 3: DeviceNet  
(IEC 62026-3:2014 + COR1:2015)

Niederspannungsschaltgeräte - Steuerung-Geräte-  
Netzwerke (CDIs) - Teil 3: DeviceNet  
(IEC 62026-3:2014 + COR1:2015)

This European Standard was approved by CENELEC on 2014-10-29. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



European Committee for Electrotechnical Standardization  
Comité Européen de Normalisation Electrotechnique  
Europäisches Komitee für Elektrotechnische Normung

**CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels**

© 2015 CENELEC All rights of exploitation in any form and by any means reserved worldwide for CENELEC Members.

## Foreword

The text of document 17B/1814/CDV, future edition 3 of IEC 62026-3 + Corrigendum March 2015, prepared by SC 121A "Low-voltage switchgear and controlgear" of IEC/TC 121 " Switchgear and controlgear and their assemblies for low voltage" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 62026-3:2015.

The following dates are fixed:

- latest date by which the document has to be (dop) 2015-10-03  
implemented at national level by  
publication of an identical national  
standard or by endorsement
- latest date by which the national (dow) 2017-09-26  
standards conflicting with the  
document have to be withdrawn

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association, and supports essential requirements of EU Directive.

For the relationship with EU Directive see informative Annex ZZ, which is an integral part of this document.

This standard covers the Principle Elements of the Safety Objectives for Electrical Equipment Designed for Use within Certain Voltage Limits (LVD - 2006/95/EC).

## Endorsement notice

The text of the International Standard IEC 62026-3:2014 + Corrigendum March 2015 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

|                     |      |                                       |
|---------------------|------|---------------------------------------|
| IEC 61131-3:2013    | NOTE | Harmonized as EN 61131-3:2013.        |
| IEC 61158 (Series)  | NOTE | Harmonized as EN 61158 (Series).      |
| IEC 61508 (Series)  | NOTE | Harmonized as EN 61508 (Series).      |
| IEC 61784-1         | NOTE | Harmonized as EN 61784-1.             |
| IEC 61784-5-2       | NOTE | Harmonized as EN 61784-5-2..          |
| ISO/IEC 7498-1:1994 | NOTE | Harmonized as EN ISO/IEC 7498-1:1995. |

## Annex ZA (normative)

### Normative references to international publications with their corresponding European publications

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.

NOTE 1 When an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: [www.cenelec.eu](http://www.cenelec.eu).

| Publication       | Year | Title                                                                                                                                                       | EN/HD                           | Year |
|-------------------|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|------|
| IEC 60529         | 1989 | Degrees of protection provided by enclosures (IP Code)                                                                                                      | EN 60529                        | 1991 |
|                   |      |                                                                                                                                                             | +EN                             | 1993 |
|                   |      |                                                                                                                                                             | 60529:1991/corrigendum May 1993 |      |
| IEC 60529:1989/A1 | 1999 |                                                                                                                                                             | EN 60529:1991/A1                | 2000 |
| IEC 60529:1989/A2 | 2013 |                                                                                                                                                             | EN 60529:1991/A2                | 2013 |
| IEC 60947-5-2     | 2007 | Low-voltage switchgear and controlgear -- Part 5-2: Control circuit devices and switching elements - Proximity switches                                     | EN 60947-5-2                    | 2007 |
| +A1               | 2012 |                                                                                                                                                             | +A1                             | 2012 |
| IEC 61000-4-2     | 2008 | Electromagnetic compatibility (EMC) -- Part 4-2: Testing and measurement techniques - Electrostatic discharge immunity test                                 | EN 61000-4-2                    | 2009 |
| IEC 61000-4-3     | 2006 | Electromagnetic compatibility (EMC) -- Part 4-3: Testing and measurement techniques - Radiated, radio-frequency, electromagnetic field immunity test        | EN 61000-4-3                    | 2006 |
| +A1               | 2007 |                                                                                                                                                             | +A1                             | 2008 |
| +A2               | 2010 |                                                                                                                                                             | +A2                             | 2010 |
| IEC 61000-4-4     | 2012 | Electromagnetic compatibility (EMC) -- Part 4-4: Testing and measurement techniques - Electrical fast transient/burst immunity test                         | EN 61000-4-4                    | 2012 |
| IEC 61000-4-5     | 2005 | Electromagnetic compatibility (EMC) -- Part 4-5: Testing and measurement techniques - Surge immunity test                                                   | EN 61000-4-5                    | 2006 |
| IEC 61000-4-6     | 2013 | Electromagnetic compatibility (EMC) -- Part 4-6: Testing and measurement techniques - Immunity to conducted disturbances, induced by radio-frequency fields | EN 61000-4-6                    | 2014 |
| IEC 61158-4-2     | 2014 | Industrial communication networks - Fieldbus specifications - Part 4-2: Data-link layer protocol specification - Type 2 elements                            | EN 61158-4-2                    | 2014 |
| IEC 61158-5-2     | 2014 | Industrial communication networks - Fieldbus specifications -- Part 5-2: Application layer service definition - Type 2 elements                             | EN 61158-5-2                    | 2014 |

|                       |      |                                                                                                                                             |                                 |      |
|-----------------------|------|---------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|------|
| IEC 61158-6-2         | 2014 | Industrial communication networks -<br>Fieldbus specifications - Part 6-2:<br>Application layer protocol specification -<br>Type 2 elements | EN 61158-6-2                    | 2014 |
| IEC 61784-3-2         | -    | Industrial communication networks -<br>Profiles -- Part 3-2: Functional safety<br>fieldbuses - Additional specifications for<br>CPF 2       | EN 61784-3-2                    | -    |
| IEC 62026-1           | 2007 | Low-voltage switchgear and controlgear -<br>Controller-device interfaces (CDIs) -- Part<br>1: General rules                                 | EN 62026-1                      | 2007 |
| ISO 11898-1           | 2003 | Road vehicles_ - Controller area network<br>(CAN)_ - Part_1: Data link layer and<br>physical signalling                                     | -                               | -    |
| ISO 11898-2           | 2003 | Road vehicles_ - Controller area network<br>(CAN)_ - Part_2: High-speed medium<br>access unit                                               | -                               | -    |
| ANSI B93.55M-<br>1981 |      | Hydraulic Fluid Power Solenoid-piloted<br>Industrial Valves - Interface Dimensions<br>for Electrical Connectors                             | -                               | -    |
| ASTM D 4566-94        | -    | Standard Test Methods for Electrical<br>Performance Properties of Insulations and<br>Jackets for Telecommunications Wire and<br>Cable       | -                               | -    |
| CISPR 11 (mod)        | 2009 | Industrial, scientific and medical equipment<br>- Radio-frequency disturbance<br>characteristics - Limits and methods of<br>measurement     | EN 55011                        | 2009 |
| CISPR 11:2009/A1      | 2010 |                                                                                                                                             | +prA<br>+AA<br>EN 55011:2009/A1 | 2010 |

## **Annex ZZ (informative)**

### **Coverage of Essential Requirements of EU Directives**

This European Standard has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association and within its scope the standard covers protection requirements of Annex I Article 1 of the EU Directive 2004/108/EC.

