

**NORME
INTERNATIONALE
INTERNATIONAL
STANDARD**

**CEI
IEC**

62026-3

Première édition
First edition
2000-07

**Appareillage à basse tension –
Interfaces appareil de commande-appareil (CDI) –**

**Partie 3:
DeviceNet**

**Low-voltage switchgear and controlgear –
Controller-device interfaces (CDIs) –**

**Part 3:
DeviceNet**



Numéro de référence
Reference number
CEI/IEC 62026-3:2000

Numéros des publications

Depuis le 1er janvier 1997, les publications de la CEI sont numérotées à partir de 60000.

Publications consolidées

Les versions consolidées de certaines publications de la CEI incorporant les amendements sont disponibles. Par exemple, les numéros d'édition 1.0, 1.1 et 1.2 indiquent respectivement la publication de base, la publication de base incorporant l'amendement 1, et la publication de base incorporant les amendements 1 et 2.

Validité de la présente publication

Le contenu technique des publications de la CEI est constamment revu par la CEI afin qu'il reflète l'état actuel de la technique.

Des renseignements relatifs à la date de reconfirmation de la publication sont disponibles dans le Catalogue de la CEI.

Les renseignements relatifs à des questions à l'étude et des travaux en cours entrepris par le comité technique qui a établi cette publication, ainsi que la liste des publications établies, se trouvent dans les documents ci-dessous:

- **«Site web» de la CEI***
- **Catalogue des publications de la CEI**
Publié annuellement et mis à jour régulièrement
(Catalogue en ligne)*
- **Bulletin de la CEI**
Disponible à la fois au «site web» de la CEI* et comme périodique imprimé

Terminologie, symboles graphiques et littéraux

En ce qui concerne la terminologie générale, le lecteur se reportera à la CEI 60050: *Vocabulaire Electrotechnique International* (VEI).

Pour les symboles graphiques, les symboles littéraux et les signes d'usage général approuvés par la CEI, le lecteur consultera la CEI 60027: *Symboles littéraux à utiliser en électrotechnique*, la CEI 60417: *Symboles graphiques utilisables sur le matériel. Index, relevé et compilation des feuilles individuelles*, et la CEI 60617: *Symboles graphiques pour schémas*.

* Voir adresse «site web» sur la page de titre.

Numbering

As from 1 January 1997 all IEC publications are issued with a designation in the 60000 series.

Consolidated publications

Consolidated versions of some IEC publications including amendments are available. For example, edition numbers 1.0, 1.1 and 1.2 refer, respectively, to the base publication, the base publication incorporating amendment 1 and the base publication incorporating amendments 1 and 2.

Validity of this publication

The technical content of IEC publications is kept under constant review by the IEC, thus ensuring that the content reflects current technology.

Information relating to the date of the reconfirmation of the publication is available in the IEC catalogue.

Information on the subjects under consideration and work in progress undertaken by the technical committee which has prepared this publication, as well as the list of publications issued, is to be found at the following IEC sources:

- **IEC web site***
- **Catalogue of IEC publications**
Published yearly with regular updates
(On-line catalogue)*
- **IEC Bulletin**
Available both at the IEC web site* and as a printed periodical

Terminology, graphical and letter symbols

For general terminology, readers are referred to IEC 60050: *International Electrotechnical Vocabulary* (IEV).

For graphical symbols, and letter symbols and signs approved by the IEC for general use, readers are referred to publications IEC 60027: *Letter symbols to be used in electrical technology*, IEC 60417: *Graphical symbols for use on equipment. Index, survey and compilation of the single sheets* and IEC 60617: *Graphical symbols for diagrams*.

* See web site address on title page.

**NORME
INTERNATIONALE
INTERNATIONAL
STANDARD**

**CEI
IEC**

62026-3

Première édition
First edition
2000-07

**Appareillage à basse tension –
Interfaces appareil de commande-appareil (CDI) –**

**Partie 3:
DeviceNet**

**Low-voltage switchgear and controlgear –
Controller-device interfaces (CDIs) –**

**Part 3:
DeviceNet**

© IEC 2000 Droits de reproduction réservés — Copyright - all rights reserved

Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'éditeur.

No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

International Electrotechnical Commission
Telefax: +41 22 919 0300

3, rue de Varembé Geneva, Switzerland
e-mail: inmail@iec.ch IEC web site <http://www.iec.ch>



Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

CODE PRIX
PRICE CODE **XG**

Pour prix, voir catalogue en vigueur
For price, see current catalogue

SOMMAIRE

	Pages
AVANT-PROPOS	12
INTRODUCTION	14
 Articles	
1 Domaine d'application	16
2 Références normatives	16
3 Définitions et abréviations	18
4 Classifications	28
4.1 Généralités	28
4.2 Modèle de communication DeviceNet	30
4.3 DeviceNet et CAN	32
5 Caractéristiques	32
5.1 Connexions DeviceNet	32
5.2 Protocole de messagerie DeviceNet	36
5.3 Classes d'objets de communication DeviceNet	68
5.4 Machine d'état d'accès à la liaison	124
5.5 Jeu de connexions maître/esclave prédéfini	130
5.6 Couche physique	150
6 Information sur le matériel	162
7 Conditions normales de service, de montage et de transport	162
7.1 Conditions normales de service	162
7.2 Conditions durant le transport et le stockage	164
7.3 Montage	164
8 Dispositions relatives à la construction et au fonctionnement	164
8.1 Voyants et commutateurs mécaniques de configuration	164
8.2 Câble DeviceNet	170
8.3 Résistances de terminaison	174
8.4 Connecteurs	174
8.5 Dérivateurs et raccords d'alimentation	176
8.6 Appareils alimentés par la liaison	180
8.7 Protection contre les erreurs de câblage	180
8.8 Blocs d'alimentation	182
8.9 Compatibilité électromagnétique (CEM)	182
9 Essais	186
9.1 Généralités	186
9.2 Essais électriques et de compatibilité électromagnétique (CEM)	188
9.3 Essais logiques	204
 Annexe A (normative) Services communs	 212
Annexe B (normative) Codes d'erreurs DeviceNet	226
Annexe C (normative) Définition de l'attribut chemin de connexion	228
Annexe D (normative) Spécifications et codage du type de données	234
Annexe E (normative) Bibliothèque des objets de communication	242
Annexe F (normative) Plages de valeurs	282

CONTENTS

	Page
FOREWORD	13
INTRODUCTION	15
Clause	
1 Scope	17
2 Normative references	17
3 Definitions and abbreviations	19
4 Classification	29
4.1 General	29
4.2 DeviceNet communication model	31
4.3 DeviceNet and CAN	33
5 Characteristics	33
5.1 DeviceNet connections	33
5.2 DeviceNet messaging protocol	37
5.3 DeviceNet communication object classes	69
5.4 Link access state machine	125
5.5 Predefined master/slave connection set	131
5.6 Physical layer	151
6 Product information	163
7 Normal service, mounting and transport conditions	163
7.1 Normal service conditions	163
7.2 Conditions during transport and storage	165
7.3 Mounting	165
8 Constructional and performance requirements	165
8.1 Indicators and configuration switches	165
8.2 DeviceNet cable	171
8.3 Terminating resistors	175
8.4 Connectors	175
8.5 Device taps and power taps	177
8.6 Link powered devices	181
8.7 Miswiring protection	181
8.8 Power supplies	183
8.9 Electromagnetic compatibility (EMC)	183
9 Tests	187
9.1 General	187
9.2 Electrical and EMC testing	189
9.3 Logical testing	205
Annex A (normative) Common services	213
Annex B (normative) DeviceNet error codes	227
Annex C (normative) Connection path attribute definition	229
Annex D (normative) Data type specification and encoding	235
Annex E (normative) Communication objects library	243
Annex F (normative) Value ranges	283

Figure 1 – Interfaces appareil de commande-appareil DeviceNet types.....	28
Figure 2 – Comparaison entre l'architecture du protocole DeviceNet et le modèle de référence OSI	32
Figure 3 – Utilisation du champ identificateur de CAN par DeviceNet.....	34
Figure 4 – Utilisation du champ de données CAN dans un message explicite	36
Figure 5 – Format du champ de données des messages explicites	38
Figure 6 – Format de l'en-tête d'un message explicite.....	38
Figure 7 – Format du champ de service	40
Figure 8 – Format de demande d'ouverture de connexion de messagerie explicite.....	42
Figure 9 – Format de la réponse d'ouverture de connexion de messagerie explicite.....	48
Figure 10 – Format de la demande de fermeture de connexion	50
Figure 11 – Format de la réponse de fermeture de connexion.....	50
Figure 12 – Format d'un message de demande explicite non fragmenté.....	54
Figure 13 – Format d'un message réponse positif non fragmenté.....	54
Figure 14 – Message réponse d'erreur.....	56
Figure 15 – Champ de données d'un message d'E/S.....	56
Figure 16 – Format du protocole de fragmentation DeviceNet.....	58
Figure 17 – Format d'un fragment de message d'E/S.....	60
Figure 18 – Format d'un fragment de message explicite.....	60
Figure 19 – Format d'un message d'acquiescement	64
Figure 20 – Champ identificateur de CAN pour la vérification de duplication d'adresses MAC ID.....	66
Figure 21 – Format du champ de données d'un message de vérification de duplication d'adresses MAC ID	68
Figure 22 – Transportclass_trigger	74
Figure 23 – Format de l'attribut initial_comm_characteristics	76
Figure 24 – Diagramme de la machine d'état d'un objet connexion d'E/S	90
Figure 25 – Diagramme de la machine d'état d'un objet connexion de messagerie explicite ..	94
Figure 26 – Contenu de l'octet de choix d'attribution	110
Figure 27 – Message de demande allocate_master/slave_connection_set	112
Figure 28 – Réponse positive à une demande allocate_master/slave_connection_set.....	114
Figure 29 – Choix d'une connexion de messagerie explicite parent.....	118
Figure 30 – Contenu de l'octet de choix de libération	120
Figure 31 – Message de demande release_master/slave_connection_set.....	120
Figure 32 – Réponse positive à une demande release_master/slave_connection_set	122
Figure 33 – Diagramme de la machine d'état d'accès à la liaison	126
Figure 34 – Diagramme de la machine d'état des connexions d'E/S maître/esclave prédéfinies.....	136
Figure 35 – Diagramme de la machine d'état pour la messagerie explicite maître/esclave prédéfinie	140
Figure 36 – Schéma fonctionnel de la couche physique	152
Figure 37 – Appareil contenant une couche physique non isolée	156
Figure 38 – Appareil contenant une couche physique isolée	158
Figure 39 – Configuration physique des câbles épais et fins	170
Figure 40 – Circuit d'essai du temps de montée du bloc d'alimentation	188
Figure 41 – Circuit d'essai de consommation de courant.....	190

Figure 1 – Typical DeviceNet controller-device interfaces	29
Figure 2 – DeviceNet protocol architecture compared with the OSI reference model	33
Figure 3 – DeviceNet's use of the CAN identifier field	35
Figure 4 – Explicit message CAN data field use	37
Figure 5 – Explicit message data field format	39
Figure 6 – Explicit message header format	39
Figure 7 – Service field format	41
Figure 8 – Open explicit messaging connection request format	43
Figure 9 – Open explicit messaging connection response format	49
Figure 10 – Close connection request format	51
Figure 11 – Close connection response format	51
Figure 12 – Non-fragmented explicit request message format	55
Figure 13 – Non-fragmented success response message format	55
Figure 14 – Error response message	57
Figure 15 – Data field of an I/O message	57
Figure 16 – Format of DeviceNet fragmentation protocol	59
Figure 17 – I/O message fragment format	61
Figure 18 – Explicit message fragment format	61
Figure 19 – Acknowledgement message format	65
Figure 20 – Duplicate MAC ID check CAN identifier field	67
Figure 21 – Duplicate MAC ID check message data field format	69
Figure 22 – Transportclass_trigger	75
Figure 23 – Initial_comm_characteristics attribute format	77
Figure 24 – I/O connection object state transition diagram	91
Figure 25 – Explicit messaging connection object state transition diagram	95
Figure 26 – Allocation choice byte contents	111
Figure 27 – Allocate_master/slave_connection_set request message	113
Figure 28 – Success response to allocate_master/slave_connection_set request	115
Figure 29 – Parent explicit messaging connection logic	119
Figure 30 – Release choice byte contents	121
Figure 31 – Release_master/slave_connection set request message	121
Figure 32 – Success response to release_master/slave_connection_set request	123
Figure 33 – Link access state transition diagram	127
Figure 34 – Predefined master/slave I/O connection state transition diagram	137
Figure 35 – Predefined master/slave explicit messaging state transition diagram	141
Figure 36 – Physical layer block diagram	153
Figure 37 – Device containing a non-isolated physical layer	157
Figure 38 – Device containing an isolated physical layer	159
Figure 39 – Physical configuration of thick and thin cable	171
Figure 40 – Power supply rise time test circuit	189
Figure 41 – Current consumption test circuit	191

Figure 42 – Circuit d'essai de mise sous tension.....	190
Figure 43 – Circuit d'essai pour l'inversion de $U+$ et $U-$, et aussi pour la coupure de $U-$	192
Figure 44 – Circuit d'essai d'impédance différentielle.....	194
Figure 45 – Installation pour l'essai des niveaux de transmission.....	196
Figure 46 – Niveaux de transmission	196
Figure 47 – Configuration de l'essai de temporisation	198
Figure 48 – Temporisation	198
Figure 49 – Configuration de l'essai CDI.....	200
Figure 50 – Circuit d'essai pour les essais logiques	204
Figure C.1 – Type/format du segment	228
Figure C.2 – Segment logique.....	230
Figure C.3 – Segment symbolique	230
Figure C.4 – Segment de données.....	232
Figure E.1 – Diagramme de la machine d'état de l'objet identité.....	252
Figure E.2 – Relations types entre temporisateurs pour la production de données acquittée.....	272
Figure E.3 – Chronogramme type pour le mode changement d'état.....	272
Figure E.4 – Flux de messages – Un objet connexion, un consommateur	274
Figure E.5 – Flux de message – Configuration à consommateurs multiples.....	274
Tableau 1 – Valeurs de format de corps de message.....	44
Tableau 2 – Valeurs de sélection des groupes.....	44
Tableau 3 – ID de message source dans une demande d'ouverture de connexion de messagerie explicite	46
Tableau 4 – ID de message de destination dans une réponse d'ouverture de connexion de messagerie explicite	48
Tableau 5 – Conditions/codes d'erreurs UCMM.....	52
Tableau 6 – Valeurs des bits du type de fragment.....	58
Tableau 7 – Valeurs du bit d'état d'acquittement	64
Tableau 8 – Attribut de la classe de connexion	70
Tableau 9 – Services de la classe de connexion	70
Tableau 10 – Attributs d'instances d'un objet connexion	72
Tableau 11 – Valeurs associées à l'attribut state (état)	74
Tableau 12 – Valeurs associées à l'attribut instance_type.....	74
Tableau 13 – Valeurs définies pour l'attribut produced_connection_id.....	76
Tableau 14 – Valeurs définies pour l'attribut consumed_connection_id	76
Tableau 15 – Valeurs du demi-octet des caractéristiques initiales de production.....	78
Tableau 16 – Valeurs du demi-octet des caractéristiques initiales de consommation.....	78
Tableau 17 – Valeurs pour watchdog_timeout_action.....	82
Tableau 18 – Services d'instance des objets connexion	88
Tableau 19 – Données de service pour la réponse apply_attributes de l'objet connexion	88
Tableau 20 – Table états-événements d'une connexion d'E/S	90
Tableau 21 – Table états-événements d'une connexion de messagerie explicite.....	96
Tableau 22 – Accès aux attributs d'un objet connexion d'E/S	98
Tableau 23 – Accès aux attributs d'un objet connexion de messagerie explicite.....	100
Tableau 24 – Attributs de classe pour l'objet DeviceNet.....	102

Figure 42 – Power ON test circuit	191
Figure 43 – Test circuit for reversal of U^+ and U^- , and also U^- interruption	193
Figure 44 – Differential impedance test circuit	195
Figure 45 – Transmit level test set-up	197
Figure 46 – Transmit levels	197
Figure 47 – Timing test set-up	199
Figure 48 – Timing	199
Figure 49 – CDI test configuration	201
Figure 50 – Test circuit for logical tests	205
Figure C.1 – Segment type/format	229
Figure C.2 – Logical segment	231
Figure C.3 – Symbolic segment	231
Figure C.4 – Data segment	233
Figure E.1 – State transition diagram for identity object	253
Figure E.2 – Typical timing relationships for acknowledged data production	273
Figure E.3 – Typical timing diagram for a change of state system	273
Figure E.4 – Message flow – One connection object, one consumer	275
Figure E.5 – Message flow – Multi-consumer configuration	275
Table 1 – Message body format values	45
Table 2 – Group select values	45
Table 3 – Source message ID in open explicit messaging connection request	47
Table 4 – Destination message ID in an open explicit messaging connection response	49
Table 5 – UCMM error conditions/codes	53
Table 6 – Fragment type bit values	59
Table 7 – Ack status bit values	65
Table 8 – Connection class attribute	71
Table 9 – Connection class services	71
Table 10 – Connection object instance attributes	73
Table 11 – Values assigned to the state attribute	75
Table 12 – Values assigned to the instance_type attribute	75
Table 13 – Values defined for the produced_connection_id attribute	77
Table 14 – Values defined for the consumed_connection_id attribute	77
Table 15 – Values for the initial production characteristics nibble	79
Table 16 – Values for the initial consumption characteristics nibble	79
Table 17 – Values for the watchdog_timeout_action	83
Table 18 – Connection object instance services	89
Table 19 – Service data for connection object apply_attributes response	89
Table 20 – I/O connection state-event matrix	91
Table 21 – Explicit messaging connection state-event matrix	97
Table 22 – I/O connection object attribute access	99
Table 23 – Explicit messaging connection object attribute access	101
Table 24 – Class attributes for the DeviceNet object	103

Tableau 25 – Services communs de la classe d'objet DeviceNet.....	102
Tableau 26 – Attributs d'instance pour l'objet DeviceNet.....	102
Tableau 27 – Valeurs de l'attribut vitesse de transmission	104
Tableau 28 – Valeurs de l'attribut BOI (interruption bus-off).....	106
Tableau 29 – Services communs de l'instance d'objet DeviceNet.....	108
Tableau 30 – Services spécifiques à une classe de l'objet DeviceNet	108
Tableau 31 – Paramètres du champ des données de service d'une demande allocate_master/slave_connection_set.....	110
Tableau 32 – Paramètres de la réponse allocate_master/slave_connection_set.....	112
Tableau 33 – Paramètres du champ des données de service d'une demande release_master/slave_connection_set.....	120
Tableau 34 – Codes d'erreurs supplémentaires spécifiques à l'objet DeviceNet.....	124
Tableau 35 – Table états-événements d'accès à la liaison.....	128
Tableau 36 – Champs identificateurs pour le jeu de connexions maître/esclave prédéfini ...	132
Tableau 37 – ID d'instance de connexion des connexions maître/esclave prédéfinies	134
Tableau 38 – Table états-événements des connexions d'E/S maître/esclave prédéfinies	136
Tableau 39 – Table états-événements des connexions d'E/S maître/esclave prédéfinies	140
Tableau 40 – Accès aux attributs des objets connexions d'E/S maître/esclave prédéfinis....	142
Tableau 41 – Caractéristiques générales de la couche physique.....	152
Tableau 42 – Caractéristiques du transmetteur.....	154
Tableau 43 – Caractéristiques du récepteur.....	154
Tableau 44 – Longueur totale de câble des lignes de dérivation	160
Tableau 45 – Limites de charge	160
Tableau 46 – Voyant d'état du module	166
Tableau 47 – Voyant d'état du CDI.....	166
Tableau 48 – Voyant d'état combiné module/CDI	168
Tableau 49 – Voyant d'état des E/S.....	168
Tableau 50 – Codage des commutateurs de vitesse de transmission pour DeviceNet	170
Tableau 51 – Marquage des voyants et commutateurs.....	170
Tableau 52 – Câble: spécifications de la paire des données	172
Tableau 53 – Câble: spécification de la prise d'alimentation à courant continu.....	172
Tableau 54 – Câble: spécifications générales	172
Tableau 55 – Spécifications générales des connecteurs ouverts.....	174
Tableau 56 – Spécifications générales des connecteurs scellés.....	174
Tableau 57 – Spécifications sur les contacts.....	176
Tableau 58 – Spécifications électriques	176
Tableau 59 – Disposition des contacts d'un connecteur	176
Tableau 60 – Spécifications des conducteurs principaux internes	178
Tableau 61 – Spécifications des conducteurs de dérivation internes	178
Tableau 62 – Spécifications des conducteurs principaux internes	178
Tableau 63 – Spécifications des conducteurs de dérivation internes	180
Tableau 64 – Spécifications du régulateur de tension	180
Tableau 65 – Spécifications des blocs d'alimentation DeviceNet.....	182
Tableau 66 – Critères de comportement pour les essais d'immunité	184