Compliance with this standard provides presumption of conformity with the specified essential requirements of the Directives concerned.

NOTE: Other requirements and other EU Directives may be applicable to the products falling within the scope of this standard.

## CONTENTS

|                                                                                         |    |
|-----------------------------------------------------------------------------------------|----|
| INTRODUCTION.....                                                                       | 12 |
| 1 Scope.....                                                                            | 13 |
| 2 Normative references.....                                                             | 13 |
| 3 Terms, definitions, symbols and abbreviated terms.....                                | 14 |
| 3.1 Terms and definitions .....                                                         | 14 |
| 3.2 Symbols and abbreviated terms .....                                                 | 20 |
| 4 Classification.....                                                                   | 20 |
| 4.1 General.....                                                                        | 20 |
| 4.2 DeviceNet communication model .....                                                 | 21 |
| 4.3 DeviceNet, CAN and CIP™ .....                                                       | 22 |
| 5 Characteristics .....                                                                 | 23 |
| 5.1 DeviceNet connections .....                                                         | 23 |
| 5.1.1 General .....                                                                     | 23 |
| 5.1.2 DeviceNet's use of the CAN identifier field .....                                 | 23 |
| 5.1.3 Connection establishment .....                                                    | 24 |
| 5.2 DeviceNet messaging protocol .....                                                  | 25 |
| 5.2.1 Explicit messaging .....                                                          | 25 |
| 5.2.2 Input/output messaging .....                                                      | 36 |
| 5.2.3 Fragmentation/reassembly .....                                                    | 37 |
| 5.2.4 Offline connection set .....                                                      | 41 |
| 5.2.5 Device heartbeat.....                                                             | 51 |
| 5.2.6 Device shutdown message .....                                                     | 52 |
| 5.2.7 Duplicate MAC ID detection protocol .....                                         | 54 |
| 5.2.8 Quick connect .....                                                               | 55 |
| 5.3 DeviceNet communication object classes.....                                         | 55 |
| 5.3.1 General .....                                                                     | 55 |
| 5.3.2 Identity object class definition (class ID code: 0x01) .....                      | 55 |
| 5.3.3 Message router object class definition (class ID code: 0x02).....                 | 56 |
| 5.3.4 DeviceNet object class definition (class ID code: 0x03).....                      | 56 |
| 5.3.5 Connection object class definition (class ID code: 0x05) .....                    | 56 |
| 5.3.6 Acknowledge handler object class definition (class ID code: 0x2B).....            | 57 |
| 5.4 Link access state machine.....                                                      | 57 |
| 5.4.1 General .....                                                                     | 57 |
| 5.4.2 State transition diagram and event matrix .....                                   | 57 |
| 5.4.3 Duplicate MAC ID detection.....                                                   | 60 |
| 5.5 Predefined master/slave connection set.....                                         | 60 |
| 5.5.1 General .....                                                                     | 60 |
| 5.5.2 Predefined master/slave connection set messages .....                             | 61 |
| 5.5.3 DeviceNet object class specific services for the master/slave connection set..... | 63 |
| 5.5.4 Slave connection object characteristics .....                                     | 70 |
| 5.5.5 Master connection object characteristics.....                                     | 75 |
| 5.5.6 Bit-strobe command/response messages .....                                        | 75 |
| 5.5.7 Poll command/response messages.....                                               | 76 |
| 5.5.8 Multicast poll command/response messages.....                                     | 77 |
| 5.5.9 Change of state/cyclic connections.....                                           | 77 |

|        |                                                                |     |
|--------|----------------------------------------------------------------|-----|
| 5.5.10 | Group 2 only devices .....                                     | 79  |
| 5.6    | CIP Safety™ on DeviceNet.....                                  | 80  |
| 5.6.1  | General .....                                                  | 80  |
| 5.6.2  | Use of CAN identifiers for CIP Safety on DeviceNet .....       | 80  |
| 5.7    | Physical layer .....                                           | 80  |
| 5.7.1  | General .....                                                  | 80  |
| 5.7.2  | Transceiver.....                                               | 82  |
| 5.7.3  | Grounding.....                                                 | 83  |
| 5.7.4  | Isolation.....                                                 | 83  |
| 5.7.5  | Transmission medium .....                                      | 85  |
| 5.7.6  | Topology.....                                                  | 85  |
| 5.7.7  | Link power .....                                               | 86  |
| 6      | Product information .....                                      | 87  |
| 7      | Normal service, mounting and transport conditions.....         | 87  |
| 7.1    | Normal service conditions .....                                | 87  |
| 7.1.1  | General .....                                                  | 87  |
| 7.1.2  | Ambient air temperature.....                                   | 87  |
| 7.1.3  | Altitude .....                                                 | 88  |
| 7.1.4  | Climatic conditions .....                                      | 88  |
| 7.2    | Conditions during transport and storage .....                  | 88  |
| 7.3    | Mounting.....                                                  | 88  |
| 8      | Constructional and performance requirements.....               | 88  |
| 8.1    | Indicators and configuration switches .....                    | 88  |
| 8.2    | DeviceNet cable.....                                           | 89  |
| 8.2.1  | Overview .....                                                 | 89  |
| 8.2.2  | Cable profile template .....                                   | 89  |
| 8.2.3  | Thick cable profile.....                                       | 90  |
| 8.2.4  | Thin cable profile .....                                       | 94  |
| 8.2.5  | Flat cable profile .....                                       | 97  |
| 8.3    | Terminating resistors.....                                     | 100 |
| 8.4    | Connectors .....                                               | 100 |
| 8.4.1  | General specifications.....                                    | 100 |
| 8.4.2  | Connector profile template .....                               | 100 |
| 8.4.3  | Open connector profile.....                                    | 101 |
| 8.4.4  | Sealed mini connector profile .....                            | 104 |
| 8.4.5  | Sealed micro connector profile .....                           | 105 |
| 8.4.6  | Flat trunk connector profile.....                              | 106 |
| 8.5    | Device taps and power taps.....                                | 109 |
| 8.5.1  | Device taps.....                                               | 109 |
| 8.5.2  | Power taps.....                                                | 110 |
| 8.6    | Link powered devices .....                                     | 111 |
| 8.7    | Miswiring protection .....                                     | 111 |
| 8.8    | Power supplies.....                                            | 111 |
| 8.9    | Electromagnetic compatibility (EMC).....                       | 112 |
| 8.9.1  | General .....                                                  | 112 |
| 8.9.2  | Immunity.....                                                  | 112 |
| 8.9.3  | Emissions .....                                                | 113 |
| 8.10   | Additional functional safety requirements related to EMC ..... | 114 |
| 9      | Tests.....                                                     | 114 |

|        |                                                                                              |     |
|--------|----------------------------------------------------------------------------------------------|-----|
| 9.1    | General.....                                                                                 | 114 |
| 9.2    | Electrical and EMC testing .....                                                             | 114 |
| 9.2.1  | Test of the DeviceNet power supply.....                                                      | 114 |
| 9.2.2  | Device peak current consumption .....                                                        | 115 |
| 9.2.3  | Power ON behaviour .....                                                                     | 115 |
| 9.2.4  | Reverse connection of V+ and V- .....                                                        | 116 |
| 9.2.5  | Disconnection of V- .....                                                                    | 117 |
| 9.2.6  | Differential input impedance test .....                                                      | 117 |
| 9.2.7  | Transmit levels .....                                                                        | 117 |
| 9.2.8  | Acknowledge delay .....                                                                      | 118 |
| 9.2.9  | CDI tests .....                                                                              | 119 |
| 9.2.10 | Electromagnetic compatibility testing .....                                                  | 120 |
| 9.3    | Logical testing.....                                                                         | 122 |
| 9.3.1  | General .....                                                                                | 122 |
| 9.3.2  | Duplicate MAC ID check test .....                                                            | 122 |
| 9.3.3  | UCMM .....                                                                                   | 123 |
| 9.3.4  | Allocation of predefined master/slave connection set – Explicit<br>messaging connection..... | 124 |
| 9.3.5  | Allocation of predefined master/slave connection set – I/O messaging<br>connection .....     | 124 |
| 9.3.6  | Logical testing of safety products .....                                                     | 125 |
|        | Bibliography .....                                                                           | 126 |
|        | Figure 1 – Typical DeviceNet controller-device interfaces .....                              | 21  |
|        | Figure 2 – DeviceNet protocol architecture compared with the OSI reference model.....        | 22  |
|        | Figure 3 – DeviceNet’s use of the CAN identifier field .....                                 | 23  |
|        | Figure 4 – Explicit message CAN data field use.....                                          | 25  |
|        | Figure 5 – Explicit message data field format.....                                           | 25  |
|        | Figure 6 – Explicit message header format .....                                              | 26  |
|        | Figure 7 – Service field format .....                                                        | 26  |
|        | Figure 8 – Open explicit messaging connection request format .....                           | 28  |
|        | Figure 9 – Open explicit messaging connection response format.....                           | 30  |
|        | Figure 10 – Close connection request format.....                                             | 32  |
|        | Figure 11 – Close connection response format .....                                           | 32  |
|        | Figure 12 – Non-fragmented explicit request message format, values 0 – 3.....                | 34  |
|        | Figure 13 – Non-fragmented explicit request message format, value 4 .....                    | 35  |
|        | Figure 14 – Non-fragmented success response message format .....                             | 35  |
|        | Figure 15 – Error response message.....                                                      | 36  |
|        | Figure 16 – Data field of an I/O message.....                                                | 36  |
|        | Figure 17 – Format of DeviceNet fragmentation protocol.....                                  | 37  |
|        | Figure 18 – I/O message fragment format.....                                                 | 38  |
|        | Figure 19 – Explicit message fragment format .....                                           | 38  |
|        | Figure 20 – Acknowledgement message format .....                                             | 40  |
|        | Figure 21 – Establishing the offline ownership .....                                         | 42  |
|        | Figure 22 – Multicast nature of the offline ownership .....                                  | 43  |
|        | Figure 23 – Offline ownership request message.....                                           |     |