Table 25 – DeviceNet object class common services	103
Table 26 – Instance attributes for the DeviceNet object	103
Table 27 – Bit rate attribute values	105
Table 28 – BOI attribute values	107
Table 29 – DeviceNet object instance common services	109
Table 30 – DeviceNet object class-specific services	109
Table 31 – Allocate_master/slave_connection_set request service data field parameters	111
Table 32 – Allocate_master/slave_connection_set response parameters	113
Table 33 – Release_master/slave_connection_set request service data field parameters ...	121
Table 34 – DeviceNet object-specific additional error codes	125
Table 35 – Link access state-event matrix	129
Table 36 – Predefined master/slave connection set identifier fields	133
Table 37 – Connection instance IDs for predefined master/slave connections	135
Table 38 – Predefined master/slave I/O connection state-event matrix	137
Table 39 – Predefined master/slave I/O connection state-event matrix	141
Table 40 – Predefined master/slave I/O connection object attribute access	143
Table 41 – General physical layer characteristics	153
Table 42 – Transmitter characteristics	155
Table 43 – Receiver characteristics	155
Table 44 – Total drop line cable length	161
Table 45 – Load limits	161
Table 46 – Module status indicator	167
Table 47 – CDI status indicator	167
Table 48 – Combined module/CDI status indicator	169
Table 49 – I/O status indicator	169
Table 50 – DeviceNet bit rate switch encoding	171
Table 51 – Indicator and switch marking	171
Table 52 – Cable data pair specification	173
Table 53 – Cable d.c. power pair specification	173
Table 54 – Cable general specifications	173
Table 55 – Open connector general specifications	175
Table 56 – Sealed connector general specifications	175
Table 57 – Contact specifications	177
Table 58 – Electrical specifications	177
Table 59 – Connector contact arrangement	177
Table 60 – Internal pass-through conductor specifications	179
Table 61 – Internal drop conductor specifications	179
Table 62 – Internal pass-through conductor specifications	179
Table 63 – Internal power drop conductor specifications	181
Table 64 – Voltage regulator specifications	181
Table 65 – DeviceNet power supply specifications	183
Table 66 – Immunity performance criteria	185

Tableau A.1 – Codes et noms des services DeviceNet.....	212
Tableau A.2 – Paramètres des champs des données de service et de réponse positive	214
Tableau A.3 – Données de service pour la réponse positive à get_attributes_all.....	214
Tableau A.4 – Données de service pour la demande set_attributes_all.....	216
Tableau A.5 – Données de service pour la réponse positive à create.....	218
Tableau A.6 – Données de service pour une demande get_attribute_single	220
Tableau A.7 – Données de service pour la réponse positive à get_attribute_single	220
Tableau A.8 – Données de service pour une demande set_attribute_single	222
Tableau A.9 – Données de service pour une réponse d'erreur	222
Tableau B.1 – Codes d'erreurs DeviceNet.....	226
Tableau D.1 – Types de données élémentaires.....	236
Tableau E.1 – Codes et noms des classes d'objets	242
Tableau E.2 – Attributs de la classe d'objet identité	244
Tableau E.3 – Attributs d'instance de l'objet identité	244
Tableau E.4 – Définitions des bits de l'attribut d'instance états d'un objet identité.....	246
Tableau E.5 – Services communs de l'objet identité.....	248
Tableau E.6 – Demande reset de l'objet identité	248
Tableau E.7 – Valeurs des paramètres du service reset de l'objet identité	248
Tableau E.8 – Réponse get_attributes_all de l'objet identité, au niveau classe	250
Tableau E.9 – Réponse get_attributes_all de l'objet identité, au niveau instance	250
Tableau E.10 – Table états-événements pour l'objet identité.....	254
Tableau E.11 – Attributs de la classe routeur de messages	256
Tableau E.12 – Attributs d'instance de l'objet routeur de messages	256
Tableau E.13 – Services communs du routeur de messages.....	256
Tableau E.14 – Réponse get_attributes_all, au niveau classe	258
Tableau E.15 – Réponse get_attributes_all, au niveau instance	258
Tableau E.16 – Attributs de la classe d'objet gestionnaire d'acquittements	260
Tableau E.17 – Attributs d'instance de l'objet gestionnaire d'acquittements	262
Tableau E.18 – Services communs de l'objet gestionnaire d'acquittements.....	264
Tableau E.19 – Services spécifiques à l'objet gestionnaire d'acquittements	264
Tableau E.20 – Demande add_ackdata_path du gestionnaire d'acquittements.....	264
Tableau E.21 – Demande remove_ackdata_path du gestionnaire d'acquittements	266
Tableau E.22 – Table états-événements pour un objet applicatif E/S producteur.....	276
Tableau E.23 – Table états-événements pour un objet gestionnaire d'acquittements	278
Tableau F.1 – Plages des ID de classes	282
Tableau F.2 – Plages des codes de services	282
Tableau F.3 – Plages des ID d'attributs.....	284
Tableau F.4 – Plage des MAC ID	284
Tableau F.5 – Plages des types d'appareils.....	284

Table A.1 – DeviceNet service codes and names.....	213
Table A.2 – Service data and success response data field parameters	215
Table A.3 – Service data for get_attributes_all success response	215
Table A.4 – Service data for set_attributes_all request	217
Table A.5 – Service data for create success response	219
Table A.6 – Service data for get_attribute_single request	221
Table A.7 – Service data for get_attribute_single success response	221
Table A.8 – Service data for set_attribute_single request.....	223
Table A.9 – Service data for error response request	223
Table B.1 – DeviceNet error codes	227
Table D.1 – Elementary data types	237
Table E.1 – Object class codes and names.....	243
Table E.2 – Identity object class attributes.....	245
Table E.3 – Identity object instance attributes.....	245
Table E.4 – Bit definitions for status instance attribute of identity object.....	247
Table E.5 – Identity object common services	249
Table E.6 – Identity object reset request.....	249
Table E.7 – Identity reset service parameter values.....	249
Table E.8 – Identity object get_attributes_all response, class level.....	251
Table E.9 – Identity object get_attributes_all response, instance level.....	251
Table E.10 – State-event matrix for identity object.....	255
Table E.11 – Message router class attributes.....	257
Table E.12 – Message router instance attributes.....	257
Table E.13 – Message router common services	257
Table E.14 – Get_attributes_all response, class level	259
Table E.15 – Get_attributes_all response, instance level	259
Table E.16 – Acknowledgement handler object class attributes.....	261
Table E.17 – Acknowledgement handler instance attributes	263
Table E.18 – Acknowledgement handler object common services	265
Table E.19 – Acknowledgement handler object-specific services	265
Table E.20 – Acknowledgement handler add_ackdata_path request	265
Table E.21 – Acknowledgement handler remove_ackdata_path request.....	267
Table E.22 – State-event matrix for producing I/O application object.....	277
Table E.23 – State-event matrix for acknowledgement handler object.....	279
Table F.1 – Class ID ranges	283
Table F.2 – Service code ranges.....	283
Table F.3 – Attribute ID ranges	285
Table F.4 – MAC ID range	285
Table F.5 – Device type ranges	285

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

APPAREILLAGE À BASSE TENSION – INTERFACES APPAREIL DE COMMANDE-APPAREIL (CDI) –

Partie 3: DeviceNet

AVANT-PROPOS

- 1) La CEI (Commission Electrotechnique Internationale) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI, entre autres activités, publie des Normes internationales. Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible un accord international sur les sujets étudiés, étant donné que les Comités nationaux intéressés sont représentés dans chaque comité d'études.
- 3) Les documents produits se présentent sous la forme de recommandations internationales. Ils sont publiés comme normes, spécifications techniques, rapports techniques ou guides et agréés comme tels par les Comités nationaux.
- 4) Dans le but d'encourager l'unification internationale, les Comités nationaux de la CEI s'engagent à appliquer de façon transparente, dans toute la mesure possible, les Normes internationales de la CEI dans leurs normes nationales et régionales. Toute divergence entre la norme de la CEI et la norme nationale ou régionale correspondante doit être indiquée en termes clairs dans cette dernière.
- 5) La CEI n'a fixé aucune procédure concernant le marquage comme indication d'approbation et sa responsabilité n'est pas engagée quand un matériel est déclaré conforme à l'une de ses normes.
- 6) L'attention est attirée sur le fait que certains des éléments de la présente Norme internationale peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

La Norme internationale CEI 62026-3 a été établie par le sous-comité 17B: Appareillage à basse tension, du comité d'études 17 de la CEI: Appareillage.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
17B/1053/FDIS	17B/1090/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 3.

Les annexes A, B, C, D, E et F font partie intégrante de cette norme.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant 2002. A cette date, la publication sera

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**LOW-VOLTAGE SWITCHGEAR AND CONTROLGEAR –
CONTROLLER-DEVICE INTERFACES (CDIs) –****Part 3: DeviceNet**

FOREWORD

- 1) The IEC (International Electrotechnical Commission) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of the IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, the IEC publishes International Standards. Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. The IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of the IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested National Committees.
- 3) The documents produced have the form of recommendations for international use and are published in the form of standards, technical specifications, technical reports or guides and they are accepted by the National Committees in that sense.
- 4) In order to promote international unification, IEC National Committees undertake to apply IEC International Standards transparently to the maximum extent possible in their national and regional standards. Any divergence between the IEC Standard and the corresponding national or regional standard shall be clearly indicated in the latter.
- 5) The IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with one of its standards.
- 6) Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. The IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62026-3 has been prepared by subcommittee 17B: Low-voltage switchgear and controlgear, of IEC technical committee 17: Switchgear and controlgear.

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

FDIS	Report on voting
17B/1053/FDIS	17B/1090/RVD

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

This publication has been drafted in accordance with the ISO /IEC Directives, Part 3.

Annexes A, B, C, D, E and F form an integral part of this standard.

The committee has decided that the contents of this publication will remain unchanged until 2002. At this date, the publication will be

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

INTRODUCTION

Les dispositions des règles générales de la CEI 62026-1 sont applicables à la présente Norme internationale, lorsque celle-ci le précise. Les articles et paragraphes des règles générales ainsi rendues applicables, de même que les tableaux, figures et annexes, sont identifiés par référence à la partie 1, par exemple paragraphe 7.2.4.1 de la CEI 62026-1.

L'utilisation de DeviceNet est destinée, entre autres, à des applications industrielles d'automatisation. Ces applications peuvent inclure des appareils tels que les détecteurs de fin de course, les détecteurs de proximité, les vannes électropneumatiques, les relais, les démarreurs, les panneaux d'interface opérateur, les entrées analogiques, les sorties analogiques et les appareils de commande.

IECNORM.COM: Click to view the full PDF of IEC 62026-3:2000

Withd 2000

INTRODUCTION

The provisions of the general rules in IEC 62026-1 are applicable to this International Standard, where specifically called for. General rules clauses and subclauses thus applicable, as well as tables, figures and annexes, are identified by reference to part 1, for example subclause 7.2.4.1 of IEC 62026-1.

DeviceNet 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.

Withd
IECNORM.COM: Click to view the full PDF of IEC 62026-3:2000

APPAREILLAGE À BASSE TENSION – INTERFACES APPAREIL DE COMMANDE-APPAREIL (CDI) –

Partie 3: DeviceNet

1 Domaine d'application

La présente Norme internationale spécifie un système d'interfaces entre un ou plusieurs appareils de commande d'une part, et des appareils et éléments de commutation pour circuits de commande d'autre part. Le système d'interfaces utilise deux paires conductrices torsadées blindées, réunies dans un câble – l'une des paires fournit un support de communication différentiel et l'autre alimente les appareils. Cette partie définit des exigences permettant l'interchangeabilité des composants incluant de telles interfaces.

La présente norme spécifie, pour DeviceNet, les exigences particulières suivantes:

- exigences concernant les interfaces entre appareils de commande et éléments de commutation;
- conditions normales de service pour les appareils;
- dispositions relatives à la construction et au fonctionnement;
- essais de vérification de la conformité aux exigences.

Ces exigences particulières s'ajoutent aux exigences générales de la CEI 62026-1.

2 Références normatives

Les documents normatifs suivants contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente partie de la CEI 62026. Pour les références datées, les amendements ultérieurs ou les révisions de ces publications ne s'appliquent pas. Toutefois, les parties prenantes aux accords fondés sur la présente partie de la CEI 62026 sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des documents normatifs indiqués ci-après. Pour les références non datées, la dernière édition du document normatif en référence s'applique. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur.

CISPR 11:1997, *Appareils industriels, scientifiques et médicaux (ISM) à fréquence radio-électrique – Caractéristiques de perturbations électromagnétiques – Limites et méthodes de mesure*

CEI 60529:1989, *Degrés de protection procurés par les enveloppes (code IP)*

CEI 60947-1:1999, *Appareillage à basse tension – Partie 1: Règles générales*

CEI 60947-5-2:1997, *Appareillage à basse tension – Partie 5-2: Appareils et éléments de commutation pour circuits de commande – Détecteurs de proximité*

CEI 61000-4-2:1995, *Compatibilité électromagnétique (CEM) – Partie 4: Techniques d'essai et de mesure – Section 2: Essais d'immunité aux décharges électrostatiques*. Publication fondamentale en CEM

CEI 61000-4-3:1995, *Compatibilité électromagnétique (CEM) – Partie 4: Techniques d'essai et de mesure – Section 3: Essais d'immunité aux champs électromagnétiques rayonnés aux fréquences radioélectriques*

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

Part 3: DeviceNet

1 Scope

This International Standard specifies an interface system between a single controller or multiple controllers, and control circuit devices or switching elements. The interface system uses two twisted shielded 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 interchangeability of components with such interfaces.

This standard 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 normative documents contain provisions which, through reference in this text, constitute provisions of this part of IEC 62026. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of IEC 62026 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of IEC and ISO maintain registers of currently valid International Standards.

CISPR 11:1997, *Industrial, scientific and medical (ISM) radio-frequency equipment – Electromagnetic disturbance characteristics – Limits and methods of measurement*

IEC 60529:1989, *Degrees of protection provided by enclosures (IP code)*

IEC 60947-1:1999, *Low-voltage switchgear and controlgear – Part 1: General rules*

IEC 60947-5-2:1997, *Low-voltage switchgear and controlgear – Part 5-2: Control circuit devices and switching elements – Proximity switches*

IEC 61000-4-2:1995, *Electromagnetic compatibility (EMC) – Part 4: Testing and measurement techniques – Section 2: Electrostatic discharge immunity test*. Basic EMC publication

IEC 61000-4-3:1995, *Electromagnetic compatibility (EMC) – Part 4: Testing and measurement techniques – Section 3: Radiated, radio-frequency, electromagnetic field immunity test*

CEI 61000-4-4:1995, *Compatibilité électromagnétique (CEM) – Partie 4: Techniques d'essai et de mesure – Section 4: Essais d'immunité aux transitoires électriques rapides en salves*. Publication fondamentale en CEM

CEI 61000-4-5:1995, *Compatibilité électromagnétique (CEM) – Partie 4: Techniques d'essai et de mesure – Section 5: Essais d'immunité aux ondes de choc*

CEI 61000-4-6:1996, *Compatibilité électromagnétique (CEM) – Partie 4: Techniques d'essai et de mesure – Section 6: Immunité aux perturbations conduites, induites par les champs radioélectriques*

CEI 61131-3:1993, *Automates programmables – Partie 3: Langages de programmation*

CEI 62026-1:2000, *Appareillage à basse tension – Interfaces appareil de commande-appareil (CDI) – Partie 1: Règles générales*

ISO/IEC 7498-1:1994, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base: Le modèle de base*

ISO 11898:1993, *Véhicules routiers – Echange d'information numérique – Gestionnaire de réseau de communication à vitesse élevée (CAN) (publiée actuellement en anglais seulement)*

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

3 Définitions et abréviations

Pour les besoins de la présente Norme internationale, les définitions et abréviations suivantes s'appliquent, en plus de celles données dans la CEI 62026-1.

3.1

fragmentation acquittée

fragmentation effectuée sur un message explicite, dans laquelle la transmission d'un fragment par l'objet émetteur est suivie de la transmission d'un acquittement par l'objet chargé de la réception. La réception de chaque fragment est acquittée par l'objet effectuant la réception

3.2

état d'acquittement

champ dans le format d'un message d'acquittement/réponse qui indique si une erreur a été trouvée ou non par le destinataire du message fragmenté. Cela s'applique spécifiquement au protocole de fragmentation de DeviceNet

3.3

objets applicatifs

ensemble de classes d'objets et de leurs instances présentes dans la station. Ces objets assurent la gestion et la production des données et messages échangés à travers des interfaces appareil de commande-appareil (CDI) DeviceNet et à l'intérieur d'une station compatible DeviceNet

3.4

attribut

description d'une caractéristique ou fonction d'un objet accessible en externe. Typiquement, les attributs fournissent des informations d'état ou régissent le fonctionnement d'un objet

IEC 61000-4-4:1995, *Electromagnetic compatibility (EMC) – Part 4: Testing and measurement techniques – Section 4: Electrical fast transient/burst immunity test*. Basic EMC publication

IEC 61000-4-5:1995, *Electromagnetic compatibility (EMC) – Part 4: Testing and measurement techniques – Section 5: Surge immunity tests*

IEC 61000-4-6:1996, *Electromagnetic compatibility (EMC) – Part 4: Testing and measurement techniques – Section 6: Immunity to conducted disturbances, induced by radio-frequency fields*

IEC 61131-3:1993, *Programmable controllers – Part 3: Programming languages*

IEC 62026-1:2000, *Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs) – Part 1: General rules*

ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO 11898:1993, *Road vehicles – Interchange of digital information – Controller area network (CAN) for high-speed communication*

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

3 Definitions and abbreviations

For the purpose of this International Standard, the definitions given in IEC 62026-1 and the following definitions and abbreviations apply.

3.1

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. The reception of each fragment is acknowledged by the receiving object

3.2

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. This applies specifically to the DeviceNet fragmentation protocol

3.3

application objects

set of object classes and their object instances that are available within the node. These objects manage and provide the exchange of data and messages across DeviceNet controller-device interfaces (CDIs) and within the DeviceNet compliant node

3.4

attribute

description of an externally accessible characteristic or feature of an object. Attributes typically provide status information or govern the operation of an object

3.5

requête multiple

communication utilisant la procédure d'échange sur requête multiple (voir définition 3.12 de la CEI 62026-1)

3.6

diffusion

communication d'une station vers toutes les autres stations

3.7

CAN (Controller Area Network)

spécification ISO qui définit une couche physique générique et une procédure d'accès au support pour la couche liaison de données, utilisant un arbitrage sur bits non destructif. Voir l'ISO 11898

3.8

CAN_H

moitié positive du signal physique différentiel CAN

3.9

CAN_L

moitié négative du signal physique différentiel CAN

3.10

client

- a) objet qui utilise les services d'un autre objet (serveur) pour accomplir une tâche. Voir serveur (3.43)
- b) initiateur d'un message auquel un serveur réagit

3.11

service commun

service utilisé par les objets DeviceNet (voir annexe A)

3.12

objets de communication

objets qui assurent la gestion et la production des messages échangés sur DeviceNet en phase d'opération

3.13

connexion

lien logique entre deux ou plusieurs objets applicatifs. Ces objets peuvent se trouver dans une même station ou dans des stations différentes

3.14

ID de connexion (CID)

identificateur de connexion attribué à toutes les transmissions associées à une connexion particulière entre plusieurs stations

3.15

objet connexion

objet qui gère les fonctions spécifiques à la communication associées aux connexions entre stations

3.16

consommateur

extrémité d'une connexion responsable de la réception de données

3.5**bit-strobe**

communication using strobing (see definition 3.13 of IEC 62026-1)

3.6**broadcast**

communication from one node to all other nodes

3.7**CAN (Controller Area Network)**

ISO specification that defines a generic physical layer and data link medium access procedure based on non-destructive bit-wise arbitration. See ISO 11898

3.8**CAN_H**

positive half of the differential physical CAN signal

3.9**CAN_L**

negative half of the differential physical CAN signal

3.10**client**

- a) object which uses the services of another (server) object to perform a task. See server (3.43)
- b) initiator of a message to which a server reacts

3.11**common service**

service used by DeviceNet objects (see annex A)

3.12**communication objects**

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

3.13**connection**

logical binding between two or more application objects. These application objects may be located at the same node or at different nodes

3.14**connection ID (CID)**

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

3.15**connection object**

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

3.16**consumer**

end-point of a connection that is responsible for receiving data

3.17

MAC ID de destination

MAC ID d'une station destinataire d'un message

3.18

dérivateur d'appareil

point physique de raccordement d'un appareil DeviceNet à un câble principal ou à un câble de dérivation

3.19

type d'appareil

identification d'une série d'informations dépendant de l'appareil et décrivant une combinaison appropriée d'options sélectionnées dans toutes les couches de la pile de communication

3.20

dominant

l'un des deux niveaux logiques complémentaires du signal physique. Le niveau dominant est un «0» logique

3.21

détection de duplication d'adresse MAC ID

protocole défini par DeviceNet qui garantit qu'un même MAC ID ne sera pas attribué à deux stations sur la même liaison

3.22

messagerie explicite

message qui commande l'exécution d'une tâche particulière et le renvoi des résultats correspondants au demandeur