|                                                                                                    |     |
|----------------------------------------------------------------------------------------------------|-----|
| Figure 24 – Offline ownership response message protocol.....                                       | 44  |
| Figure 25 – Communication faulted request message – Multicast protocol.....                        | 45  |
| Figure 26 – Communication faulted request message – Point-to-point protocol.....                   | 46  |
| Figure 27 – Identify communication faulted request message – Multicast protocol .....              | 47  |
| Figure 28 – Communication faulted identify response message.....                                   | 48  |
| Figure 29 – Identify communication faulted request message – Point-to-point protocol .....         | 48  |
| Figure 30 – Who communication faulted request message .....                                        | 49  |
| Figure 31 – Who response message.....                                                              | 50  |
| Figure 32 – Change MAC ID communication faulted request message .....                              | 50  |
| Figure 33 – Device heartbeat message.....                                                          | 51  |
| Figure 34 – Device shutdown message .....                                                          | 53  |
| Figure 35 – Duplicate MAC ID check CAN identifier field.....                                       | 54  |
| Figure 36 – Duplicate MAC ID check message data field format.....                                  | 54  |
| Figure 37 – Link access state transition diagram .....                                             | 58  |
| Figure 38 – Allocation choice byte contents .....                                                  | 63  |
| Figure 39 – Allocate_master/slave_connection_set request message.....                              | 64  |
| Figure 40 – Success response to allocate_master/slave_connection_set request .....                 | 65  |
| Figure 41 – Parent explicit messaging connection logic .....                                       | 67  |
| Figure 42 – Release choice byte contents .....                                                     | 68  |
| Figure 43 – Release_master/slave_connection set request message .....                              | 68  |
| Figure 44 – Success response to release_master/slave_connection_set request.....                   | 69  |
| Figure 45 – Predefined master/slave I/O connection state transition diagram .....                  | 71  |
| Figure 46 – Predefined master/slave explicit messaging connection state transition<br>diagram..... | 73  |
| Figure 47 – Physical layer block diagram.....                                                      | 81  |
| Figure 48 – Device containing a non-isolated physical layer.....                                   | 84  |
| Figure 49 – Device containing an isolated physical layer.....                                      | 85  |
| Figure 50 – DeviceNet medium topology .....                                                        | 86  |
| Figure 51 – Thick cable: physical configuration.....                                               | 93  |
| Figure 52 – Thick cable: current available on the DeviceNet power bus.....                         | 93  |
| Figure 53 – Thin cable: physical configuration .....                                               | 96  |
| Figure 54 – Thin cable: current available on the DeviceNet power bus .....                         | 96  |
| Figure 55 – Flat cable: physical configuration .....                                               | 99  |
| Figure 56 – Flat cable: current available on the DeviceNet power bus .....                         | 99  |
| Figure 57 – Open connector pinout .....                                                            | 103 |
| Figure 58 – Open connector geometry.....                                                           | 103 |
| Figure 59 – Mini connector pinout .....                                                            | 105 |
| Figure 60 – Micro connector pinout .....                                                           | 106 |
| Figure 61 – Flat trunk connector layout – Part 1 .....                                             | 108 |
| Figure 62 – Flat trunk connector layout – Part 2 .....                                             | 109 |
| Figure 63 – Power supply rise time test circuit.....                                               | 114 |
| Figure 64 – Current consumption test circuit.....                                                  | 115 |
| Figure 65 – Power ON test circuit.....                                                             | 116 |

|                                                                                                |     |
|------------------------------------------------------------------------------------------------|-----|
| Figure 66 – Test circuit for reversal of V+ and V– and also V– interruption .....              | 116 |
| Figure 67 – Differential impedance test circuit .....                                          | 117 |
| Figure 68 – Transmit level test setup .....                                                    | 118 |
| Figure 69 – Transmit levels .....                                                              | 118 |
| Figure 70 – Timing test setup .....                                                            | 119 |
| Figure 71 – Timing .....                                                                       | 119 |
| Figure 72 – CDI test configuration .....                                                       | 120 |
| Figure 73 – Test circuit for logical tests .....                                               | 122 |
| <br>                                                                                           |     |
| Table 1 – Message body format values .....                                                     | 29  |
| Table 2 – Group select values .....                                                            | 29  |
| Table 3 – Source message ID in open explicit messaging connection request .....                | 30  |
| Table 4 – Destination message ID in open explicit messaging connection response .....          | 31  |
| Table 5 – UCMM error conditions/codes .....                                                    | 33  |
| Table 6 – Fragment type bit values .....                                                       | 37  |
| Table 7 – Ack status bit values .....                                                          | 40  |
| Table 8 – Offline connection set .....                                                         | 42  |
| Table 9 – Addresses reporting based upon mask .....                                            | 46  |
| Table 10 – Device shutdown message shutdown code ranges .....                                  | 53  |
| Table 11 – Device shutdown message “Open” shutdown codes .....                                 | 54  |
| Table 12 – Link access state event matrix (1 of 2) .....                                       | 58  |
| Table 13 – Predefined master/slave connection set identifier fields .....                      | 62  |
| Table 14 – Allocate_master/slave_connection_set request service data field<br>parameters ..... | 63  |
| Table 15 – Allocate_master/slave_connection set response parameters .....                      | 64  |
| Table 16 – Release_master/slave_connection_set request service data field<br>parameters .....  | 68  |
| Table 17 – DeviceNet object specific additional error codes .....                              | 70  |
| Table 18 – Connection instance IDs for predefined master/slave connections .....               | 71  |
| Table 19 – Predefined master/slave I/O connection state event matrix (1 of 2) .....            | 72  |
| Table 20 – Predefined master/slave explicit messaging connection state event matrix .....      | 74  |
| Table 21 – Predefined master/slave I/O connection object attribute access .....                | 75  |
| Table 22 – General physical layer characteristics .....                                        | 81  |
| Table 23 – Transmitter characteristics .....                                                   | 82  |
| Table 24 – Receiver characteristics .....                                                      | 83  |
| Table 25 – Load limits .....                                                                   | 87  |
| Table 26 – Cable profile: data pair specifications .....                                       | 89  |
| Table 27 – Cable profile: DC power pair specifications .....                                   | 89  |
| Table 28 – Cable profile: general specifications .....                                         | 90  |
| Table 29 – Cable profile: topology .....                                                       | 90  |
| Table 30 – Thick cable: data pair specifications .....                                         | 91  |
| Table 31 – Thick cable: DC power pair specifications .....                                     | 91  |
| Table 32 – Thick cable: general specifications .....                                           | 92  |
| Table 33 – Thick cable: topology .....                                                         | 92  |

|                                                                                     |     |
|-------------------------------------------------------------------------------------|-----|
| Table 34 – Thick cable: maximum current available (A) based on network length ..... | 93  |
| Table 35 – Thin cable: data pair specifications .....                               | 94  |
| Table 36 – Thin cable: DC power pair specifications .....                           | 95  |
| Table 37 – Thin cable: general specifications .....                                 | 95  |
| Table 38 – Thin cable: topology .....                                               | 95  |
| Table 39 – Thin cable: maximum current available (A) based on network length .....  | 96  |
| Table 40 – Flat cable: data pair specifications .....                               | 97  |
| Table 41 – Flat cable: DC power pair specifications .....                           | 98  |
| Table 42 – Flat cable: general specifications .....                                 | 98  |
| Table 43 – Flat cable: topology .....                                               | 98  |
| Table 44 – Flat cable: maximum current available (A) based on network length .....  | 99  |
| Table 45 – Connector profile template .....                                         | 101 |
| Table 46 – Open connector .....                                                     | 102 |
| Table 47 – Sealed mini connector .....                                              | 104 |
| Table 48 – Sealed micro connector ( <i>1 of 2</i> ) .....                           | 105 |
| Table 49 – Flat trunk connector .....                                               | 107 |
| Table 50 – Internal pass-through conductor specifications .....                     | 110 |
| Table 51 – Internal drop conductor specifications .....                             | 110 |
| Table 52 – Internal pass-through conductor specifications .....                     | 110 |
| Table 53 – Internal power drop conductor specifications .....                       | 111 |
| Table 54 – Voltage regulator specifications .....                                   | 111 |
| Table 55 – DeviceNet power supply specifications .....                              | 112 |
| Table 56 – Immunity performance criteria .....                                      | 113 |

## INTRODUCTION

DeviceNet™<sup>1</sup> 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.

---

<sup>1</sup> 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.

# 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*

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 4566-942, *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.

---

<sup>2</sup> A newer version of this document exists (ASTM D4566-08e1), however the listed revision applies for this standard

### 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).

[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

### 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

### 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

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

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,<br>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.

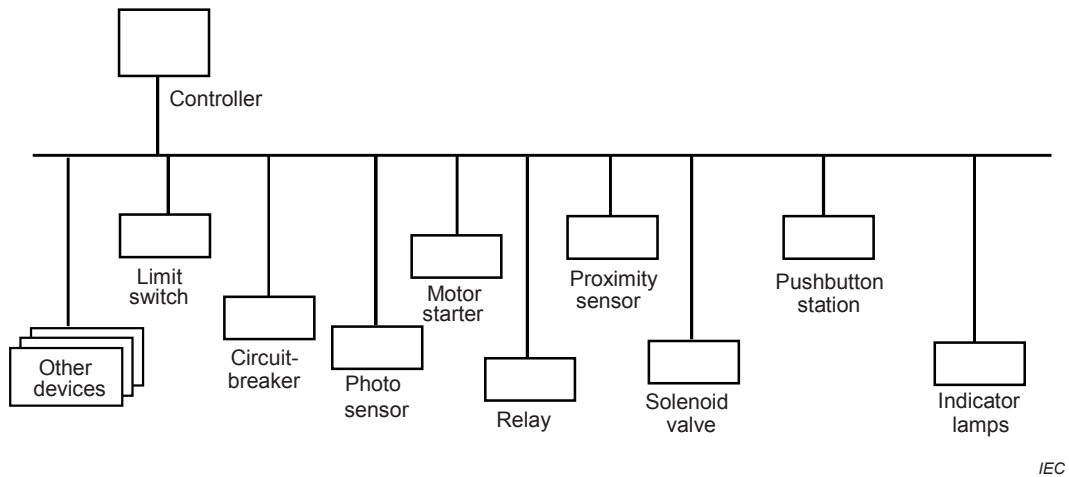


**Figure 1 – Typical DeviceNet controller-device interfaces**

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:

- **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™<sup>3</sup>) 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.

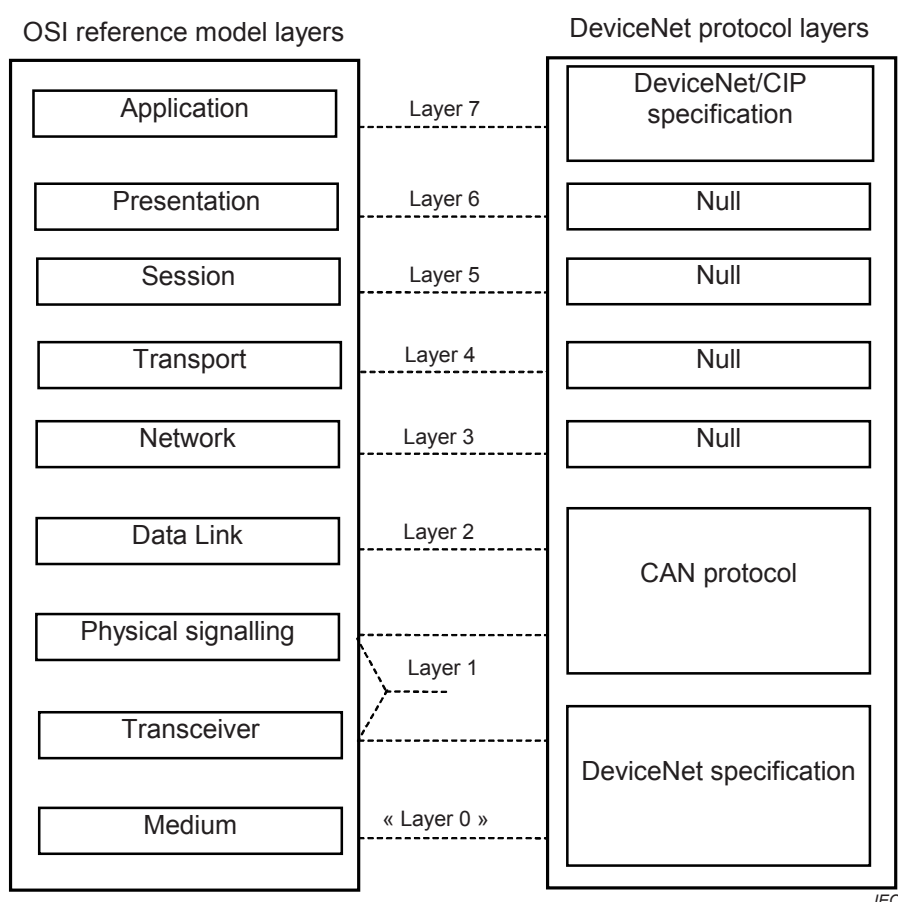


Figure 2 – DeviceNet protocol architecture compared with the OSI reference model

<sup>3</sup> 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.