3.23

fragmentation

protocole DeviceNet fourni par l'objet connexion et qui définit une méthode grâce à laquelle les données d'une longueur supérieure à huit (8) octets peuvent être transmises

3.24

client du groupe 2

appareil disposant de la fonction gestionnaire de messages non connectés (UCMM) et ayant obtenu la propriété du jeu de connexions maître/esclave prédéfini d'un serveur, de telle sorte qu'il peut agir en tant que client sur ces connexions

3.25

client du groupe 2 seulement

appareil agissant comme un client du groupe 2 envers un serveur du groupe 2 seulement. Le client du groupe 2 seulement fournit la fonctionnalité UCMM aux serveurs du groupe 2 seulement qui lui sont alloués

3.26

serveur du groupe 2

appareil disposant de la fonction gestionnaire de messages non connectés (UCMM), configuré afin d'agir comme serveur des connexions maître/esclave prédéfinies

3.27

serveur du groupe 2 seulement

appareil esclave privé de la fonction UCMM et qui utilise le jeu de connexions maître/esclave prédéfini pour établir des communications. Un appareil du groupe 2 seulement ne peut transmettre et recevoir qu'avec les identificateurs spécifiés par le jeu de connexions maître/esclave prédéfini

3.17**destination MAC ID**

MAC ID of a node that is to receive a message

3.18**device tap**

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

3.19**device type**

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

3.20**dominant**

one of two complementary logic levels on the physical signal. The dominant level is a logical "0"

3.21**duplicate MAC ID detection**

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

3.22**explicit messaging**

message that commands the performance of a particular task and the return of the results of the task performance to the requester

3.23**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

3.24**group 2 client**

Unconnected Message Manager (UCMM) capable device that has gained ownership of the predefined master/slave connection set within a server so that it may act as the client on those connections

3.25**group 2 only client**

device that is acting as a group 2 client to a group 2 only server. The group 2 only client provides the UCMM functionality for group 2 only servers that are allocated to it

3.26**group 2 server**

UCMM capable device that has been configured to act as the server for the predefined master/slave identifier connections

3.27**group 2 only server**

slave device that is UCMM incapable and uses the predefined master/slave connection set to establish communications. A group 2 only device can transmit and receive only those identifiers defined by the predefined master/slave connection set

3.28

connexion d'E/S

connexion entre un producteur et un ou plusieurs consommateurs dans le but d'échanger des données d'E/S à temps critique spécifiques à une application

3.29

données d'E/S

informations transférées entre des points d'E/S et les appareils de commande qui utilisent et définissent ces valeurs

3.30

messagerie d'E/S

échange de données dans un format précédemment défini

3.31

appareil isolé

appareil dans lequel certains composants n'ont pas comme référence le *U*– de la couche physique. Voir appareil non isolé (3.36)

3.32

maître

station qui rassemble et distribue les E/S en utilisant le jeu de connexions maître/esclave prédéfini

3.33

Medium Access Control (MAC) ID

adresse d'une station DeviceNet pour la couche liaison de données

3.34

connexion multidestinataire

connexion logique entre un objet et plusieurs autres objets. Une connexion multidestinataire permet le transfert des données en une seule transaction entre un producteur et de multiples consommateurs partageant la même connexion

3.35

station

entité DeviceNet identifiée au niveau de la couche liaison de données par un MAC ID unique

NOTE Plusieurs stations DeviceNet peuvent être intégrées à un même appareil mais elles apparaissent sur la liaison DeviceNet comme des stations logiquement distinctes.

3.36

appareil non isolé

appareil dans lequel tous les composants ont pour référence le *U*– de la couche physique. Voir appareil isolé (3.31)

3.37

objet

- a) représentation abstraite des capacités d'un appareil. Les objets peuvent être composés de certains ou de tous les composants suivants: 1) données (informations qui changent avec le temps); 2) configuration (paramètres définissant le comportement); 3) procédures (actions qui peuvent être effectuées à l'aide de données et d'une configuration)
- b) ensemble cohérent de données (sous forme de variables) et de procédures opératoires sur ces données

3.28**I/O connection**

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

3.29**I/O data**

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

3.30**I/O messaging**

exchange of data in a previously defined format

3.31**isolated device**

device in which some of the components are not referenced to the *U*– of the physical layer. See non-isolated device (3.36)

3.32**master**

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

3.33**Medium Access Control (MAC) ID**

link address of a DeviceNet node

3.34**multicast connection**

logical connection from one object to multiple other objects. A multicast connection allows data to be transferred in a single transaction from a producer to multiple consumers sharing the same connection

3.35**node**

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

NOTE Multiple DeviceNet nodes may be implemented in one device but they appear as logically distinct nodes on the DeviceNet link.

3.36**non-isolated device**

device in which all components are referenced to the *U*– of the physical layer. See isolated device (3.31)

3.37**object**

- a) abstract representation of a device's capabilities. Objects may be composed of any or all of the following components: 1) data (information which changes with time); 2) configuration (parameters for behaviour); 3) procedures (actions that may be performed using data and configuration)
- b) collection of related data (in the form of variables) and procedures for operating on that data

3.38

connexion point à point

connexion qui existe entre deux objets seulement. Les connexions de messagerie explicite sont toujours point à point. Les connexions d'E/S peuvent être point à point ou multidestinatoires. Voir connexion multidestinataire (3.34)

3.39

jeu de connexions maître/esclave prédéfini

utilisation d'une connexion de messagerie explicite pour créer et configurer des objets connexion à chaque extrémité d'une connexion. Utilise les règles générales comme base pour définir un jeu de connexions qui facilite les communications types d'une relation maître/esclave

3.40

producteur

extrémité d'une connexion responsable de l'envoi de données

3.41

récessif

l'un des deux niveaux logiques complémentaires du signal physique. Le niveau récessif est un «1» logique

3.42

numéro de série

nombre entier de 32 bits attribué par chaque fabricant à chaque appareil DeviceNet. Le nombre est stocké dans l'appareil comme attribut de l'objet identité; il est unique pour le fabricant

3.43

serveur

objet qui fournit des services à un autre objet (client). Voir client (3.10)

3.44

service

opération ou fonction qu'un objet effectue à la demande d'un autre objet

3.45

esclave

station qui retourne des données à son maître en utilisant le jeu de connexions maître/esclave prédéfini et une méthode de communication déterminée par le maître

3.46

MAC ID source

MAC ID d'une station émettrice d'un message

3.47

déclenchement

service utilisé par une application pour lancer la production de données

3.48

appareil avec capacité UCMM

appareil disposant de la fonctionnalité UCMM (voir 3.51)

3.49

appareil sans capacité UCMM

appareil privé de la fonctionnalité UCMM (voir 3.51)

3.38**point-to-point connection**

connection that exists between two objects only. Explicit messaging connections are always point-to-point. I/O connections may be either point-to-point or multicast. See multicast connection (3.34)

3.39**predefined master/slave connection set**

utilisation of an explicit messaging connection to create and configure connection objects within each connection end-point. Uses the general rules as a basis for the definition of a set of connections which facilitates communications typically seen in a master/slave relationship

3.40**producer**

end-point of a connection that is responsible for sending data

3.41**recessive**

one of two complementary logic levels on the physical signal. The recessive level is a logical "1"

3.42**serial number**

unique 32-bit integer assigned by each manufacturer to every DeviceNet device. The number is stored within the device as an attribute of the identity object and is unique with respect to the manufacturer

3.43**server**

object which provides services to another (client) object. See client (3.10)

3.44**service**

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

3.45**slave**

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

3.46**source MAC ID**

MAC ID of a node that is transmitting a message

3.47**trigger**

service used by an application to initiate the production of data

3.48**UCMM capable device**

device that supports the UCMM (see 3.51)

3.49**UCMM incapable device**

device that does not support the UCMM (see 3.51)

3.50

message explicite non connecté

message explicite entre stations n'ayant pas encore établi de connexion entre elles

3.51

gestionnaire de messages non connectés (UCMM)

fonction de réception et de traitement de messages explicites non connectés dans une station

3.52

nombre entier court non signé (USINT)

nombre entier de 8 bits

3.53

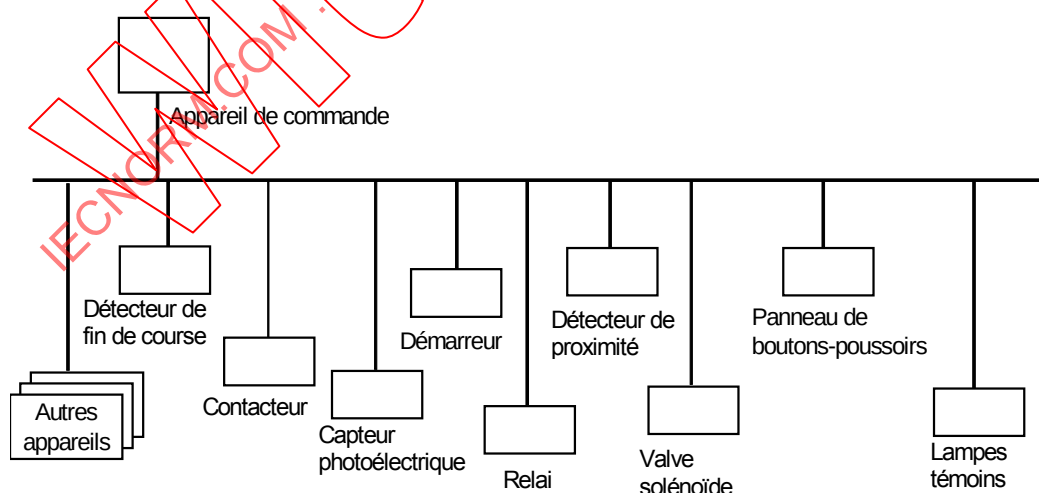
nombre entier non signé (UINT)

nombre entier de 16 bits

4 Classifications

4.1 Généralités

DeviceNet définit un système d'interface entre appareils de commande et appareils ou éléments de commutation pour circuits de commande. DeviceNet utilise deux paires conductrices torsadées blindées réunies dans un câble; l'une des paires fournit un support de communication différentiel et l'autre alimente les appareils. Le courant maximal accepté est de 8 A à 24 V c.c. Les données sont transmises aux vitesses de 125 kbit/s, 250 kbit/s ou 500 kbit/s avec des câbles de longueurs maximales de 500 m, 250 m et 100 m respectivement. Un maximum de huit octets de données peut être transmis sans fragmentation. Soixante quatre stations maximum peuvent être connectées à l'aide d'une topologie linéaire comprenant une ligne principale et des lignes de dérivation (voir figure 1). DeviceNet permet la transmission de données d'E/S, de diagnostics, de messages et d'informations pour la programmation et la configuration. Les échanges de données peuvent être de nature événementielle (changement d'état), cyclique, sur interrogation ou multidestinataires.



IEC 1204/2000

Figure 1 – Interfaces appareil de commande-appareil DeviceNet types

3.50**unconnected explicit message**

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

3.51**Unconnected Message Manager (UCMM)**

function within a node that receives and processes unconnected explicit messages

3.52**Unsigned Short Integer (USINT)**

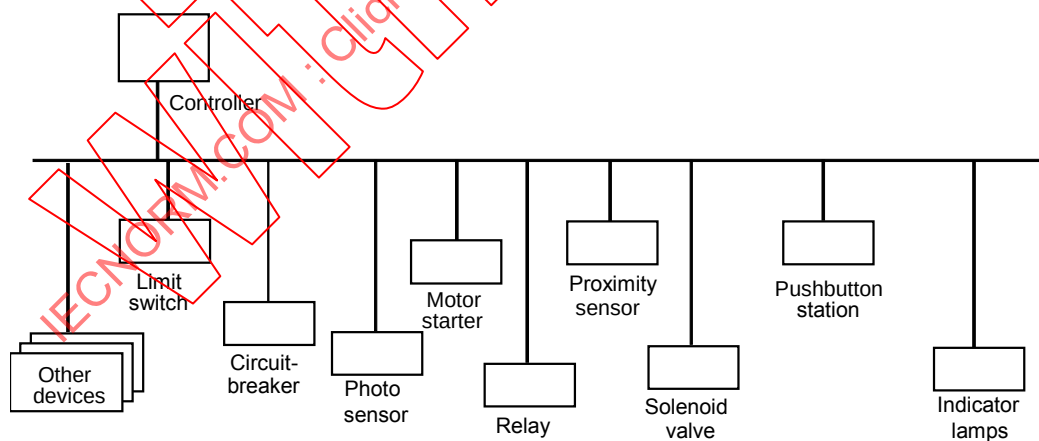
8-bit integer

3.53**Unsigned Integer (UINT)**

16-bit integer

4 Classification**4.1 General**

DeviceNet interfaces controlling devices to control circuit devices or switching elements. DeviceNet uses two twisted shielded 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 eight 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 or multicast.



IEC 1204/2000

Figure 1 – Typical DeviceNet controller-device interfaces

La présente partie de la CEI 62026 définit un système basé sur des connexions, afin de faciliter toutes les communications entre applications. Une connexion DeviceNet fournit une voie de communication entre ses extrémités multiples. Les extrémités d'une connexion sont des applications qui ont besoin de partager des données. Les transmissions associées à une connexion particulière reçoivent une valeur d'identification lorsque cette connexion est établie. Cette valeur d'identification est appelée ID de la connexion (CID).

Les objets connexion modélisent les caractéristiques de communication d'une relation particulière entre applications.

Le système de connexions DeviceNet définit les moyens dynamiques par lesquels les deux types suivants de connexions peuvent être établis.

- **Connexions d'E/S:** fournissent des voies de communication dédiées, spécifiques, entre une application productrice et une ou plusieurs applications consommatrices. Les données d'E/S spécifiques à l'application sont transférées par ces voies.

Les messages d'E/S sont échangés via les connexions d'E/S. Un message d'E/S comprend un CID et des données d'E/S associées. Les extrémités de la connexion doivent avoir connaissance de l'utilisation prévue ou de la signification du message d'E/S.

La présente partie de la CEI 62026 ne définit aucune utilisation particulière de la messagerie d'E/S. De très nombreuses fonctions peuvent être accomplies avec la messagerie d'E/S. La signification et/ou l'utilisation prévue de tous les messages d'E/S doivent être portées à la connaissance du système, soit par le type particulier de produit transmettant un message d'E/S, soit à partir de la configuration effectuée en utilisant la messagerie explicite.

- **Connexions de messagerie explicite:** fournissent des voies de communication génériques tous usages entre deux appareils. Les messages explicites assurent les communications du type demande/réponse.

Les messages explicites sont échangés via des connexions de messagerie explicite. Ils sont utilisés pour commander l'exécution d'une tâche particulière et pour en rapporter les résultats.

DeviceNet définit un protocole de messagerie explicite qui indique la signification du message. Un message explicite consiste en un CID et des informations pour le protocole de messagerie associé.

Les règles qui régissent l'établissement dynamique de ces connexions servent de bases à partir desquelles un jeu prédéfini de connexions est déterminé.

4.2 Modèle de communication DeviceNet

Le modèle abstrait de communication orienté objet d'une station DeviceNet comprend les éléments suivants:

- **gestionnaire de messages non connectés (UCMM):** traite les messages DeviceNet explicites non connectés;
- **objet identité:** identifie et fournit des informations générales concernant l'appareil;
- **classe connexion:** attribue et gère les ressources internes associées aux connexions d'E/S et aux connexions de messagerie explicite;
- **objet connexion:** gère les aspects spécifiques de la communication associés à une relation particulière entre applications;
- **objet DeviceNet:** fournit la configuration et l'état d'un CDI DeviceNet physique;
- **routeur de messages:** transmet un message explicite de demande à l'objet approprié;
- **objets applicatifs:** réalisent les fonctions prévues pour le produit.

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. Transactions associated with a particular connection are assigned an identification value when a connection is established. This identification value is called connection ID (CID).

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

The 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 move 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 the 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 explicit request messages to the appropriate object;
- **application objects:** implement the intended purpose of the product.

4.3 DeviceNet et CAN

DeviceNet est basé sur l'ISO 11898 et utilise la technologie CAN (Controller Area Network).

Les correspondances entre DeviceNet, CAN et le modèle de référence OSI (ISO/IEC 7498-1) sont illustrées dans la figure 2.

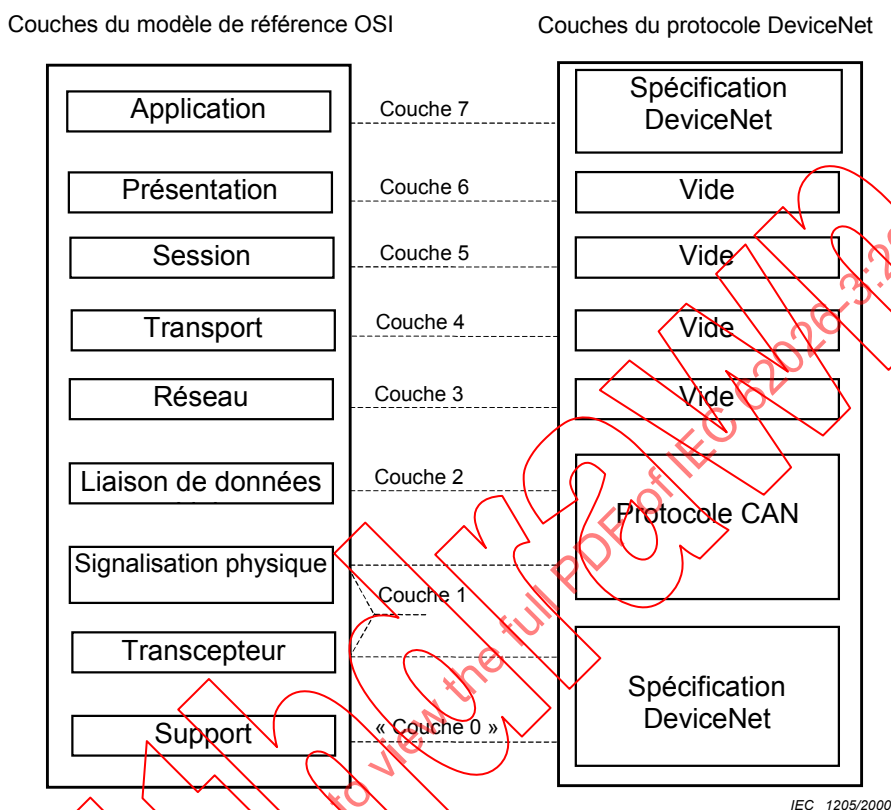


Figure 2 – Comparaison entre l'architecture du protocole DeviceNet et le modèle de référence OSI

5 Caractéristiques

5.1 Connexions DeviceNet

5.1.1 Généralités

DeviceNet est une interface appareil de commande-appareil basée sur les connexions. Les connexions DeviceNet sont utilisées pour offrir des voies logiques entre plusieurs applications. Lorsqu'une connexion est établie, les transmissions associées reçoivent un ID de la connexion (CID). Si la connexion implique un échange bidirectionnel, deux valeurs CID doivent être attribuées à la connexion.

DeviceNet utilise le champ identificateur de CAN et définit les étapes à suivre dans l'établissement dynamique des connexions d'E/S et de messagerie explicite.

1.1.2 Utilisation du champ identificateur de CAN par DeviceNet

NOTE DeviceNet utilise le champ identificateur de CAN de 11 bits («Format de trame standard») décrit dans l'ISO 11898.

Les 11 bits de l'identificateur de CAN disponibles dans DeviceNet sont répartis entre quatre groupes de messages: groupe 1, groupe 2, groupe 3 et groupe 4.

4.3 DeviceNet and CAN

DeviceNet is based on ISO 11898 and uses the Controller Area Network (CAN) technology.

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

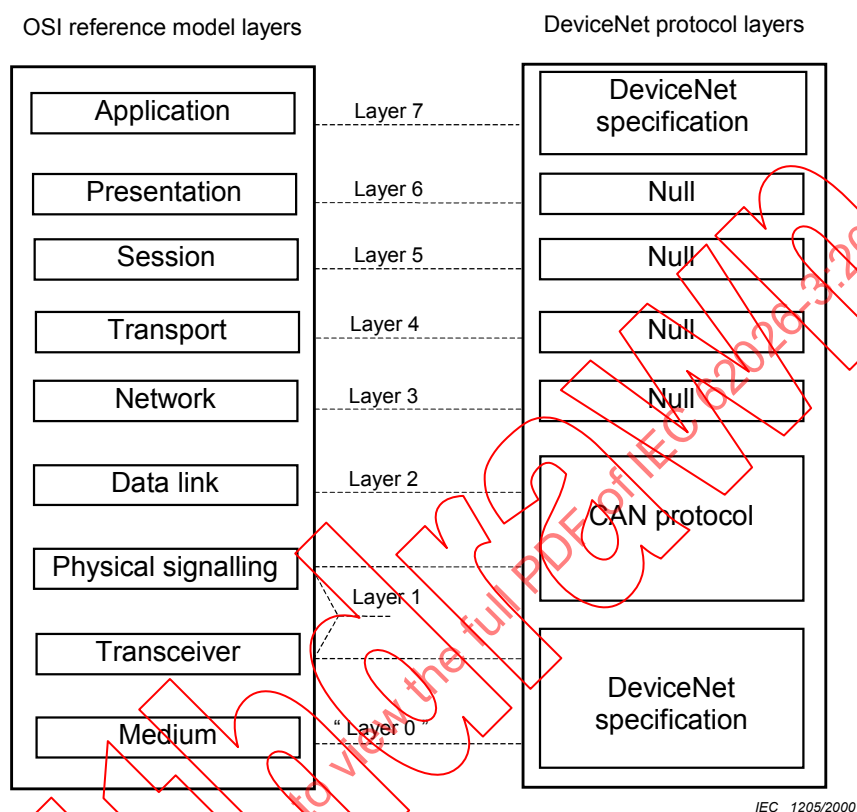


Figure 2 – DeviceNet protocol architecture compared with the OSI reference model

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

NOTE DeviceNet uses the 11 bit CAN identifier field ("Standard frame format") as described in ISO 11898.