## 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.

| IDENTIFIER BITS |                    |                    |   |   |                               |   |                    |   |   |   | HEX RANGE     | IDENTITY USAGE  |                         |
|-----------------|--------------------|--------------------|---|---|-------------------------------|---|--------------------|---|---|---|---------------|-----------------|-------------------------|
| 10              | 9                  | 8                  | 7 | 6 | 5                             | 4 | 3                  | 2 | 1 | 0 |               |                 |                         |
| 0               | Group 1 message ID |                    |   |   | Source MAC ID                 |   |                    |   |   |   | 0x000 – 0x3FF | Message group 1 |                         |
| 1               | 0                  | MAC ID             |   |   |                               |   | Group 2 message ID |   |   |   | 0x400 – 0x5FF | Message group 2 |                         |
| 1               | 1                  | Group 3 message ID |   |   | Source MAC ID                 |   |                    |   |   |   | 0x600 – 0x7BF | Message group 3 |                         |
| 1               | 1                  | 1                  | 1 | 1 | Group 4 message ID (0 – 0x2F) |   |                    |   |   |   | 0x7C0 – 0x7EF | Message group 4 |                         |
| 1               | 1                  | 1                  | 1 | 1 | 1                             | 1 | 1                  | X | X | X | X             | 0x7F0 – 0x7FF   | Invalid CAN identifiers |
| 10              | 9                  | 8                  | 7 | 6 | 5                             | 4 | 3                  | 2 | 1 | 0 |               |                 |                         |

IEC

**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.

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.

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).

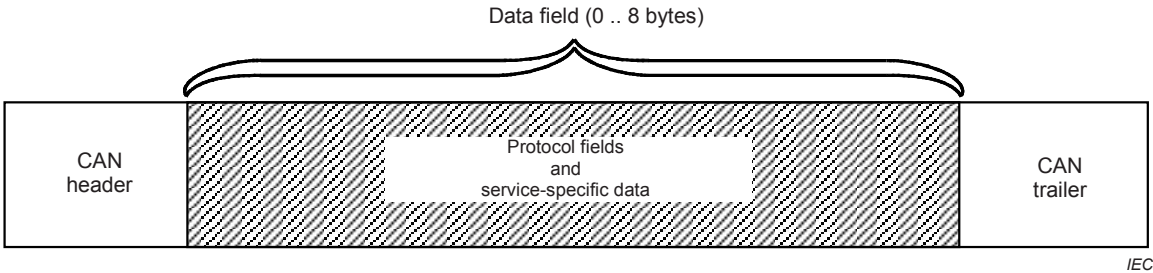


Figure 4 – Explicit message CAN data field use

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

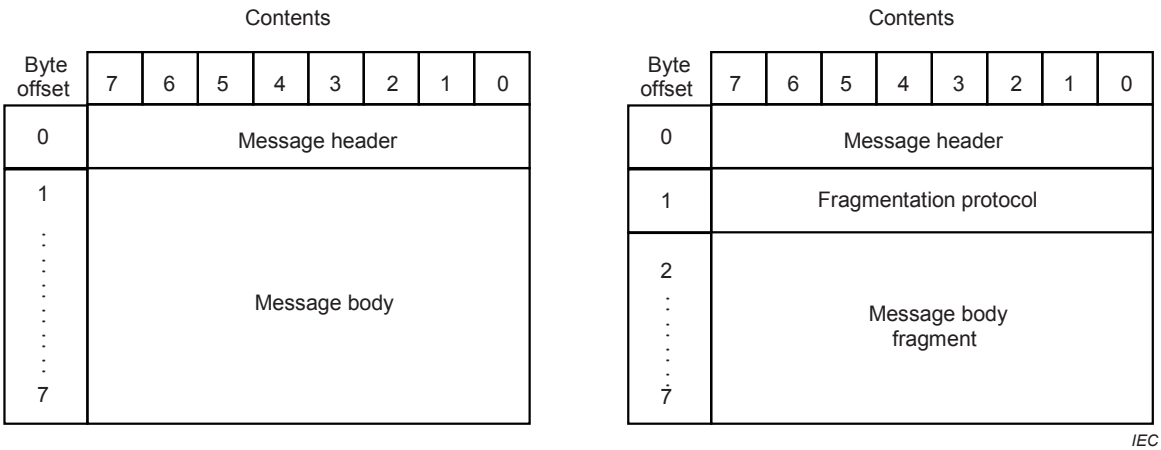


Figure 5 – Explicit message data field format

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:

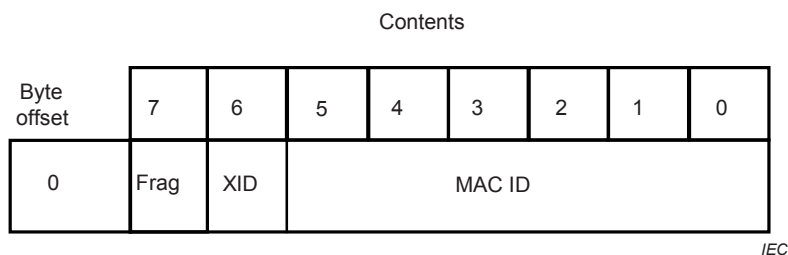


**Figure 6 – Explicit message header format**

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.

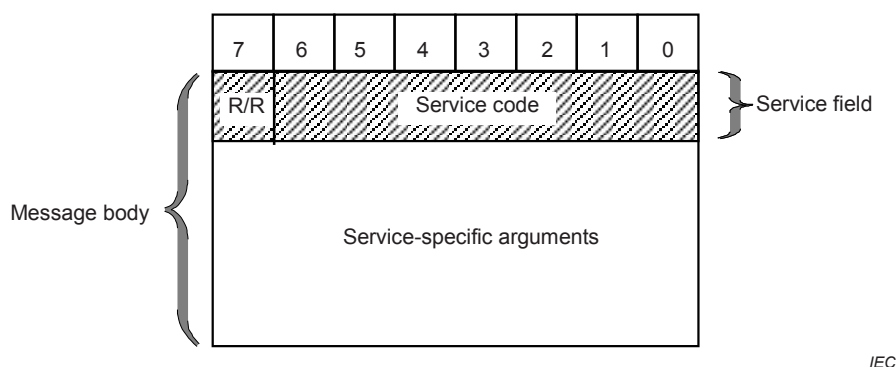


**Figure 7 – Service field format**

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;

- **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).