The 11 CAN identifier bits available within DeviceNet are subdivided into four separate message groups: group 1, group 2, group 3 and group 4.

Pour les messages connectés, le CID est placé dans le champ identificateur de CAN. La figure 3 décrit également les éléments d'un CID DeviceNet.

BITS DE L' IDENTIFICATEUR											PLAGE HEXA	UTILISATION DE L'IDENTIFICATEUR	
10	9	8	7	6	5	4	3	2	1	0			
0	ID de message du groupe 1				MAC ID source						000 - 3ff	Groupe de messages 1	
1	0	MAC ID					ID de message du groupe 2					400 - 5ff	Groupe de messages 2
1	1	ID message du groupe 3			MAC ID source						600 - 7bf	Groupe de messages 3	
1	1	1	1	1	ID de message du groupe 4 (0 - 2f)						7c0 - 7ef	Groupe de messages 4	
1	1	1	1	1	1	1	X	X	X	X	7f0 - 7ff	Identificateurs de CAN incorrects	
10	9	8	7	6	5	4	3	2	1	0			

IEC 1206/2000

Figure 3 – Utilisation du champ identificateur de CAN par DeviceNet

Comme le montre la figure 3, le champ identificateur de CAN de DeviceNet contient les éléments suivants:

- **ID de message:** identifie un message au sein d'un groupe de messages dans une station particulière. Lorsqu'une connexion est établie, les stations utilisent un ID de message combiné avec un MAC ID pour générer un CID. Le CID qui en résulte est placé dans le champ identificateur de CAN associé aux transmissions correspondantes;
- **MAC ID source:** les groupes 1 et 3 imposent de spécifier un MAC ID source dans le champ identificateur de CAN;
- **MAC ID:** le groupe 2 permet de spécifier la source ou la destination dans la partie MAC ID du champ identificateur de CAN.

Les deux types de connexions, de messagerie explicite et d'E/S, peuvent être établis en utilisant les groupes de messages 1, 2 et 3.

Dans le groupe 2, les valeurs 6 et 7 d'ID de message sont utilisées de la manière suivante:

- l'ID de message 6 du groupe 2 est utilisée pour la configuration des communications utilisées dans une application maître/esclave (voir 5.5);
- l'ID de message 7 du groupe 2 est utilisée pour la détection de stations ayant reçu des MAC ID identiques (voir 5.4).

Dans le groupe 3, les valeurs 5, 6 et 7 d'ID de message sont utilisées comme suit:

- l'ID de message 5 du groupe 3 est utilisée lors de l'envoi de réponses associées à des messages explicites non connectés de demande;
- l'ID de message 6 du groupe 3 est utilisée lors de l'envoi de messages explicites non connectés de demande;
- l'ID de message 7 du groupe 3 n'est pas valide et ne doit pas être utilisée.

Le groupe 4 est réservé à un usage futur.

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						000 - 3ff	Message group 1
1	0	MAC ID					Group 2 message ID				400 - 5ff	Message group 2
1	1	Group 3 message ID			Source MAC ID						600 - 7bf	Message group 3
1	1	1	1	1	Group 4 message ID (0 - 2f)						7c0 - 7ef	Message group 4
1	1	1	1	1	1	1	X	X	X	X	7f0 - 7ff	Invalid CAN identifiers
10	9	8	7	6	5	4	3	2	1	0		

IEC 1206/2000

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 the source or the 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 5.4).

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;
- 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 is reserved for future use.

1.1.3 Etablissement des connexions

1.1.3.1 Connexions de messagerie explicite et UCMM

Le groupe de messages 3 définit le codage des identificateurs pour les messages explicites non connectés. Les messages explicites non connectés établissent et gèrent les connexions de la messagerie explicite. Les messages de demande non connectés sont spécifiés par la transmission d'un message du groupe 3 dont l'élément ID de message est fixé à 6. Les seuls services valides pouvant être transmis via des messages de demande explicites non connectés sont les suivants:

- demande d'ouverture de connexion de messagerie explicite;
- demande de fermeture de connexion.

Les réponses aux demandes explicites non connectées sont transmises comme messages de réponse non connectés. Les messages de réponse non connectés sont spécifiés par la transmission d'un message du groupe 3 dont l'élément ID de message est fixé à 5. Les seuls services valides pouvant être transmis via des messages de réponse explicites non connectés sont les suivants:

- réponse à une ouverture de connexion de messagerie explicite;
- réponse à une fermeture de connexion;
- réponse d'erreur.

1.1.3.2 Connexions d'E/S

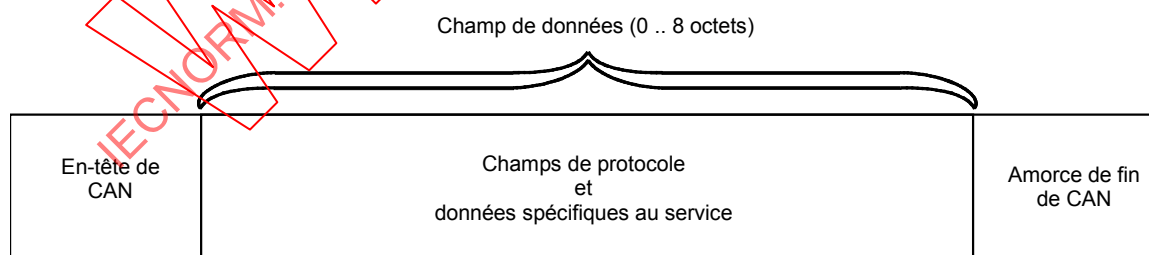
Les connexions d'E/S sont établies dynamiquement par l'intermédiaire de la classe connexion via une connexion de messagerie explicite préalablement établie. Une instance connexion doit être créée et configurée dans chaque station.

1.2 Protocole de messagerie DeviceNet

1.2.1 Messagerie explicite

1.2.1.1 Généralités

Le présent paragraphe décrit le protocole de messagerie explicite et présente les détails associés à l'établissement dynamique des connexions de la messagerie explicite (voir figure 4).



IEC 1207/2000

Figure 4 – Utilisation du champ de données CAN dans un message explicite

5.1.3 Connection establishment

5.1.3.1 Explicit messaging connections and 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.

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.

5.1.3.2 I/O connections

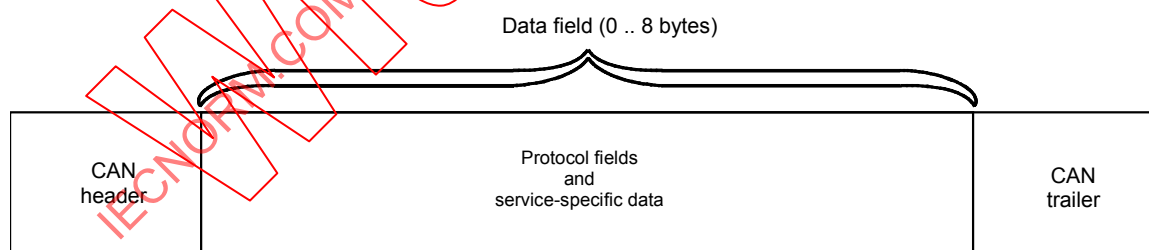
I/O connections are dynamically established by interfacing with the connection class across a previously established explicit messaging connection. A connection instance shall be created and configured at each node.

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).



IEC 1207/2000

Figure 4 – Explicit message CAN data field use

La figure 5 indique le format du champ de données CAN associé aux messages explicites.

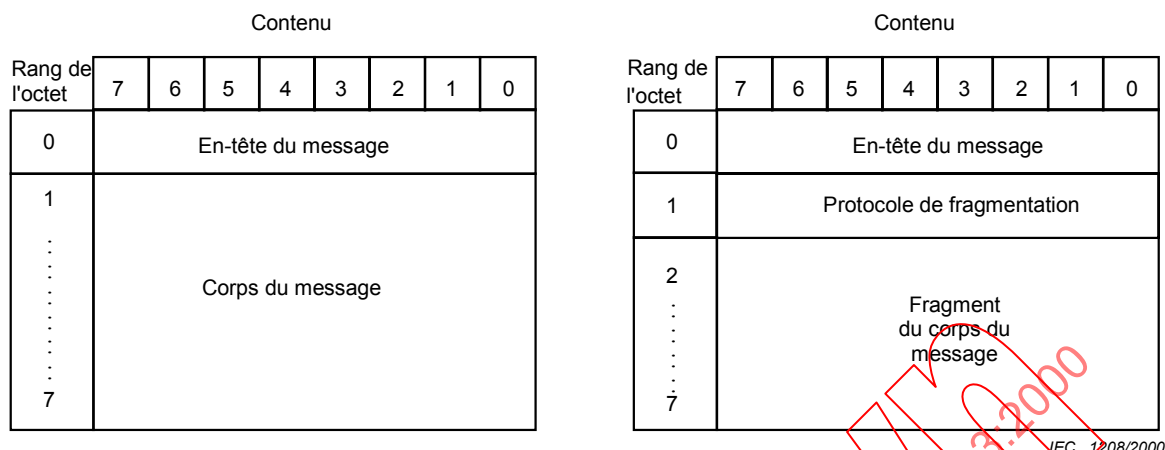


Figure 5 – Format du champ de données des messages explicites

Le champ de données d'une transmission qui contient la totalité du message explicite comprend:

- un en-tête de message;
- la totalité du corps du message.

Si le message explicite est d'une longueur supérieure à huit octets, il doit être transmis par fragments. La fonction de fragmentation/réassemblage est fournie par l'objet connexion. Un tel fragment de message explicite comprend:

- un en-tête de message;
- le protocole de fragmentation (voir 5.2.3.2);
- un fragment du corps du message.

1.2.1.2 En-tête de message

L'en-tête de message se trouve dans l'octet de rang zéro du champ de données CAN d'un message explicite et doit être formaté comme indiqué dans la figure 6.

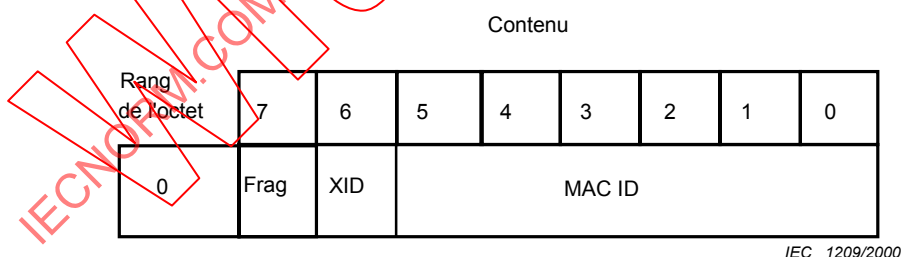


Figure 6 – Format de l'en-tête d'un message explicite

Contenu de l'en-tête du message:

- **frag** (bit de fragment): indique si cette transmission est ou n'est pas une partie d'un message explicite fragmenté:
 0 = non fragmenté
 1 = fragmenté
- **XID (ID de la transaction)**: ce champ est utilisé par une application pour faire correspondre une réponse à la demande associée. Le serveur se contente de renvoyer le contenu de ce champ dans le message réponse. Le serveur n'utilise pas ce champ pour effectuer une quelconque logique de détection de messages dupliqués;
- **MAC ID**: contient le MAC ID source ou de destination.

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

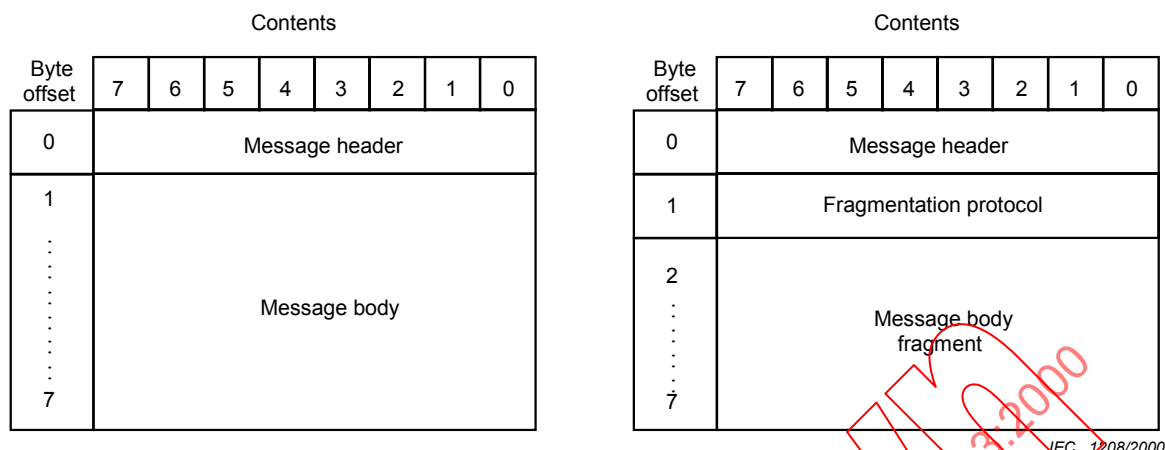


Figure 5 – Explicit message data field format

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

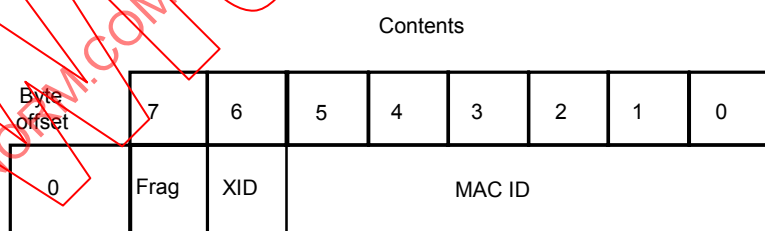
- a message header;
- the entire message body.

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.



IEC 1209/2000

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.

Lorsqu'un message explicite est reçu, le champ MAC ID de l'en-tête du message est examiné. Si le MAC ID de destination est spécifié dans le CID, le MAC ID source de l'autre extrémité doit être spécifié dans l'en-tête du message. Si le MAC ID source est spécifié dans le CID, le MAC ID de la station de réception doit être spécifié dans l'en-tête du message. Si aucune de ces conditions n'est remplie, le message doit être rejeté.

1.2.1.3 Corps du message

Le corps du message se compose d'un champ de service et d'arguments spécifiques au service.

Le premier argument spécifié dans le corps du message est le champ de service, qui identifie la demande ou la réponse particulière délivrée par ce message. La figure 7 illustre le format de ce champ.

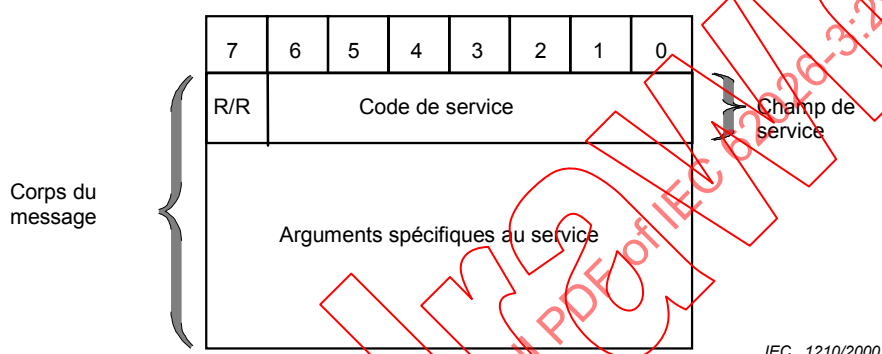


Figure 7 – Format du champ de service

Contenu du champ de service:

- **code de service:** valeur spécifiée dans les sept bits de poids faible de l'octet du champ de service et qui indique le type de service transmis;
- **R/R:** bit de poids fort du champ de service. Sa valeur détermine si le message est une demande ou une réponse:
 - 0 = demande
 - 1 = réponse

DeviceNet définit un ensemble de services communs. Ces services communs de DeviceNet sont l'ensemble des services publics dont les paramètres et comportements requis sont définis dans la présente norme (voir annexe A).

Les informations qui suivent le champ de service sont spécifiques au type particulier de service transmis.

1.2.1.4 Protocole de fragmentation

Si la transmission est une portion d'un message explicite fragmenté, le champ de données contient l'en-tête du message, le protocole de fragmentation et un fragment du corps du message. Le protocole de fragmentation facilite la fragmentation et le réassemblage des messages explicites dont le contenu dépasse huit octets (voir 5.2.3).

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.

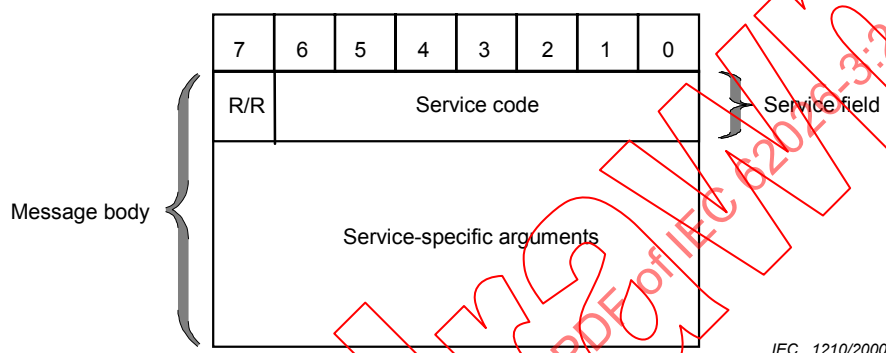


Figure 7 – Service field format

Service field contents:

- **service code:** the value specified within the least significant seven 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 this standard (see annex A).

The 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 contains 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 eight bytes (see 5.2.3).

1.2.1.5 Services UCMM

1.2.1.5.1 Généralités

Le gestionnaire de messages non connectés (UCMM) assure l'établissement dynamique des connexions de messagerie explicite. Le présent paragraphe présente une description détaillée des arguments spécifiques aux services d'ouverture et de fermeture de connexions de messagerie explicite fournis par l'UCMM. Voir 9.3.3 pour les spécifications d'essais concernant l'établissement des connexions de messagerie explicite.

L'UCMM traite deux services qui gèrent l'affectation et la désaffectation des connexions de la messagerie explicite:

- **ouverture d'une connexion de messagerie explicite:** code de service = $4B_{hex}$. Utilisé pour établir une connexion de messagerie explicite;
- **fermeture d'une connexion:** code de service = $4C_{hex}$. Utilisé pour supprimer un objet connexion et libérer toutes les ressources associées.

L'accès à ces services s'effectue à l'aide des champs identificateurs de CAN de demande et de réponse explicites non connectées représentés à la figure 3. Lors de l'exécution d'une demande explicite non connectée, l'UCMM peut avoir besoin de retourner une indication d'erreur au demandeur et, par suite, le message réponse explicite d'erreur peut être transmis à l'aide de l'identificateur de CAN de réponse explicite non connectée.

1.2.1.5.2 Demande d'ouverture de connexion de messagerie explicite

Ce service demande l'établissement d'une connexion logique entre deux stations par laquelle les messages explicites seront transmis. Ce service est transmis en tant que message de demande non connecté (groupe de messages 3, ID de message 6) (voir figure 8).

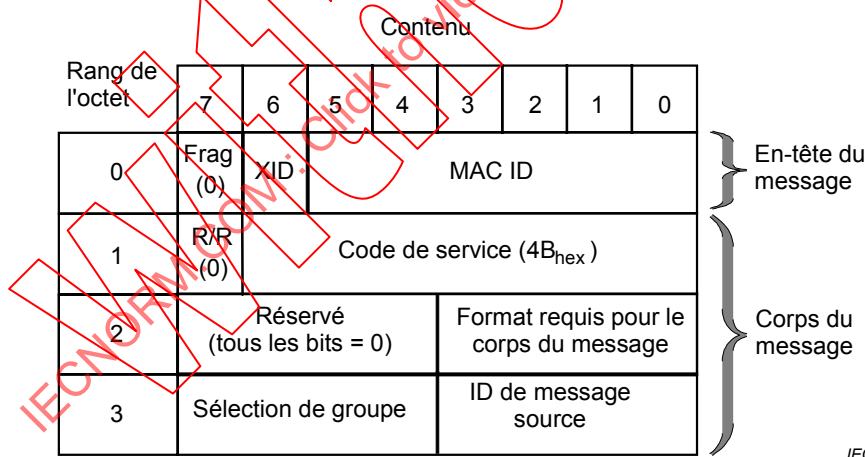


Figure 8 – Format de demande d'ouverture de connexion de messagerie explicite

Contenu de la demande d'ouverture de connexion de messagerie explicite:

- **frag (0)/ ID de transaction/MAC ID:** voir 5.2.1.2. Le MAC ID de destination est toujours spécifié dans l'en-tête du message associé à une demande/réponse d'ouverture de connexion de messagerie explicite;
- **bit R/R (0):** indique qu'il s'agit d'un message de demande;
- **code de service ($4B_{hex}$):** identifie ce message en tant que service d'ouverture de connexion de messagerie explicite;
- **bits réservés:** ils doivent être mis à zéro par l'émetteur;

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 and close explicit messaging connection services provided by the UCMM. See 9.3.3 for the test specifications regarding the establishment of explicit messaging connections.

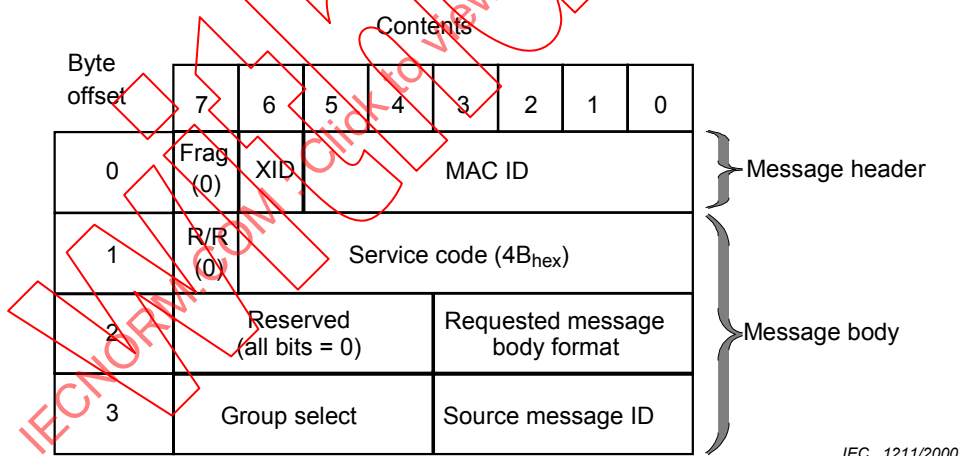
The UCMM processes two services which manage the allocation and deallocation of explicit messaging connections:

- **open explicit messaging connection:** service code = $4B_{\text{hex}}$. Used to establish an explicit messaging connection;
- **close connection:** service code = $4C_{\text{hex}}$. 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 shown in figure 3. 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).



IEC 1211/2000

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 ($4B_{\text{hex}}$):** identifies this message as an open explicit messaging connection service;
- **reserved bits:** these shall be set to zero by the transmitter;

- **format requis pour le corps du message:** champ utilisé par le client pour demander un format particulier de corps de message applicable aux messages explicites transmis ultérieurement via cette connexion.

La station serveur répondant à cette demande d'ouverture de connexion de messagerie explicite choisit le format de corps de message à utiliser sur cette connexion. Voir le tableau 1 pour les valeurs de format de corps de message. Les serveurs doivent agir de l'une ou l'autre des façons suivantes:

- refuser la demande et renvoyer la valeur de format appropriée dans la réponse d'ouverture de connexion de messagerie explicite;
- accepter cette demande en renvoyant la même valeur de format dans la réponse d'ouverture de connexion de messagerie explicite;

Tableau 1 – Valeurs de format de corps de message

Valeur	Signification
0	DeviceNet (8/8). Classe = nombre entier de 8 bits, ID d'instance = nombre entier de 8 bits ¹⁾
1	DeviceNet (8/16). Classe = nombre entier de 8 bits, ID d'instance = nombre entier de 16 bits ²⁾
2	DeviceNet (16/16). Classe = nombre entier de 16 bits, ID d'instance = nombre entier de 16 bits ³⁾
3	DeviceNet (16/8). Classe = nombre entier de 16 bits, ID d'instance = nombre entier de 8 bits ⁴⁾
4 – F _{hex}	Réservé
NOTE Les messages transmis via cette connexion sont formatés comme décrit en 5.2.1.6.	
¹⁾ Les champs d'ID de classe et d'instance sont spécifiés comme nombres entiers de 8 bits (USINT). ²⁾ L'ID de classe est spécifié comme nombre entier de 8 bits (USINT) et l'ID d'instance est spécifié comme nombre entier de 16 bits (UINT). ³⁾ Les champs d'ID de classe et d'instance sont spécifiés comme nombres entiers de 16 bits (UINT). ⁴⁾ L'ID de classe est spécifié comme nombre entier de 16 bits (UINT) et l'ID d'instance est spécifié comme nombre entier de 8 bits (USINT).	

- **sélection de groupe:** champ qui indique le groupe de messages à utiliser pour échanger les messages associés à cette connexion. Les valeurs de sélection des groupes sont indiquées dans le tableau 2;

Tableau 2 – Valeurs de sélection des groupes

Valeur	Signification
0	Groupe de messages 1
1	Groupe de messages 2 ¹⁾
2	Réservé
3	Groupe de messages 3
4 – F _{hex}	Réservé
¹⁾ L'identificateur du groupe de messages 2 permet la spécification du MAC ID source ou de destination. Pour les connexions de messagerie explicite établies via le groupe de messages 2, le client place le MAC ID du serveur dans l'ID de connexion lors de la transmission de messages via cette connexion. Le serveur place son propre MAC ID dans l'ID de connexion lors de la transmission de messages via cette connexion. Cette procédure impose au serveur d'allouer deux ID de message distincts à partir de son pool de groupe 2.	

- **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.

The server node responding to this open explicit messaging request defines the 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 the appropriate format value in the open explicit messaging connection response;
- 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 = 8-bit integer, instance ID = 8-bit integer ¹⁾
1	DeviceNet (8/16). Class = 8-bit integer, instance ID = 16-bit integer ²⁾
2	DeviceNet (16/16). Class = 16-bit integer, instance ID = 16-bit integer ³⁾
3	DeviceNet (16/8). Class = 16-bit integer, instance ID = 8-bit integer ⁴⁾
4 – F _{hex}	Reserved
NOTE Messages transmitted across this connection are formatted as described in 5.2.1.6.	
¹⁾ The class and instance ID fields are specified as 8-bit integers (USINT). ²⁾ The class ID is specified as an 8-bit integer (USINT) and the instance ID is specified as a 16-bit integer (UINT). ³⁾ The class and instance ID fields are specified as 16-bit integers (UINT). ⁴⁾ The class ID is specified as a 16-bit integer (UINT) and the instance ID is specified as an 8-bit integer (USINT).	

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

Table 2 – Group select values

Value	Meaning
0	Message group 1
1	Message group 2 ¹⁾
2	Reserved
3	Message group 3
4 – F _{hex}	Reserved
¹⁾ 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.	

Le client sélectionne le groupe de messages par lequel les transmissions associées à cette connexion de messagerie explicite s'effectueront. Si le serveur ne peut pas satisfaire à cette demande, il doit alors la rejeter et renvoyer une réponse d'erreur;

- **ID de message source:** l'utilisation de ce champ dépend de la valeur dans le champ de sélection de groupe (voir tableau 3).

Tableau 3 – ID de message source dans une demande d'ouverture de connexion de messagerie explicite

Si la sélection de groupe est égale à:	Alors l'ID de message source
0 ou 3	Spécifie l'ID de message alloué par le client à partir du pool d'ID de messages du groupe 1 ou 3. Le client utilisera cet ID de message en combinaison avec son propre MAC ID pour générer l'ID de connexion spécifique lorsqu'il transmet un message via cette connexion ¹⁾
1	Est ignoré/fixé à la valeur zéro (0) ²⁾
¹⁾ Le client place cette valeur dans l'élément ID de message de l'identificateur des groupes 1 ou 3. ²⁾ Les connexions de messagerie explicite établies via le groupe de messages 2 imposent au serveur d'allouer deux ID de message du groupe 2 et de les renvoyer dans le message réponse d'ouverture de connexion de messagerie explicite. Le client doit utiliser l'un de ces ID de message pour générer l'ID de connexion qu'il spécifie lors de la transmission d'un message via cette connexion. L'autre ID sera utilisé par le serveur pour générer l'ID de connexion qu'il spécifie lors de la transmission d'un message via cette connexion.	

L'UCMM du serveur valide les arguments d'ouverture de connexion de messagerie explicite. S'ils sont valides, l'UCMM fait appel au service create (création) de la classe connexion pour obtenir une instance d'objet connexion (voir 5.3.2.6). L'objet connexion qui en résulte est automatiquement configuré pour être un objet connexion de messagerie explicite.

Si le serveur accepte plusieurs formats de corps de message et si le client a demandé l'un de ces formats, le serveur renvoie le format de corps de message demandé dans la réponse d'ouverture de connexion explicite. Si le serveur n'accepte pas plusieurs formats de corps de message, il précise son format par défaut dans la réponse d'ouverture de connexion de messagerie explicite.

Si aucune erreur n'est détectée, une réponse positive d'ouverture de connexion de messagerie explicite est renvoyée. Si une erreur est détectée, un message réponse d'erreur est renvoyé.

1.2.1.5.3 Réponse positive d'ouverture de connexion de messagerie explicite

Ce service est utilisé pour répondre à un message couronné de succès de demande d'ouverture de connexion de messagerie explicite (voir figure 9).

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 ¹⁾
1	Is ignored/set to the value zero (0) ²⁾
¹⁾ The client places this value within the message ID component of the message group 1 or 3 Identifier. ²⁾ 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 5.3.2.6). 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 echoes the requested message body format in the open explicit response message. If the server does not support multiple message body formats, then the server specifies its default format in the open explicit messaging connection response.

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

Contenu								
Rang de l'octet	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (1)	Code de service (4B _{hex})						
2	Réservé (tous les bits = 0)				Format réel du corps du message			
3	ID de message de destination				ID de message source			
4	octet poids faible							
5	octet poids fort							

En-tête du message

Corps du message

En-tête du message

Corps du message

Figure 9 – Format de la réponse d'ouverture de connexion de messagerie explicite

Contenu de la réponse d'ouverture de connexion de messagerie explicite:

- **frag (0)/ ID de transaction/MAC ID:** voir 5.2.1.2. Le MAC ID de destination est toujours spécifié dans l'en-tête du message associé à une demande/réponse d'ouverture de connexion de messagerie explicite;
- **bit R/R (1):** indique qu'il s'agit d'un message de réponse;
- **code de service (4B_{hex}):** identifie ce message en tant que service d'ouverture de connexion de messagerie explicite;
- **bits réservés:** ils doivent être mis à zéro par l'émetteur;
- **format réel du corps du message:** champ utilisé par le serveur pour définir le format de corps de message applicable aux messages explicites transmis ultérieurement via cette connexion (comme décrit dans le tableau 1);
- **ID de message de destination:** l'utilisation de ce champ dépend du groupe de message via lequel la connexion s'effectue (comme décrit dans le tableau 4);

Tableau 4 – ID de message de destination dans une réponse d'ouverture de connexion de messagerie explicite

Si la sélection de groupe dans la demande d'ouverture était:	L'ID de message de destination de la réponse ouverte:
0 ou 3	Est ignoré et doit être mis à la valeur zéro (0).
1	Est utilisé par le client en combinaison avec le MAC ID du serveur pour générer l'ID de connexion qu'il spécifie lors de transmissions via cette connexion

- **ID de message source:** valeur de l'ID de message alloué par le serveur. Le serveur alloue un ID de message à partir de son pool d'ID de messages du groupe 1, 2 ou 3. Cet ID de message est utilisé conjointement avec son propre MAC ID (MAC ID source) pour générer l'ID de connexion spécifié lorsqu'il transmet un message via cette connexion;
- **ID d'instance de connexion:** le serveur crée un objet connexion de messagerie explicite lorsqu'il traite avec succès une demande d'ouverture. Ce champ contient la valeur d'ID d'instance (nombre entier de 16 bits) attribuée à cet objet connexion de messagerie explicite.

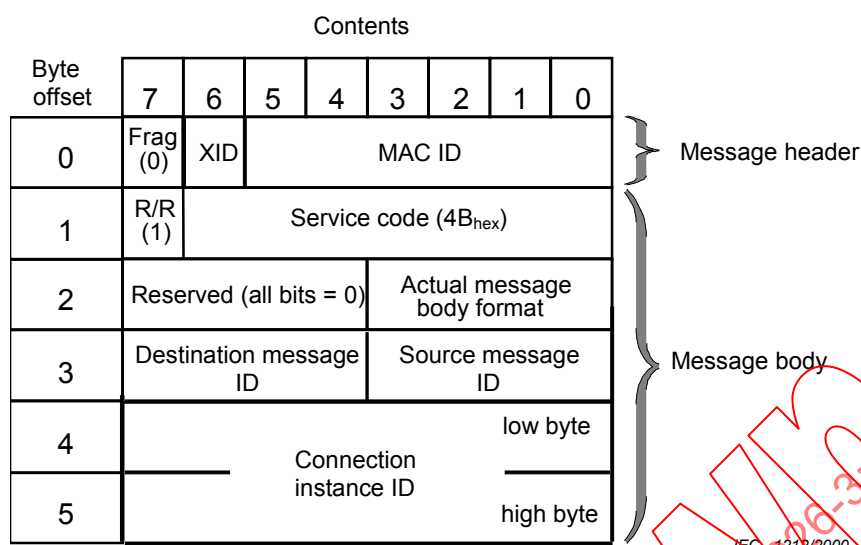


Figure 9 – Open explicit messaging connection response format

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 (4B_{hex}):** 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 an 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.

1.2.1.5.4 Demande de fermeture de connexion

Ce service est utilisé pour mettre fin à une connexion (d'E/S ou de messagerie) à l'une de ses extrémités. La réception du message de demande de fermeture de connexion par l'UCMM provoque l'exécution du service delete (suppression) de la classe de connexion (voir 1.3.2.6). Un message de demande de fermeture de connexion est transmis en tant que message de demande non connecté.

Le répondeur vérifie que l'instance de connexion spécifiée existe. Si elle existe et si elle peut être supprimée, elle est alors supprimée. Toutes les ressources associées à l'instance de connexion sont libérées. Si la demande est exécutée avec succès, une réponse de fermeture de connexion est renvoyée. Si la demande échoue, une réponse d'erreur est renvoyée.

Contenu de la demande de fermeture de connexion (voir figure 10):

- **frag (0)/ ID de transaction/MAC ID:** voir 1.2.1.2;
- **bit R/R (0):** indique qu'il s'agit d'un message de demande;
- **code de service (4C_{hex}):** identifie qu'il s'agit du service de fermeture de connexion;
- **ID d'instance de connexion:** champ qui spécifie l'instance de connexion à supprimer. Le format de l'ID d'instance de connexion est toujours spécifié dans ce message en tant que nombre entier de 16 bits.

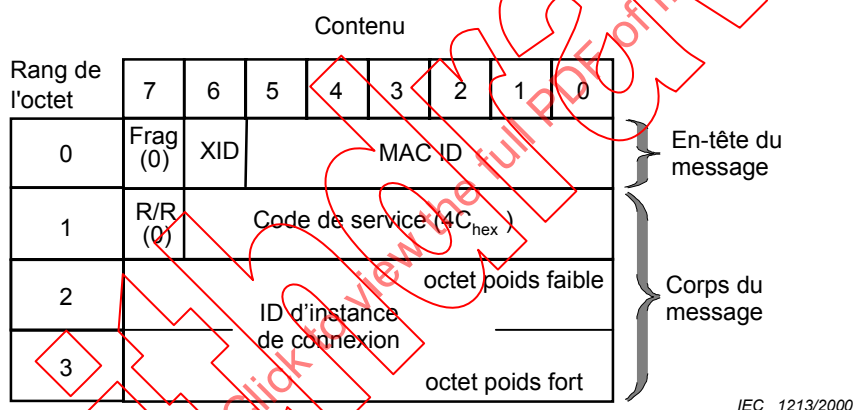


Figure 10 – Format de la demande de fermeture de connexion

1.2.1.5.5 Réponse de fermeture de connexion

Ce service est utilisé pour répondre à un message couronné de succès de demande de fermeture de connexion.

Contenu du format de la réponse de fermeture de connexion (voir figure 11):

- **frag (0)/ID de transaction/MAC ID:** voir 1.2.1.2;
- **bit R/R (1):** indique qu'il s'agit d'un message de réponse;
- **code de service (4C_{hex}):** identifie qu'il s'agit du service de fermeture de connexion.

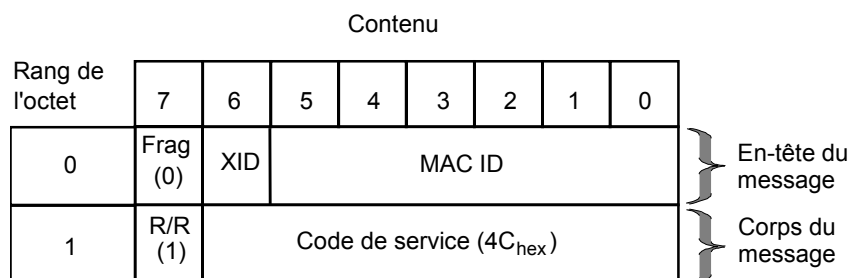


Figure 11 – Format de la réponse de fermeture de connexion

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 5.3.2.6). 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 can be deleted, then it is deleted. All resources associated with the connection instance are freed. If the request is successfully serviced, then a close connection response is returned. If the request is not successful, an error response is 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 (4C_{hex}):** 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.

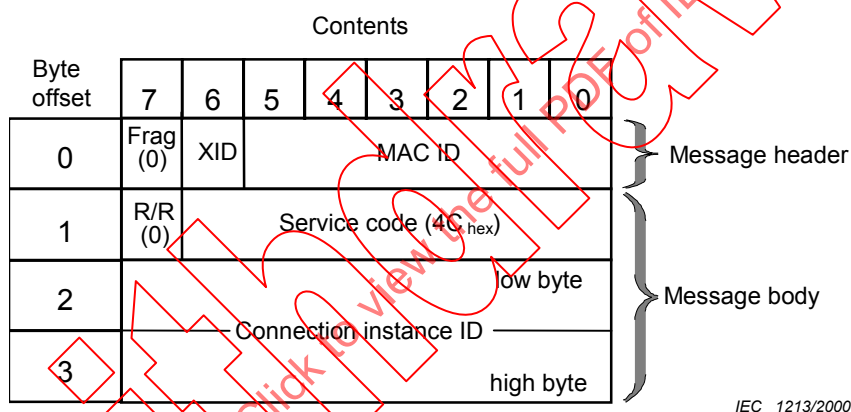


Figure 10 – Close connection request format

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 (4C_{hex}):** identifies this message as a close connection service.

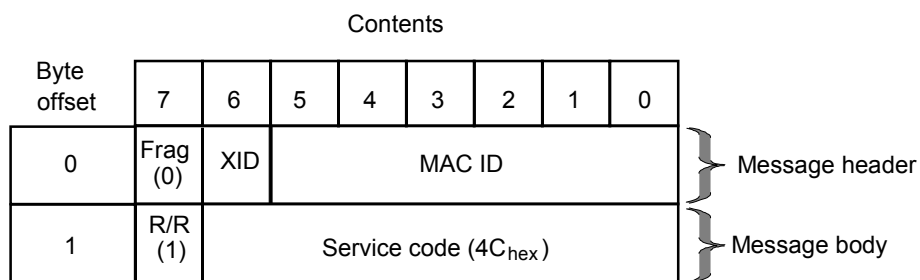


Figure 11 – Close connection response format

1.2.1.5.6 Réponse d'erreur

Le tableau 5 représente un ensemble standard de conditions d'erreur liées à l'UCMM et les codes d'erreurs correspondants à utiliser dans les messages de réponse d'erreur (codes d'erreurs généraux et supplémentaires). Le format d'un message de réponse d'erreur est décrit en 1.2.1.7.

Tableau 5 – Conditions/codes d'erreurs UCMM

Condition de l'erreur	Nom d'erreur général	Code d'erreurs général (hex)	Code d'erreurs supplémentaire (hex)
Code de service ni ouverture ni fermeture	Service non accepté	08	FF
Erreur de ressource (sélection de groupe)	Ressource non disponible	02	01
Sélection de groupe hors limites	Paramètre incorrect	20	01
Aucune connexion de serveur disponible	Ressource non disponible	02	02
Aucun ID de message serveur disponible	Ressource non disponible	02	03
ID de message source client incorrect	Paramètre incorrect	20	02
ID de message source client dupliqué	Ressource non disponible	02	04
ID d'instance de connexion incorrect	L'objet n'existe pas	16	FF

Description des conditions d'erreur:

- **service ni ouverture ni fermeture:** un service reçu via l'accès UCMM n'est ni de type ouverture ni de type fermeture et n'est donc pas accepté par l'UCMM;
- **erreur de ressource (sélection de groupe):** l'argument de sélection de groupe spécifie un groupe de messages qui n'est pas accepté par l'appareil;
- **sélection de groupe hors limites:** le champ de sélection de groupe contient une valeur incorrecte;
- **aucune connexion de serveur disponible:** le nombre maximal de connexions acceptées par ce serveur a déjà été atteint;
- **aucun ID de message serveur disponible:** le serveur a alloué tous les ID de message dans le groupe de messages demandé par le client;
- **ID de message source client incorrect:** l'ID de message source reçu dans une demande d'ouverture de connexion de messagerie explicite est incorrect pour le groupe de messages spécifié;
- **ID de message source client dupliqué:** l'ID de message source et le MAC ID reçus dans une demande d'ouverture de connexion de messagerie explicite sont déjà utilisés pour une connexion de messagerie explicite du groupe 1 ou 3;
- **ID d'instance de connexion incorrect:** l'ID d'instance de connexion reçu dans la demande de fermeture de connexion n'existe pas.