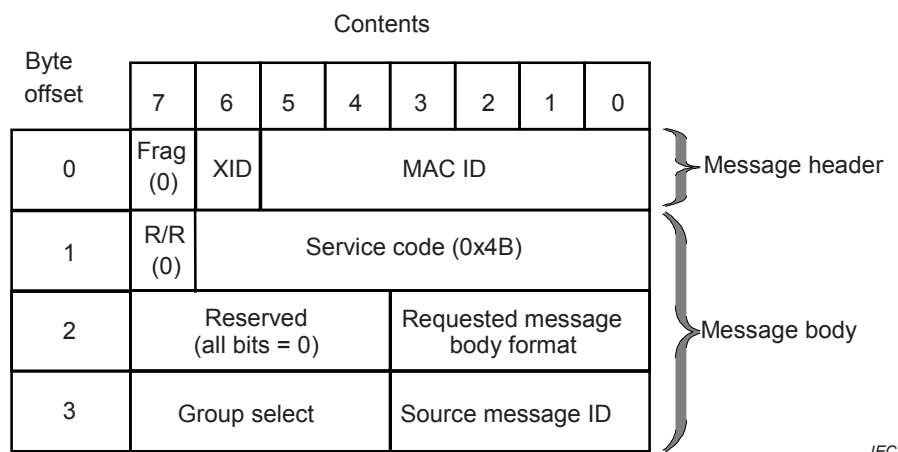


**Figure 8 – Open explicit messaging connection request format**

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;

**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 <sup>a</sup> |
| 5 – 0xF                                                                                                                                                                                                                                                                                                                                  | Reserved                                                                                                 |
| NOTE Messages transmitted across this connection are formatted as described in 5.2.1.6.                                                                                                                                                                                                                                                  |                                                                                                          |
| <sup>a</sup> 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 <sup>a</sup>        |
| 2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Reserved                            |
| 3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Message group 3                     |
| 4 – 0xE                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Reserved                            |
| 0xF                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Reserved for node ping <sup>b</sup> |
| <sup>a</sup> 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. |                                     |
| <sup>b</sup> 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):

**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 <sup>a</sup> |
| 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Is ignored/set to the value zero (0) <sup>b</sup>                                                                                                                                                                                                                        |
| <sup>a</sup> The client places this value within the message ID component of the message group 1 or 3 Identifier.<br><sup>b</sup> 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).

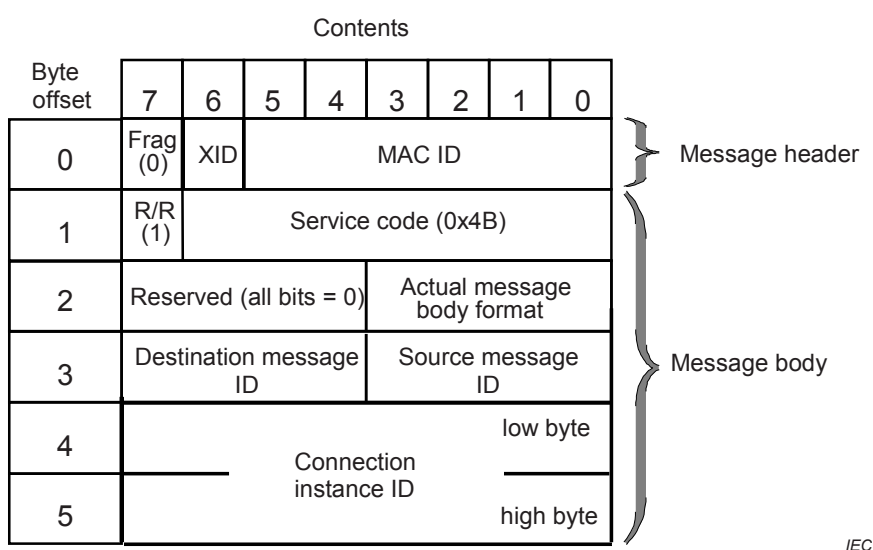


**Figure 9 – Open explicit messaging connection response format**

Open explicit messaging connection response contents:

- **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.