1.2.1.6 Messagerie explicite basée sur une connexion

Un message explicite basé sur une connexion est un message transmis via une connexion de messagerie explicite. Un message explicite basé sur une connexion doit se conformer aux formats décrits dans le présent paragraphe.

La figure 12 indique le format du message associé à une demande explicite non fragmentée.

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 5.2.1.7.

Table 5 – UCMM error conditions/codes

Error condition	General error name	General error code (hex)	Additional error code (hex)
Service code not open or close	Service not supported	08	FF
Group select resource error	Resource unavailable	02	01
Group select out of range	Invalid parameter	20	01
No server connections available	Resource unavailable	02	02
No server message IDs available	Resource unavailable	02	03
Client source message ID invalid	Invalid parameter	20	02
Duplicate client source message ID	Resource unavailable	02	04
Connection instance ID invalid	Object does not exist	16	FF

Error condition description:

- **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 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

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.

Figure 12 shows the format for the message associated with a non-fragmented explicit request.

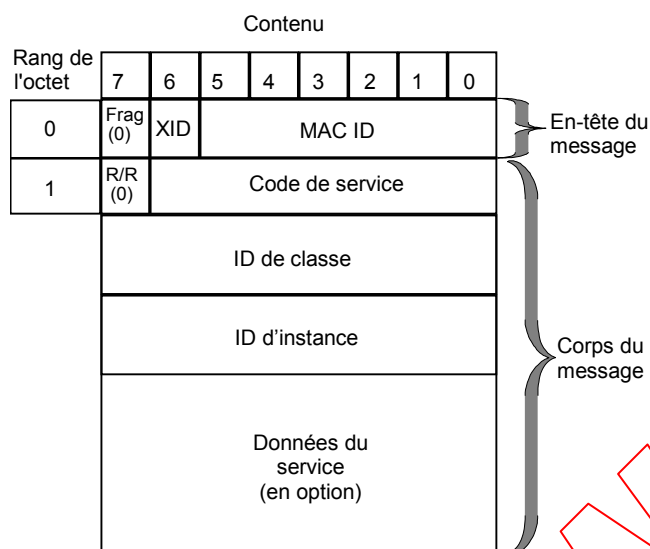


Figure 12 – Format d'un message de demande explicite non fragmenté

Contenu d'un message de demande explicite non fragmenté:

- **frag (0)/ ID de transaction/MAC ID:** voir 1.2.1.2;
- **bit R/R (0):** indique qu'il s'agit d'un message de demande;
- **code de service:** définit le service demandé;
- **ID de classe:** définit la classe d'objet vers laquelle cette demande est dirigée. L'ID de classe est spécifié par un nombre entier de 8 ou 16 bits en fonction de la valeur effective du format de corps de message renvoyée dans la réponse d'ouverture de connexion de messagerie explicite;
- **ID d'instance:** définit l'instance particulière dans la classe d'objet vers laquelle cette demande est dirigée. L'ID d'instance est spécifié par un nombre entier de 8 ou 16 bits en fonction de la valeur effective de format renvoyée dans la réponse d'ouverture de connexion de messagerie explicite. DeviceNet utilise la valeur zéro pour indiquer que la demande est dirigée vers la classe elle-même plutôt que vers une instance particulière de la classe;
- **données du service:** ce champ contient les données spécifiques à la demande. Pour les services communs de DeviceNet, les formats sont décrits à l'annexe A. Les définitions des services spécifiques à une classe et à un objet comprennent le format particulier de ce champ.

La figure 13 indique le format du message associé à une réponse positive non fragmentée:

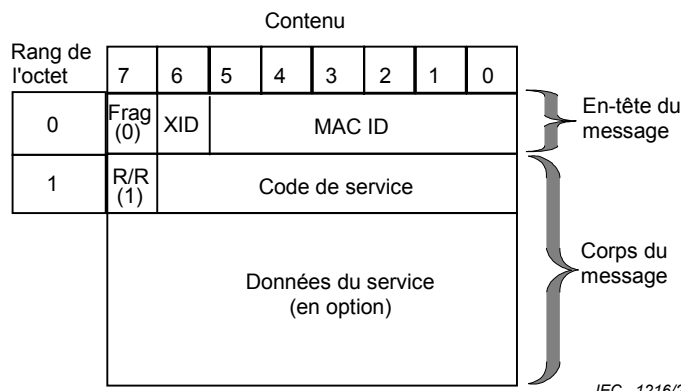


Figure 13 – Format d'un message réponse positif non fragmenté

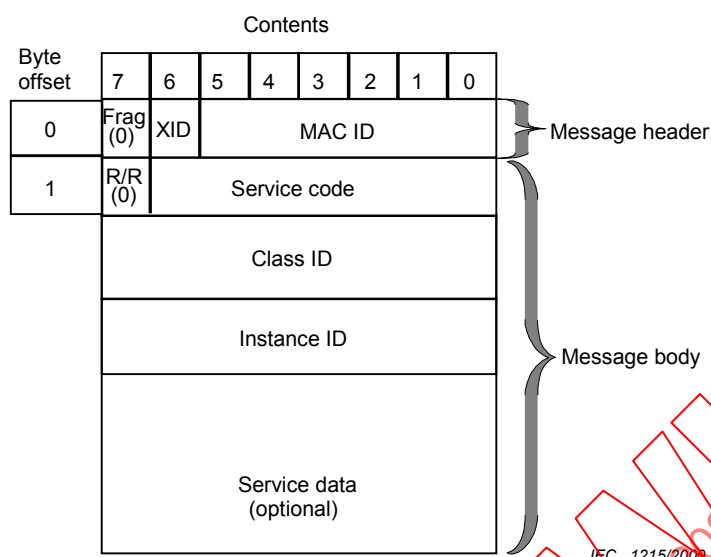


Figure 12 – Non-fragmented explicit request message format

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- 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- 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 annex A. Class-specific and object-specific service definitions include the format of this field.

Figure 13 shows the format for the message associated with a non-fragmented success response:

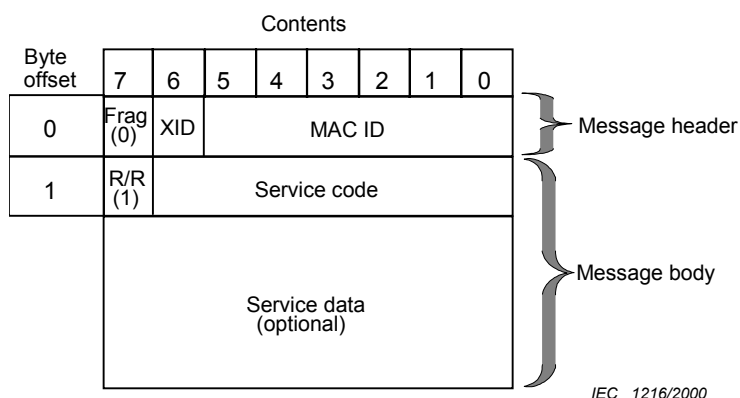


Figure 13 – Non-fragmented success response message format

Contenu d'un message réponse positif non fragmenté:

- **frag (0)/ ID de transaction/MAC ID:** voir 1.2.1.2;
- **bit R/R (1):** indique qu'il s'agit d'un message de réponse;
- **code de service:** contient le code de service envoyé dans le message de demande;
- **données du service:** ce champ contient les données spécifiques à la demande.

1.2.1.7 Message réponse d'erreur

Le message réponse d'erreur est renvoyé lors de la survenue d'une erreur au cours de l'exécution d'un service demandé via un message explicite reçu précédemment. La réponse d'erreur peut être envoyée soit comme message réponse basé sur une connexion (demande reçue via une connexion de messagerie explicite), soit comme message réponse non connecté (la demande était un message de demande explicite non connecté). La figure 14 indique le format d'un message réponse d'erreur.

Contenu								
Rang de l'octet	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (1)	Code de service (14 _{hex})						
2	Code d'erreurs general							
3	Code supplementaire							

IEC 1217/2000

IEC 1217/2000

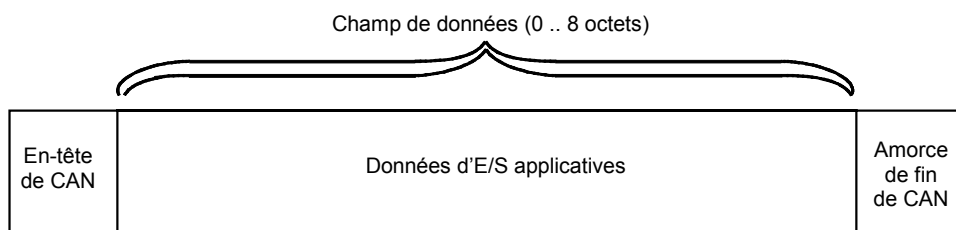
Figure 14 – Message réponse d'erreur

Contenu d'un message réponse d'erreur:

- **frag (0)/ ID de transaction/MAC ID:** voir 1.2.1.2;
- **bit R/R (1):** indique qu'il s'agit d'un message de réponse;
- **code de service (14_{hex}):** identifie ce message en tant que réponse d'erreur;
- **code d'erreur général:** identifie l'erreur rencontrée. La liste des codes d'erreurs généraux est donnée à l'annexe B;
- **code supplémentaire:** contient une valeur spécifique à l'objet ou au service qui précise la condition d'erreur. Si l'objet qui répond n'a pas d'informations supplémentaires à fournir, la valeur FF_{hex} est alors placée dans ce champ.

1.2.2 Messagerie d'E/S

DeviceNet définit un protocole de fragmentation pour le transfert de messages d'E/S d'une longueur supérieure à huit octets. Ce sont les seules informations de protocole contenues dans le champ de données d'un message d'E/S (voir figure 15).



IEC 1218/2000

Figure 15 – Champ de données d'un message d'E/S

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.

5.2.1.7 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 14 shows the format of an error response message.

Byte offset	Contents							
	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (1)	Service code (14 _{hex})						
2			General error code					
3			Additional code					

IEC 1217/2000

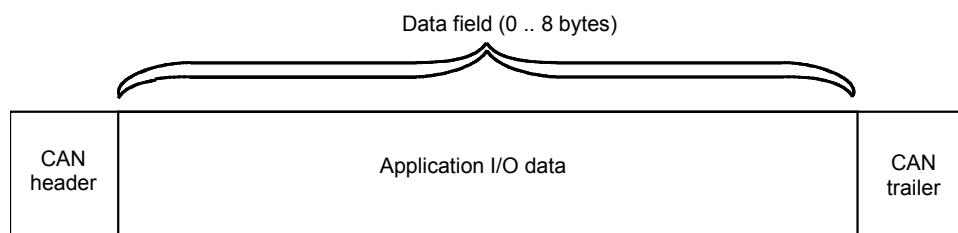
Figure 14 – Error response message

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 (14_{hex}):** identifies this message as an error response;
- **general error code:** identifies the encountered error. See annex B 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 FF_{hex} is 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 15).



IEC 1218/2000

Figure 15 – Data field of an I/O message

1.2.3 Fragmentation/réassemblage

1.2.3.1 Généralités

Le présent paragraphe définit le moyen par lequel un message dont la longueur est supérieure à huit octets est fragmenté et réassemblé. La fonction de fragmentation/réassemblage est fournie par l'objet connexion de DeviceNet.

La logique qui déclenche une transmission fragmentée est la suivante:

- les instances de l'objet connexion de messagerie examinent la longueur de chaque message à transmettre. Si le message est d'une longueur supérieure à huit octets, le protocole de fragmentation est utilisé;
- les instances de l'objet connexion d'E/S examinent l'attribut `produced_connection_size` (taille de connexion produite) de l'objet connexion (voir 5.3.2). Si cet attribut est supérieur à huit, le protocole de fragmentation est utilisé.

Deux types de fragmentation sont définis:

- **acquittée**: utilisé lors de la fragmentation d'un message explicite;
- **non acquittée**: utilisé lors de la fragmentation d'un message d'E/S.

1.2.3.2 Protocole de fragmentation

Le protocole de fragmentation est situé dans un seul octet du champ de données de CAN; il est formaté comme indiqué à la figure 16.

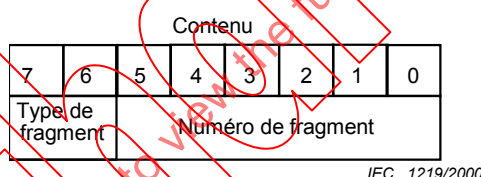


Figure 16 – Format du protocole de fragmentation DeviceNet

Contenu du protocole de fragmentation:

- **type de fragment**: indique s'il s'agit du premier fragment, d'un fragment du milieu ou du dernier fragment (voir tableau 6);

Tableau 6 – Valeurs des bits du type de fragment

Valeur	Signification
0	Premier fragment. Le champ numéro de fragment doit contenir la valeur 0 ou 3F ¹⁾
1	Fragment du milieu ²⁾
2	Dernier fragment ³⁾
3	Acquittement de fragment ⁴⁾

¹⁾ Si le numéro de fragment contient la valeur zéro (0), il s'agit du premier d'une série de fragments. Si le champ numéro de fragment contient la valeur 3F, il s'agit également de la dernière transmission de la série.
²⁾ Ce fragment n'est ni le premier ni le dernier de la série.
³⁾ Indique que ce fragment est le dernier.
⁴⁾ Valeur utilisée par le récepteur d'un message fragmenté pour acquitter la réception d'un fragment.

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 is 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 is 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 16.

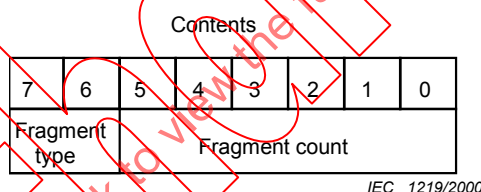


Figure 16 – Format of DeviceNet fragmentation protocol

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 3F ¹⁾
1	Middle fragment ²⁾
2	Last fragment ³⁾
3	Fragment acknowledgement ⁴⁾
¹⁾ 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 3F, then this is also the last transmission in the series. ²⁾ This fragment is neither the first nor the last fragment in the series. ³⁾ Marks this as the last fragment. ⁴⁾ The value the receiver of a fragmented message uses to acknowledge the reception of a fragment.	

- **numéro de fragment:** marque chaque fragment individuel de sorte que le récepteur peut déterminer si un fragment manque. Si le type de fragment indique qu'il s'agit du premier fragment, ce champ a une signification spéciale (comme décrit au tableau 6). Le numéro de fragment est augmenté de un à chaque fragment successif d'une série; il se remet à zéro en cas de dépassement (numéro de fragment = (numéro de fragment + 1) mod 64).

Pour la fragmentation d'un message d'E/S, les informations de protocole de fragmentation sont placées dans l'octet de rang 0 (voir figure 17).

		Contenu							
Rang de l'octet		7	6	5	4	3	2	1	0
0	Type de fragment	Numéro de fragment							
		Fragment de message d'E/S							

IEC 1220/2000

Figure 17 – Format d'un fragment de message d'E/S

Pour une fragmentation de message explicite, les informations de protocole de fragmentation sont placées dans l'octet de rang 1 (voir figure 18).

		Contenu							
Rang de l'octet		7	6	5	4	3	2	1	0
0	Frag (1)	XID	MAC ID						
1	Type de fragment	Numéro de fragment							
		Fragment du corps d'un message explicite							

IEC 1221/2000

Figure 18 – Format d'un fragment de message explicite

Le bit de fragmentation se trouvant dans l'en-tête de message de la figure 18 est mis à 1 pour indiquer qu'il s'agit d'un fragment du message explicite et non de tout le message. Dans ce cas, la valeur 1 indique également que l'octet suivant contient le protocole de fragmentation.

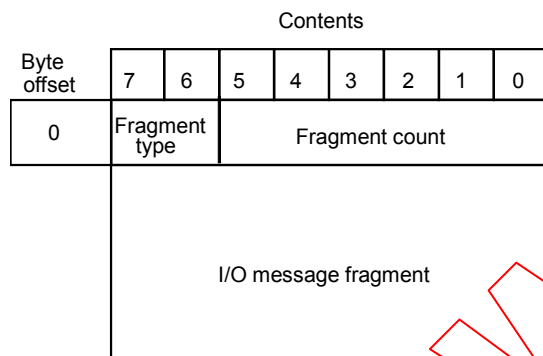
Le récepteur d'une transmission fragmentée analyse le message fragmenté comme défini dans le présent paragraphe. Cette procédure s'applique aux deux types de fragmentation: messages d'E/S et messages explicites.

Si la connexion est en attente de la première transmission et si le type de fragment reçu est égal au premier fragment:

- si le numéro de fragment est $3F_{\text{hex}}$, alors il s'agit de la seule transmission de la série; la connexion traite le message et attend le début d'une nouvelle série;
- si le numéro de fragment est 0, alors il s'agit de la première d'une série de transmissions; la connexion stocke le fragment et enregistre le numéro de fragment.

- **fragment count:** marks each separate fragment such that the receiver may determine whether or not a fragment has been missed. If the fragment type is the first fragment, then this field takes on a special meaning (as described in table 6). The fragment count is increased by one for each successive fragment in a series and resets to zero at overflow ($\text{fragmentcount} = (\text{fragmentcount} + 1) \bmod 64$).

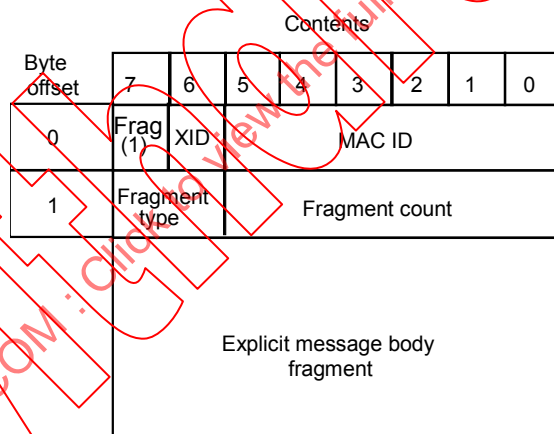
For I/O message fragmentation, the fragmentation protocol information is placed within byte offset 0 (see figure 17).



IEC 1220/2000

Figure 17 – I/O message fragment format

For explicit message fragmentation, the fragmentation protocol information is placed within byte offset 1 (see figure 18).



IEC 1221/2000

Figure 18 – Explicit message fragment format

The frag bit within the message header in figure 18 is set to 1 to indicate that this is a piece of the explicit message, not the entire message. The value 1 here also indicates that the next byte contains the fragmentation protocol.

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

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

- if the fragment count is $3F_{\text{hex}}$, then this is the only transmission in the series, and the connection processes the message and awaits the beginning of a new series;
- if the fragment count is 0, then this is the first in a series of transmissions and the connection stores the fragment and saves the fragment count.

Si la connexion est en attente de la première transmission et si le type de fragment n'est pas égal au premier fragment ou si le numéro de fragment est différent de 0 ou de $3F_{\text{hex}}$, la connexion ignore la transmission et attend le début d'un nouveau message.

Si la connexion n'est pas en attente de la première transmission, la connexion vérifie alors:

- que le type de fragment n'est pas le premier fragment, et
- que le numéro de fragment est numériquement supérieur d'une unité (1) à la précédente valeur reçue.

Si l'une de ces vérifications échoue, alors une erreur a été détectée. Si les deux vérifications réussissent, le fragment est ajouté à celui ou ceux reçus antérieurement, et le type de fragment est analysé pour déterminer s'il faut attendre ou non d'autres fragments.

Si d'autres fragments sont sur le point d'arriver, alors la connexion enregistre le numéro de fragment reçu et attend le suivant. S'il s'agit de la dernière transmission de la série et qu'il ne s'est pas produit d'erreur, la connexion traite le message et se réinitialise en attente du début d'une nouvelle série.

Si un message explicite est fragmenté, le récepteur doit générer et transmettre un acquittement à la réception de chaque fragment.

Si une erreur est détectée, il y a lieu d'effectuer une correction d'erreur en fonction du type de message (d'E/S ou explicite).

Si la détection d'un fragment manquant a été déclenchée par la réception du premier fragment de la série suivante, tout traitement associé à la série en cours est alors immédiatement arrêté, les fragments stockés en mémoire sont ignorés et le traitement commence immédiatement sur la nouvelle série.

1.2.3.3 Fragmentation non acquittée

La fragmentation d'un message d'E/S s'effectue de façon non acquittée. La fragmentation non acquittée consiste en une transmission des fragments l'un après l'autre depuis l'unité de transmission. La ou les unités de réception ne renvoient pas d'acquittement.

Lorsque le service `send_message` (envoyer message) d'une connexion d'E/S est invoqué, celle-ci examine son attribut `connection_size` (taille de la connexion) afin de déterminer s'il faut transmettre une série de messages fragmentés. Si l'attribut `connection_size` est supérieur à huit (8), le protocole de fragmentation est alors placé dans le message d'E/S.

Si l'attribut `connection_size` est inférieur ou égal à huit (8) octets, les données brutes sont transmises sans la présence du protocole de fragmentation.

Si l'application demande la transmission d'un bloc de données dont la longueur est supérieure à l'attribut `connection_size`, une erreur est indiquée en interne et la transmission n'a pas lieu. Si l'objet connexion d'E/S récepteur détecte un fragment manquant, c'est-à-dire si le numéro de fragment reçu n'est pas égal à celui reçu antérieurement plus un (1), les mesures de correction d'erreur suivantes sont entreprises:

- tous les fragments ultérieurs de cette suite sont ignorés et l'application n'est pas informée de la réception d'un message d'E/S;
- l'objet connexion attend le début d'une nouvelle série de transmissions fragmentées et ignore les fragments restants.

If the first transmission is expected by the connection and either the fragment type is not equal to the first fragment or the fragment count is not equal to 0 or $3F_{\text{hex}}$, the connection discards the transmission and awaits the beginning of a new message.

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

- that the fragment type is not the first fragment, and
- that the fragment count is numerically one (1) greater than the previous value received.

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

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

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

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

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

5.2.3.3 Unacknowledged fragmentation

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

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

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

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

- all subsequent fragments in this series are dropped and the application is not informed of an I/O message reception;
- the connection object begins looking for the beginning of a new fragmented series of transmissions and discards the remaining fragments in this series.

1.2.3.4 Fragmentation acquittée

La fragmentation acquittée est utilisée pour les messages explicites fragmentés. La fragmentation acquittée comprend la transmission d'un fragment provenant de la station émettrice, suivie de la transmission d'un acquittement par la station réceptrice. La station réceptrice acquitte la réception de chaque fragment.

La figure 19 indique le format du message d'acquittement transmis par le récepteur après la réception de chaque fragment de message explicite.

Rang de l'octet	Contenu							
	7	6	5	4	3	2	1	0
0	Frag (1)	XID	MAC ID					
1	Type (3) de fragment		Numéro de fragment					
2	Etat d'acquittement							

IEC 1222/2000

Figure 19 – Format d'un message d'acquittement

Contenu d'un message d'acquittement:

- **type de fragment:** indique qu'il s'agit d'un acquittement de fragment en plaçant la valeur 3 dans ce champ;
- **numéro de fragment:** retourne la dernière valeur du numéro de fragment reçu;
- **état d'acquittement:** indique si une erreur a été détectée ou non par le récepteur du message fragmenté. Les valeurs du bit d'état d'acquittement sont définies dans le tableau 7.

Tableau 7 – Valeurs du bit d'état d'acquittement

Valeur	Signification
0	Réussite. Aucune erreur n'a été détectée et la transmission fragmentée doit continuer.
1	Trop de données. La quantité maximale de données pouvant être reçues par le récepteur via cette connexion a été dépassée.
2 – 255	Réservé

La station de transmission fonctionne de la manière suivante:

- l'objet connexion construit l'en-tête du message en activant le bit de fragmentation (mise à 1), en plaçant dans le champ XID (ID de transaction) la valeur spécifiée par l'application, et initialise le champ MAC ID de la même manière que pour un message explicite non fragmenté. Les en-têtes de message associés à chaque fragment individuel sont identiques;
- l'objet connexion place ensuite les informations de protocole de fragmentation appropriées dans le message. Il stocke le numéro de fragment inséré dans le message;
- puis l'objet connexion prend la partie de corps de message suivante et la place dans le message;
- le message est transmis et attend que le temporisateur d'acquittement démarre. Le délai d'attente dépend de l'application;
- si le délai d'attente du temporisateur d'acquittement expire, l'objet connexion relance automatiquement la dernière transmission. Il essaie encore une fois. Si le temporisateur expire une nouvelle fois (délai d'attente expiré par deux fois consécutives), l'application est informée de la survenue d'une erreur et de l'impossibilité d'effectuer la transmission demandée;

5.2.3.4 Acknowledged fragmentation

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

Figure 19 shows the format for the acknowledgement message transmitted by the receiver after each explicit message fragment is received.

Byte offset	Contents							
	7	6	5	4	3	2	1	0
0	Frag (1)	XID	MAC ID					
1	Fragment type (3)		Fragment count					
2	Ack status							

Figure 19 – Acknowledgement message format

Acknowledgement message contents:

- **fragment type:** indicates that this is a fragment acknowledgement by placing the value 3 within this field;
- **fragment count:** echoes the last fragment count value received;
- **ack status:** indicates whether or not an error has been detected by the receiver of the fragmented message. The ack status bit values are defined in table 7.

Table 7 – Ack status bit values

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

The transmitting node functions as specified below:

- the connection object formulates the message header by setting the frag bit to one (1), the XID (transaction ID) field to the value specified by the application, and initialises the MAC ID field in the same manner it would a non-fragmented explicit message. The message headers associated with each separate fragment are identical;
- the connection object then places the appropriate fragmentation protocol information into the message. The connection object stores the fragment count that was inserted into the message;
- the connection object then takes the next piece of the message body and places it into the message;
- the message is transmitted and a wait for the acknowledgement timer is started. The amount of time to wait is application-specific;
- if the wait for the acknowledgement timer expires, then the connection object automatically retries the last transmission. The connection object retries once. If the timer expires once again (two consecutive wait for acknowledgement time-outs), then the application is informed that an error has been detected and the requested transmission cannot take place;

- si un acquittement est reçu, l'objet connexion détermine si le numéro de fragment de l'acquittement est égal ou non au dernier numéro transmis. S'il s'agit du même numéro, le fragment a bien été transmis et acquitté, et le traitement normal continue. Si les valeurs sont différentes, l'objet connexion continue à attendre l'acquittement du numéro de fragment correspondant.

L'état initial associé à l'unité de réception lui impose d'attendre soit le premier fragment d'une transmission fragmentée, soit la réception d'un message explicite complet. Le côté réception fonctionne de la manière suivante:

- si l'en-tête de message indique qu'il s'agit d'une partie fragmentée d'un message explicite, l'objet connexion examine le protocole de fragmentation afin de déterminer sa validité. Si la connexion n'a pas encore reçu la première transmission de la série (état initial) et si le champ type de fragment n'indique pas premier fragment, le fragment est alors ignoré et aucun acquittement n'est renvoyé;
- si le type de fragment indique premier fragment, le numéro de fragment doit être égal à zéro (0). Si c'est le cas, le fragment de message est enregistré et un acquittement est renvoyé. S'il s'agit du premier fragment et si le numéro de fragment n'est pas égal à zéro (0), le fragment est alors ignoré et aucun acquittement n'est renvoyé;
- si le numéro de fragment est supérieur de un (1) au numéro précédemment reçu et si le type de fragment n'indique pas premier fragment, alors le fragment suivant a été reçu. Ce fragment est ajouté à celui ou ceux reçus antérieurement et un acquittement est renvoyé. Le numéro de fragment est enregistré;
- si le numéro de fragment n'est ni supérieur de un (1), ni égal au numéro reçu antérieurement, le fragment est ignoré et aucun acquittement n'est renvoyé. Le récepteur se remet à l'état initial;
- lorsque le dernier fragment est reçu et l'acquittement transmis, l'objet connexion continue le traitement du message comme s'il n'était pas fragmenté.

1.2.4 Protocole de détection de duplication d'adresses MAC ID

Chaque station DeviceNet doit recevoir un MAC ID et est tenue de participer à l'algorithme de détection de duplication d'adresses MAC ID, comme indiqué en 1.4.

Une valeur spéciale d'ID de message est définie dans le groupe 2 pour identifier le message de vérification de duplication d'adresses MAC ID (voir figure 20).

BITS DE L'IDENTIFICATEUR											SIGNIFICATION DE L'ID DE MESSAGE
10	9	8	7	6	5	4	3	2	1	0	
1	0	MAC ID						ID message groupe 2			Messages du groupe 2
0	0	MAC ID de destination						1	1	1	Message de vérification de duplication de MAC ID

IEC 1223/2000

Figure 20 – Champ identificateur de CAN pour la vérification de duplication d'adresses MAC ID

Le champ de données associé au message de vérification de duplication d'adresses MAC ID est au format indiqué à la figure 21.

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

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

- if the message header indicates that this is a fragmented portion of an explicit message, then the connection object examines the fragmentation protocol to determine its validity. If the connection has yet to receive the first transmission in the series (in the initial state) and the fragment type field is not equal to the first fragment, then the fragment is dropped and no acknowledgement is returned;
- if the fragment type indicates that this is the first fragment, then the fragment count shall equal zero (0). If this is the case, then the message fragment is stored and an acknowledgement is returned. If the fragment type indicates first fragment and the fragment count is not zero (0), then the fragment is dropped and no acknowledgement is returned;
- if the fragment count is one (1) greater than the previously received count and the fragment type does not indicate that this is the first fragment, then the next fragment has been received. The fragment is appended to the previously received fragment(s) and an acknowledgement is returned. The fragment count associated with this fragment is stored;
- if the fragment count is neither one (1) greater nor equal to the previously received count, then the fragment is discarded and no acknowledgement is returned. The receiver resets to the initial state;
- when the final fragment is received and the acknowledgement is transmitted, the connection object continues processing the message as if it were non-fragmented.

5.2.4 Duplicate MAC ID detection protocol

Each DeviceNet node shall be assigned a MAC ID and is required to participate in the duplicate MAC ID detection algorithm, as defined in 5.4.

A special message ID value is defined within group 2 to identify the duplicate MAC ID check message (see figure 20).

IDENTIFIER BITS										MESSAGE ID MEANING
10	9	8	7	6	5	4	3	2	1	
1	0	MAC ID					Group 2 message ID			Group 2 messages
1	0	Destination MAC ID					1	1	1	Duplicate MAC ID check message

IEC 1223/2000

Figure 20 – Duplicate MAC ID check CAN identifier field

The data field associated with the duplicate MAC ID check message has the format shown in figure 21.

Rang de l'octet	Contenu							
	7	6	5	4	3	2	1	0
0	R/R	Numéro d'accès physique						
1	octet poids faible							
2	ID de fabricant							
3	octet poids fort							
4	octet poids faible							
5	Numéro de série							
6	octet poids fort							

Figure 21 – Format du champ de données d'un message de vérification de duplication d'adresses MAC ID

Contenu du champ de données d'un message de vérification de duplication d'adresses MAC ID:

- **bit R/R:** bit de demande/réponse. La valeur de ce champ indique s'il s'agit d'une demande de vérification de duplication d'adresses MAC ID ou d'un message réponse:
0 = demande
1 = réponse
- **numéro de l'accès physique:** valeur d'identification attribuée en interne à chaque accès physique DeviceNet d'un appareil. Les appareils munis d'un seul accès doivent placer la valeur zéro (0) dans ce champ.
- **ID de fabricant:** nombre entier de 16 bits (UINT) contenant le code d'identification attribué au fabricant de l'appareil qui transmet le message;
- **numéro de série:** voir 3.42.

1.3 Classes d'objets de communication DeviceNet

1.3.1 Généralités

Les objets de communication DeviceNet assurent la gestion et la production des messages échangés. Les services, attributs et comportements associés aux principaux objets de communication sont détaillés dans le présent paragraphe. Voir l'annexe D pour une description des spécifications et du codage des types de données DeviceNet, et l'annexe E pour la définition des objets auxiliaires de communication.

1.3.2 Définition de la classe d'objet connexion (code d'ID de classe: 5)

1.3.2.1 Généralités

La classe connexion attribue et gère les ressources internes associées aux connexions d'E/S et de messagerie explicite. L'instance spécifique générée par la classe de connexion est appelée instance de connexion ou objet connexion.

1.3.2.2 Attribut de la classe d'objet connexion

L'attribut de la classe de connexion est défini dans le tableau 8.

Byte offset	Contents							
	7	6	5	4	3	2	1	0
0	R/R	Physical port number						
1	Manufacturer ID						low byte	
2							high byte	
3	Serial number						low byte	
4								
5								
6							high byte	

IEC 1224/2000

Figure 21 – Duplicate MAC ID check message data field format

Duplicate MAC ID check message data field contents:

- **R/R bit:** request/response bit. The value in this field indicates whether this is a duplicate MAC ID check request or a response message.
0 = request
1 = response
- **physical port number:** an identification value internally assigned to each physical DeviceNet port of a device. Devices that have a single port shall place the value zero (0) within this field;
- **manufacturer ID:** a 16-bit integer field (UINT) containing the identification code assigned to the manufacturer of the device that is transmitting the message;
- **serial number:** see 3.42.

5.3 DeviceNet communication object classes

5.3.1 General

The DeviceNet communication objects manage and provide the exchange of messages. The services, attributes and behaviours associated with the main communication objects are detailed in this subclause. See annex D for a description of DeviceNet data type specification and encoding, and annex E for the definition of auxiliary communication objects.

5.3.2 Connection object class definition (class ID code: 5)

5.3.2.1 General

The connection class allocates and manages the internal resources associated with both I/O and explicit messaging connections. The specific instance generated by the connection class is referred to as a connection instance or a connection object.

5.3.2.2 Connection object class attribute

The connection class attribute is defined in table 8.

Tableau 8 – Attribut de la classe de connexion

ID d'attribut	Mise en œuvre	Règle d'accès	Nom de l'attribut	Type de données	Description de l'attribut
1	En option	Get	Révision	UINT	Version de la définition de la classe de connexion utilisée pour la mise en œuvre dans l'appareil

L'attribut doit être fixé à 1.

Tous les autres attributs sont réservés.

1.3.2.3 Services de la classe d'objet connexion

La classe de connexion accepte les services communs suivants de DeviceNet (comme décrit dans le tableau 9).

Tableau 9 – Services de la classe de connexion

Mise en œuvre	Nom du service	Description du service
En option	Create	Création d'un objet connexion
En option	Delete	Supprime tous les objets connexion et libère toutes les ressources associées. Lorsque le service delete est envoyé à la classe de connexion, toutes les instances sont supprimées
En option	Reset	Remet à zéro tous les objets connexion pouvant l'être. Les conditions dans lesquelles une connexion peut être remise à zéro sont indiquées dans la table états-événements (tableau 20)
En option	Find_next_object_instance	Recherche les ID d'instance correspondant aux objets connexion existants
1)	Get_attribute_single	Lit la valeur d'un attribut de la classe de connexion
1) Get_attribute_single est requis si l'attribut de la classe de connexion est mis en œuvre.		

1.3.2.4 Attributs d'instances d'un objet connexion

Le tableau 10 indique les attributs des instances de connexion et les types de données correspondants.

Table 8 – Connection class attribute

Attribute ID	Need in implementation	Access rule	Attribute name	Data type	Attribute description
1	Optional	Get	Revision	UINT	Revision of the connection class definition upon which the implementation is based

The attribute shall be set to 1.

All other attributes are reserved.

5.3.2.3 Connection object class services

The connection class supports the following DeviceNet common services (as described in table 9).

Table 9 – Connection class services

Need in implementation	Service name	Service description
Optional	Create	Creates a connection object
Optional	Delete	Deletes all connection objects and releases all associated resources. When the delete service is sent to the connection class, all instances are deleted
Optional	Reset	Resets all resettable connection objects. The conditions under which a connection is resettable are listed in the state-event matrix in table 20
Optional	Find_next_object_instance	Searches for instance IDs associated with existing connection objects
1)	Get_attribute_single	Reads a connection class attribute value
1) Get_attribute_single is required if the connection class attribute is supported.		

5.3.2.4 Connection object instance attributes

Table 10 shows the connection instance attributes and their associated data types.

Tableau 10 – Attributs d'instances d'un objet connexion

ID d'attribut (décimal)	Mise en œuvre	Nom de l'attribut	Type de données	Description abrégée de l'attribut
1	Requis	state	USINT	Etat de l'objet
2	Requis	instance_type	USINT	Indique une connexion d'E/S ou de messagerie
3	Requis	transportclass_trigger	BYTE	Définit le comportement de la connexion
4	Requis	produced_connection_id	UINT	Placé dans le champ d'identificateur de CAN lorsque la connexion émet
5	Requis	consumed_connection_id	UINT	Valeur du champ d'identificateur de CAN pour le message à recevoir
6	Requis	initial_comm_characteristics	BYTE	Définit le ou les groupes de messages utilisés pour produire ou consommer via cette connexion
7	Requis	produced_connection_size	UINT	Nombre maximal d'octets émis via cette connexion
8	Requis	consumed_connection_size	UINT	Nombre maximal d'octets reçus via cette connexion
9	Requis	expected_packet_rate	UINT	Définit les temporisations associées à cette connexion
10-11				Réservé
12	Requis	watchdog_timeout_action	USINT	Définit comment gérer les expirations du délai d'inactivité ou de surveillance (chien de garde)
13	Requis	produced_connection_path_length	UINT	Nombre d'octets dans l'attribut produced_connection_path
14	Requis	produced_connection_path	Array of USINT	Spécifie le ou les objets applicatifs dont les données doivent être produites par cet objet connexion
15	Requis	consumed_connection_path_length	UINT	Nombre d'octets dans l'attribut consumed_connection_path
16	Requis	consumed_connection_path	Array of USINT	Spécifie le ou les objets applicatifs devant recevoir les données consommées par cet objet connexion
17	Requis	production_inhibit_time	UINT	Définit l'intervalle minimal entre les productions de nouvelles données

La liste ci-après donne une description détaillée des attributs des instances d'objets connexion DeviceNet. Les valeurs définies par défaut doivent être utilisées lorsqu'aucune autre règle interne et/ou définie par le système n'existe.

- 1) **state (état):** Définit l'état courant de l'instance de connexion. Le tableau 11 définit les états possibles et leur attribue une valeur, utilisée pour indiquer cet état.

Table 10 – Connection object instance attributes

Attribute ID (decimal)	Need in implementation	Attribute name	Data type	Brief description of attribute
1	Required	state	USINT	State of the object
2	Required	instance_type	USINT	Indicates either I/O or messaging connection
3	Required	transportclass_trigger	BYTE	Defines behaviour of the connection
4	Required	produced_connection_id	UINT	Placed in CAN identifier field when the connection transmits
5	Required	consumed_connection_id	UINT	CAN identifier field value that denotes message to be received
6	Required	initial_comm_characteristics	BYTE	Defines the message group(s) across which productions and consumptions associated with this connection occur
7	Required	produced_connection_size	UINT	Maximum number of bytes transmitted across this connection
8	Required	consumed_connection_size	UINT	Maximum number of bytes received across this connection
9	Required	expected_packet_rate	UINT	Defines timing associated with this connection
10 – 11				Reserved
12	Required	watchdog_timeout_action	USINT	Defines how to handle inactivity/watchdog time-outs
13	Required	produced_connection_path_length	UINT	Number of bytes in the produced_connection_path attribute
14	Required	produced_connection_path	Array of USINT	Specifies the application object(s) whose data is to be produced by this connection object
15	Required	consumed_connection_path_length	UINT	Number of bytes in the consumed_connection_path attribute
16	Required	consumed_connection_path	Array of USINT	Specifies the application object(s) that are to receive the data consumed by this connection object
17	Required	production_inhibit_time	UINT	Defines minimum time between new data production

The list below provides detailed descriptions of the DeviceNet connection object instance attributes. The defined default attribute values shall be used when no other internal and/or system-defined rules exist.

- 1) **state:** Defines the current state of the connection instance. Table 11 defines the possible states and assigns a value used to indicate that state.

Tableau 11 – Valeurs associées à l'attribut state (état)

Valeur	Nom de l'état	Description
0	Non existant	Aucune connexion n'existe
1	Configuration	La connexion a été créée et attend: (a) d'être configurée et (b) de recevoir l'ordre d'appliquer la configuration
2	Attente d'ID de connexion	L'instance de connexion attend l'établissement de l'attribut consumed_connection_id et/ou de l'attribut produced_connection_id ¹⁾
3	Etablie	La connexion a été configurée et la configuration a été appliquée
4	Délai expiré	Si un objet connexion subit une expiration du délai d'inactivité/de surveillance (chien de garde), une transition doit être faite vers cet état. Voir la description de l'attribut watchdog_timeout_action (tableau 17) et la description du temporisateur d'inactivité/de surveillance (1.3.2.5.3) pour plus de détails
5-255	Réservé	

¹⁾ Lorsque les attributs d'instance de connexion sont appliqués, il se peut que la station ne puisse pas générer les valeurs produced_connection_id et/ou consumed_connection_id (voir l'attribut initial_comm_characteristics au tableau 15 pour les règles). Si c'est le cas, toutes les tâches nécessaires sont accomplies, sauf l'initialisation de ces attributs dans la connexion et celle des objets producteur/consommateur associés, et l'instance de connexion se met dans l'état Attente d'ID de connexion.