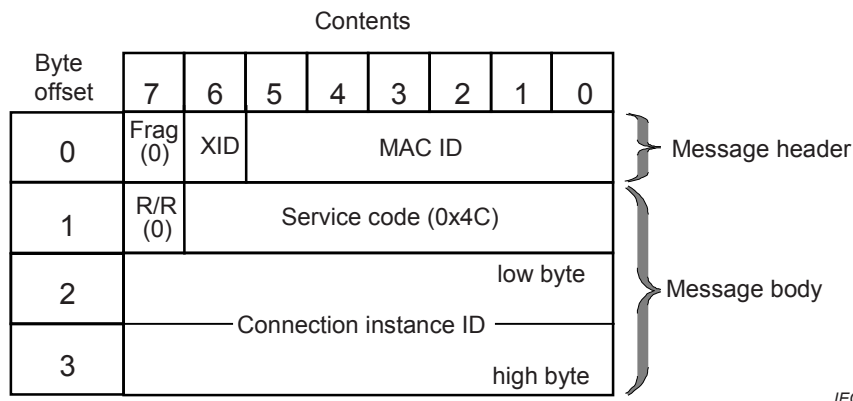


**Figure 10 – Close connection request format**

#### 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.

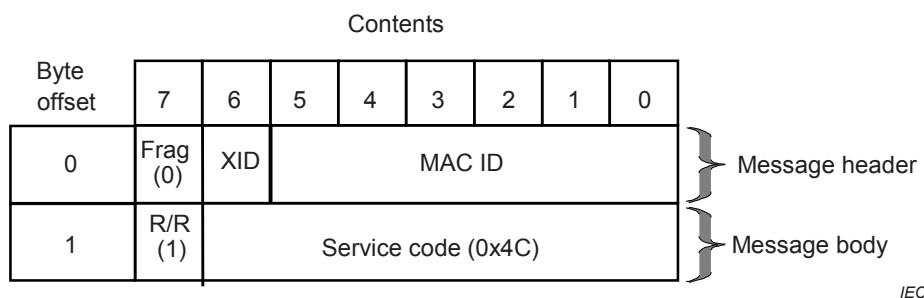


**Figure 11 – Close connection response format**

#### 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.

**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:

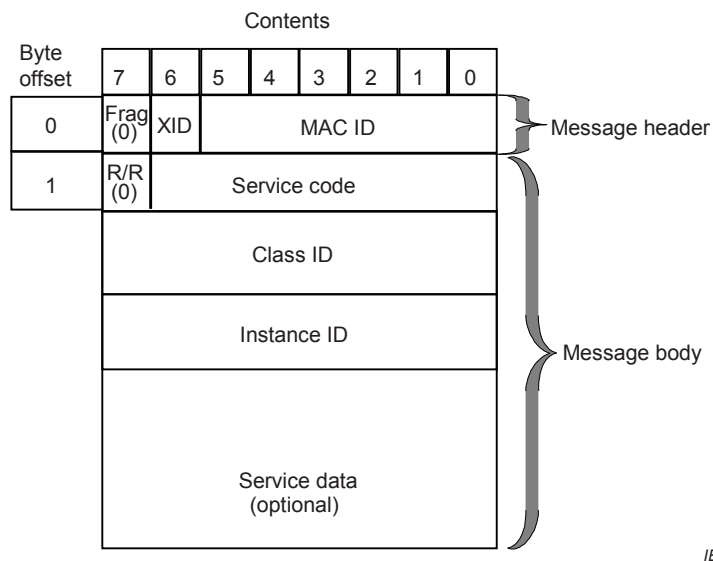


**Figure 12 – Non-fragmented explicit request message format, values 0 – 3**

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:

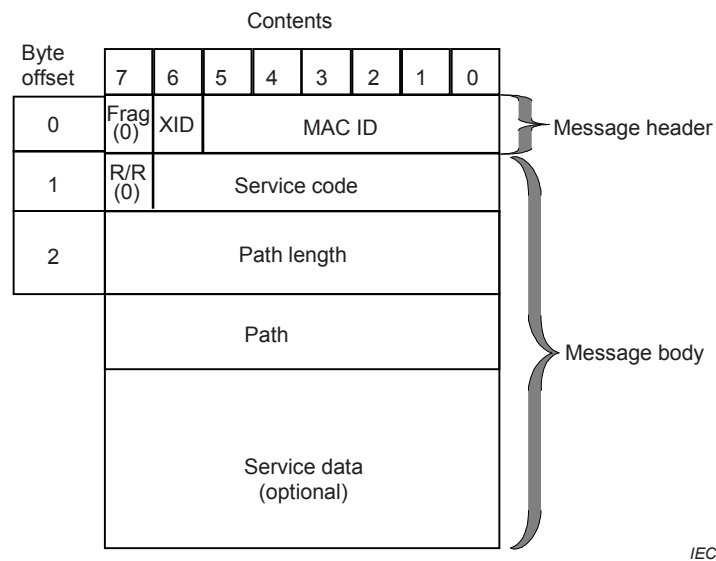


**Figure 13 – Non-fragmented explicit request message format, value 4**

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:

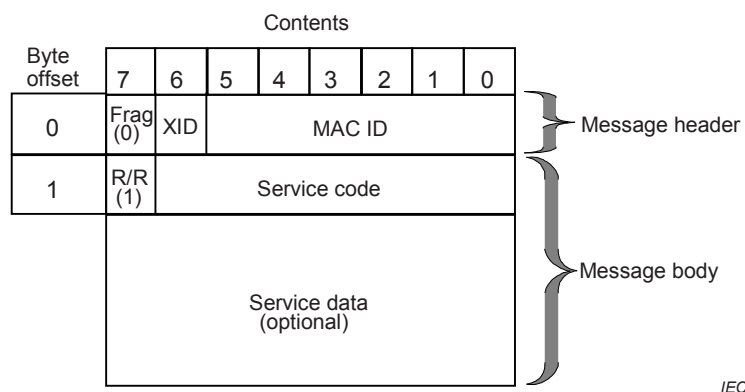


**Figure 14 – Non-fragmented success response message format**

Non-fragmented success response message contents:

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

- **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.

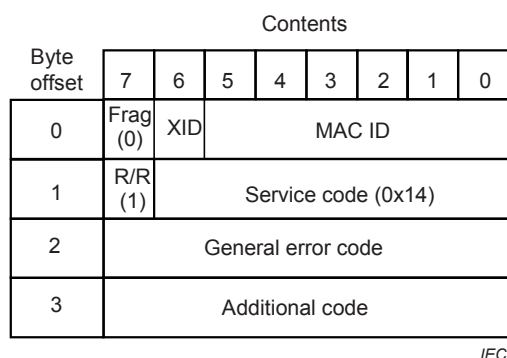


**Figure 15 – Error response message**

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).

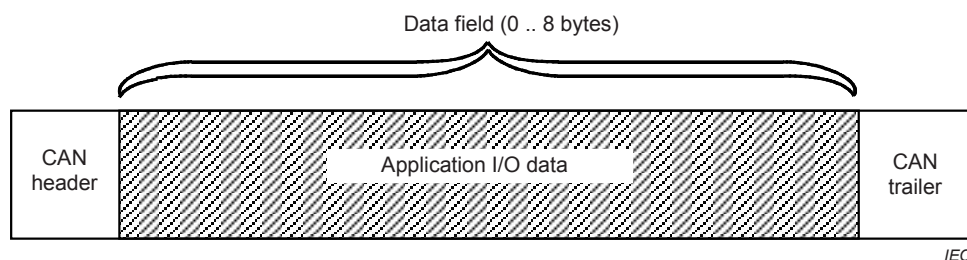


**Figure 16 – Data field of an I/O message**

### 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:

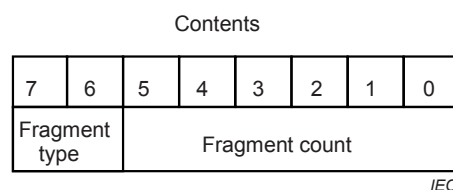


**Figure 17 – Format of DeviceNet fragmentation protocol**

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 <sup>a</sup> |
| 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Middle fragment <sup>b</sup>                                                            |
| 2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Last fragment <sup>c</sup>                                                              |
| 3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Fragment acknowledge <sup>d</sup>                                                       |
| <sup>a</sup> 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.<br><sup>b</sup> This fragment is neither the first nor the last fragment in the series.<br><sup>c</sup> Marks this as the last fragment.<br><sup>d</sup> The value the receiver of a fragmented message uses to acknowledge the reception of a fragment. |                                                                                         |