Toutes les ressources associées à une instance de connexion créée dynamiquement via une connexion de messagerie explicite doivent être libérées si cette connexion de messagerie explicite se rompt avant la transition vers l'état «établie» de l'instance de connexion créée dynamiquement.

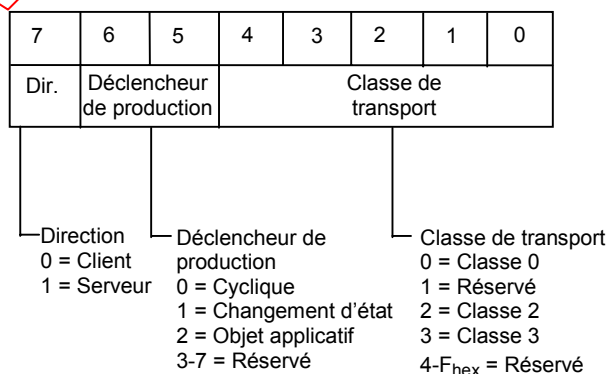
Lorsqu'une transition vers l'état «établie» est effectuée, tous les temporisateurs associés à l'objet connexion doivent être activés (voir 1.3.2.5).

2) **instance_type (type d'instance):** Définit le type d'instance (comme décrit dans le tableau 12).

Tableau 12 – Valeurs associées à l'attribut instance_type

Valeur	Signification
0	Messagerie explicite
1	E/S
2-255	Réservé

3) **transportclass_trigger (classe de transport et déclencheur):** Identifie s'il s'agit d'une connexion de production, de consommation, ou de production et consommation. Si cette extrémité de la connexion doit produire des données, cet attribut identifie également l'événement qui déclenche la production. Les huit bits sont répartis comme suit (voir figure 22).



IEC 1225/2000

Figure 22 – Transportclass_trigger

Table 11 – Values assigned to the state attribute

Value	State name	Description
0	Non-existent	No connection exists
1	Configuring	The connection has been created and is waiting: (a) to be configured and (b) to be instructed to apply the configuration
2	Waiting for connection ID	The connection instance is waiting for its consumed_connection_id and/or produced_connection_id attribute to be set ¹⁾
3	Established	The connection has been configured and the configuration has been applied
4	Timed-out	If a connection object experiences an inactivity/watchdog time-out, then a transition shall be made to this state. See the watchdog_timeout_action attribute description (table 17) and the description of the inactivity/watchdog timer (5.3.2.5.3) for more details
5-255	Reserved	

¹⁾ When the connection instance attributes are applied, it may not be possible for the node to generate the produced_connection_id and/or consumed_connection_id values (see initial_comm_characteristics attribute in table 15 for rules). If this is the case, then all the tasks required to apply the attributes are performed except for the initialisation of these attributes within the connection and associated link producer/consumer objects, and the connection instance assumes the Waiting for connection ID state

All resources associated with a connection instance that has been dynamically created across an explicit messaging connection shall be released if the explicit messaging connection breaks down before the dynamically created connection instance has transitioned to the established state.

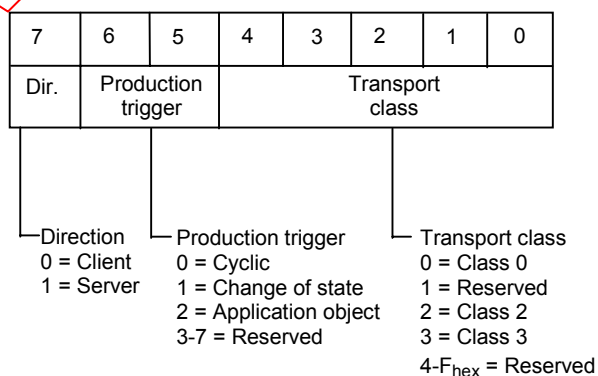
When a transition is made to the established state, all timers associated with the connection object shall be activated (see 5.3.2.5).

2) **instance_type**: Defines the instance type (as described in table 12).

Table 12 – Values assigned to the instance_type attribute

Value	Meaning
0	Explicit messaging
1	I/O
2-255	Reserved

3) **transportclass_trigger**: Identifies whether this is a producing only, consuming only, or both producing and consuming connection. If this end-point is to perform a data production, this attribute also identifies the event that triggers the production. The eight bits are divided as follows (see figure 22).



IEC 1225/2000

Figure 22 – Transportclass_trigger

- 4) **produced_connection_id**: Contient le CID à associer aux transmissions émises via cette connexion (le cas échéant). Cette valeur doit être spécifiée dans le champ de l'identificateur de CAN lorsque cette connexion émet (voir 5.1.2).

Tableau 13 – Valeurs définies pour l'attribut produced_connection_id

Valeur	Signification
0 – 7F0 _{hex}	Valeur à placer dans le champ de l'identificateur de CAN lorsque cette connexion émet
7F1 _{hex} – FFFE _{hex}	Réservé
FFFF _{hex}	Valeur par défaut associée à cet attribut dans une connexion d'E/S. L'attribut doit conserver cette valeur si cette instance de connexion ne produit aucune donnée (consommateur uniquement)

- 5) **consumed_connection_id**: Contient le CID qui identifie les messages à recevoir via cette connexion (le cas échéant). C'est la valeur du champ de l'identificateur de CAN utilisée pour les messages à recevoir via cet objet connexion (voir 5.1.2). Les valeurs suivantes sont définies (voir tableau 14).

Tableau 14 – Valeurs définies pour l'attribut consumed_connection_id

Valeur	Signification
0 – 7F0 _{hex}	Valeur qui identifie les messages à consommer par cette instance d'objet connexion. Cela doit être spécifié dans le champ identificateur de CAN des messages à consommer
7F1 _{hex} – FFFE _{hex}	Réservé
FFFF _{hex}	Valeur par défaut associée à cet attribut dans une connexion d'E/S. L'attribut doit conserver cette valeur si cette instance de connexion ne consomme aucune donnée (producteur uniquement)

- 6) **initial_comm_characteristics**: Définit le ou les groupes de messages via lesquels cette connexion produit et consomme. Cet octet est divisé en deux demi-octets (voir figure 23).

7	6	5	4	3	2	1	0
Caractéristiques initiales de production				Caractéristiques initiales de consommation			

IEC 1226/2000

Figure 23 – Format de l'attribut initial_comm_characteristics

Le tableau 15 donne la liste des valeurs possibles dans le demi-octet des caractéristiques initiales de production (demi-octet de poids fort) de l'attribut initial_comm_characteristics.

- 4) **produced_connection_id**: contains the CID to be associated with transmissions sent across this connection (if any). This value shall be specified in the CAN identifier field when this connection transmits (see 5.1.2).

Table 13 – Values defined for the produced_connection_id attribute

Value	Meaning
0 – 7F0 _{hex}	The value to be placed in the CAN identifier field when this connection transmits
7F1 _{hex} – FFFE _{hex}	Reserved
FFFF _{hex}	Default value assigned to this attribute within an I/O connection. This attribute shall retain this value if this connection instance is not producing any data (consumer only)

- 5) **consumed_connection_id**: Contains the CID which identifies messages to be received across this connection (if any). This is the CAN identifier field value that is used with messages to be received by this connection object (see 5.1.2). The following values are defined (see table 14).

Table 14 – Values defined for the consumed_connection_id attribute

Value	Meaning
0 – 7F0 _{hex}	The value that identifies messages to be consumed by this connection object instance. This shall be specified in the CAN identifier field of messages that are to be consumed
7F1 _{hex} – FFFE _{hex}	Reserved
FFFF _{hex}	Default value assigned to this attribute within an I/O connection. This attribute shall retain this value if this connection instance is not consuming any data (producer only)

- 6) **initial_comm_characteristics**: Defines the message group(s) across which this connection produces and consumes. This byte is divided into two nibbles (see figure 23).

7	6	5	4	3	2	1	0
Initial production characteristics				Initial consumption characteristics			

IEC 1226/2000

Figure 23 – Initial_comm_characteristics attribute format

Table 15 lists the values that are possible within the initial production characteristics nibble (upper nibble) of the initial_comm_characteristics attribute.

Tableau 15 – Valeurs du demi-octet des caractéristiques initiales de production

Valeur	Signification
0	Production via le groupe de messages 1 ¹⁾
1	Production via le groupe de messages 2 (destination) ²⁾
2	Production via le groupe de messages 2 (source) ³⁾
3	Production via le groupe de messages 3 ⁴⁾
4 – E _{hex}	Réservé
F _{hex}	Valeur par défaut ⁵⁾

¹⁾ La station productrice génère la valeur CID et la charge dans l'attribut produced_connection_id de l'objet connexion. Elle attribue un ID de message provenant de son pool d'ID de messages du groupe 1 et la combine avec son MAC ID source pour générer le CID. L'ID de message du groupe 1 ayant la plus petite valeur numérique disponible doit être utilisé pour générer la valeur de l'attribut produced_connection_id. Cette valeur doit également être chargée dans l'attribut consumed_connection_id correspondant associé à l'objet ou aux objets connexion consommateurs.
²⁾ Le MAC ID de destination doit être placé dans l'élément MAC ID du champ identificateur du groupe 2. Dans ce cas, la station qui consomme génère la valeur CID à associer aux transmissions via cette connexion. Lorsque la station qui consomme a généré cette valeur et l'a chargée dans l'attribut consumed_connection_id de l'objet connexion approprié, cette valeur doit être lue et ultérieurement chargée dans l'attribut produced_connection_id de l'objet connexion producteur.
³⁾ Le MAC ID source doit être placé dans l'élément MAC ID de l'identificateur du groupe 2. Dans ce cas, la station qui produit génère la valeur CID et la charge dans l'attribut produced_connection_id de l'objet connexion. L'ID de message du groupe 2 ayant la plus petite valeur numérique disponible doit être utilisé pour générer la valeur de l'attribut produced_connection_id. Cette valeur doit également être chargée dans l'attribut consumed_connection_id correspondant, associé à l'objet ou aux objets connexion consommateurs.
⁴⁾ La station qui produit génère la valeur CID et la charge dans l'attribut produced_connection_id de l'objet connexion. Elle attribue un ID de message provenant du pool d'ID de messages du groupe 3 et la combine avec son MAC ID source pour générer le CID. L'ID de message du groupe 3 ayant la plus petite valeur numérique disponible doit être utilisé pour générer la valeur de l'attribut produced_connection_id. Cette valeur doit également être chargée dans l'attribut consumed_connection_id correspondant, associé à l'objet ou aux objets connexion consommateurs.
⁵⁾ Valeur par défaut attribuée au demi-octet des caractéristiques initiales de production, dans une connexion d'E/S. S'il s'agit d'une connexion d'E/S de consommation seulement, ce demi-octet conserve la valeur par défaut. Les objets connexion de messagerie explicite configurent automatiquement cet attribut lorsque la connexion est établie.

Le tableau 16 donne la liste des valeurs possibles dans le demi-octet des caractéristiques initiales de consommation (demi-octet de poids faible) de l'attribut initial_comm_characteristics.

Tableau 16 – Valeurs du demi-octet des caractéristiques initiales de consommation

Valeur	Signification
0	Consomme un message du groupe 1 ¹⁾
1	Consomme un message du groupe 2 (destination) ²⁾
2	Consomme un message du groupe 2 (source) ³⁾
3	Consomme un message du groupe 3 ⁴⁾
4 – E _{hex}	Réservé par DeviceNet
F _{hex}	Valeur par défaut ⁵⁾

¹⁾ La station qui produit génère la valeur CID. Cette valeur doit être chargée dans l'attribut consumed_connection_id associé à l'objet ou aux objets connexion consommateurs.
²⁾ Le MAC ID de destination est spécifié dans l'identificateur du groupe 2. La station qui consomme génère la valeur CID et la charge dans l'attribut consumed_connection_id associé à l'objet connexion. L'ID de message du groupe 2 ayant la plus petite valeur numérique disponible doit être utilisé pour générer la valeur de l'attribut consumed_connection_id. Cette valeur doit être chargée dans l'attribut produced_connection_id de l'objet connexion producteur.
³⁾ Le MAC ID source est spécifié dans l'identificateur du groupe 2. Dans ce cas, la station qui produit génère la valeur CID et la charge dans l'attribut produced_connection_id de l'objet connexion. Cette valeur doit être chargée dans l'attribut consumed_connection_id associé à l'objet ou aux objets connexion consommateurs.
⁴⁾ La station qui produit génère la valeur CID. La valeur CID doit être chargée dans l'attribut consumed_connection_id de cet objet connexion.
⁵⁾ Valeur par défaut attribuée au demi-octet des caractéristiques initiales de consommation, dans une connexion d'E/S. S'il s'agit d'une connexion d'E/S de production uniquement, ce demi-octet conserve la valeur par défaut. Les connexions de messagerie explicite configurent automatiquement cet attribut lorsque la connexion est établie.

Table 15 – Values for the initial production characteristics nibble

Value	Meaning
0	Produce across message group 1 ¹⁾
1	Produce across message group 2 (destination) ²⁾
2	Produce across message group 2 (source) ³⁾
3	Produce across message group 3 ⁴⁾
4 – E _{hex}	Reserved
F _{hex}	Default value ⁵⁾

¹⁾ The producing node generates the CID value and loads it into the connection object's produced_connection_id attribute. The producing node allocates a message ID from its group 1 message ID pool and combines this with its source MAC ID to generate the CID. The numerically lowest available group 1 message ID shall be used in generating the produced_connection_id attribute value. This value shall also be loaded into the corresponding consumed_connection_id attribute associated with the consuming connection object(s).
²⁾ The destination MAC ID shall be placed within the MAC ID component of the group 2 identifier field. In this case, the consuming node generates the CID value to be associated with transmissions across this connection. When the consuming node has generated this value and loaded it into the appropriate connection object's consumed_connection_id attribute, this value shall be read and subsequently loaded into the producing connection object's produced_connection_id attribute.
³⁾ The source MAC ID shall be placed within the MAC ID component of the group 2 identifier. In this case, the producing node generates the CID value and loads it into the connection object's produced_connection_id attribute. The numerically lowest available group 2 message ID shall be used in generating the produced_connection_id attribute value. This value shall also be loaded into the corresponding consumed_connection_id attribute associated with the consuming connection object(s).
⁴⁾ The producing node generates the CID value and loads it into the connection object's produced_connection_id attribute. The producing node allocates a message ID from its group 3 message ID pool and combines this with its source MAC ID to generate the CID. The numerically lowest available group 3 message ID shall be used in generating the produced_connection_id attribute value. This value shall also be loaded into the corresponding consumed_connection_id attribute associated with the consuming connection object(s).
⁵⁾ The default value assigned to the initial production characteristics nibble within an I/O connection. If this is a consuming only I/O connection, then the default value remains in this nibble. Explicit messaging connection objects automatically configure this attribute when the connection is established.

Table 16 lists the possible values within the initial consumption characteristics nibble (lower nibble) of the initial_comm_characteristics attribute.

Table 16 – Values for the initial consumption characteristics nibble

Value	Meaning
0	Consume a group 1 message ¹⁾
1	Consume a group 2 message (destination) ²⁾
2	Consume a group 2 message (source) ³⁾
3	Consume a group 3 message ⁴⁾
4 – E _{hex}	Reserved by DeviceNet
F _{hex}	Default value ⁵⁾

¹⁾ The producing node generates the CID value. This value shall be loaded into the consumed_connection_id attribute associated with the consuming connection object(s).
²⁾ The destination MAC ID is specified within the group 2 identifier. The consuming node generates the CID value and loads it into the consumed_connection_id attribute associated with this connection object. The numerically lowest available group 2 message ID shall be used in generating the consumed_connection_id attribute value. This value shall be loaded into the producing connection object's produced_connection_id attribute.
³⁾ The source MAC ID is specified within the group 2 identifier. In this case, the producing node generates the CID value and loads it into the connection object's produced_connection_id attribute. This value shall be loaded into the consumed_connection_id attribute associated with the consuming connection object(s).
⁴⁾ The producing node generates the CID value. The CID value shall be loaded into this connection object's consumed_connection_id attribute.
⁵⁾ The default value assigned to the initial consumption characteristics nibble within an I/O connection. If this is a producing only I/O connection, then the default value remains in this nibble. Explicit messaging connections automatically configure this attribute when the connection is established.

La station qui génère un CID doit s'assurer qu'elle n'attribue pas de paire ID (de message/ MAC ID) telle que deux stations différentes puissent transmettre des séquences binaires identiques dans le champ identificateur (voir 1.3.2.9).

- 7) **produced_connection_size**: La signification de cet attribut est différente pour les connexions de messagerie explicite et les connexions d'E/S.

Pour les connexions de messagerie explicite:

Cet attribut définit le nombre maximal d'octets du corps d'un message qu'une station peut transmettre via cette connexion.

Les stations qui n'acceptent pas la transmission du protocole de fragmentation initialisent cet attribut à la valeur 7 (en-tête de message (1 octet) + corps du message (7 octets) = 8 octets, longueur maximale d'un message explicite non fragmenté). Les stations qui ne peuvent pas prédéfinir ou ne prédéfinissent pas une longueur maximale de transmission placent la valeur FFFF_{hex} dans cet attribut. Les stations qui acceptent le protocole de fragmentation mais qui imposent une limite au nombre d'octets du corps d'un message pouvant être transmis en une seule série fragmentée initialisent cet attribut avec cette limite.

Pour les connexions d'E/S:

Si l'attribut transportclass_trigger indique que cette instance de connexion est un producteur, cet attribut indique alors la quantité maximale de données d'E/S qui peut être produite en un seul message via cette connexion. La quantité de données d'E/S à transmettre doit être inférieure ou égale à l'attribut produced_connection_size.

La valeur par défaut de cet attribut dans une connexion d'E/S est 0. S'il est mis à une valeur supérieure à 8, la connexion doit alors utiliser le protocole de fragmentation.

Les messages d'E/S ne contenant pas de données applicatives d'E/S et qui ont été configurés pour contenir des données indiquent un événement «pas de données» pour le ou les objets applicatifs de réception. Le comportement d'un objet applicatif sur détection de l'événement «pas de données» est spécifique à cet objet.

- 8) **consumed_connection_size**: La signification de cet attribut est différente pour les connexions de messagerie explicite et les connexions d'E/S.

Pour les connexions de messagerie explicite:

Cet attribut définit le nombre maximal d'octets du corps des messages qu'une station peut recevoir via cette connexion.

Les stations qui n'acceptent pas la réception du protocole de fragmentation initialisent cet attribut à la valeur 7 (en-tête de message (1 octet) + corps du message (7 octets) = 8 octets, longueur maximale d'un message explicite non fragmenté). Les stations qui ne peuvent pas prédéfinir ou qui ne prédéfinissent pas une longueur maximale de réception placent la valeur FFFF_{hex} dans cet attribut. Les stations qui acceptent le protocole de fragmentation mais qui imposent une limite au nombre d'octets du corps d'un message pouvant être reçus en une seule série fragmentée initialisent cet attribut avec cette limite.

Pour les connexions d'E/S:

Si l'attribut transportclass_trigger indique que cette instance de connexion est un consommateur, cet attribut indique alors la quantité maximale de données d'E/S pouvant être reçue en un seul message via cette connexion. La quantité de données d'E/S reçue doit être inférieure ou égale à l'attribut consumed_connection_size. Si un objet connexion d'E/S reçoit un message dont la longueur est supérieure à celle spécifiée par la valeur de cet attribut, il ignore ce message et arrête tout traitement ultérieur.

La valeur par défaut de cet attribut dans une connexion d'E/S est 0. S'il est mis à une valeur supérieure à 8, la connexion doit alors utiliser le protocole de fragmentation.

- 9) **expected_packet_rate**: Cet attribut est utilisé pour générer les valeurs chargées dans le temporisateur de déclenchement de transmission et dans le temporisateur d'inactivité/surveillance (chien de garde). Il détermine la ou les valeurs chargées dans les temporisateurs en millisecondes. Voir 1.3.2.5 pour une description des temporisateurs de déclenchement de transmission et d'inactivité/surveillance.

The node that generates a CID shall guarantee that it does not allocate the message ID/MAC ID pair in such a way that two separate nodes are capable of transmitting identical bit patterns within the identifier field (see 5.3.2.9).

- 7) **produced_connection_size**: The meaning of this attribute is different for explicit messaging connections and for I/O connections.

For explicit messaging connections:

This attribute defines the maximum number of message body bytes that a node is able to transmit across this connection.

Nodes that do not support the transmission of the fragmentation protocol initialise this attribute to the value 7 (message header (1 byte) + message body (7 bytes) = 8 bytes, which is the maximum length of a non-fragmented explicit message). Nodes that cannot or do not predefine a transmit length limit place the value FFFF_{hex} into this attribute. Nodes that support the fragmentation protocol, but place a limit on the amount of message body bytes that can be transmitted in a single fragmented series initialise this attribute with this limit.

For I/O connections:

If the transportclass_trigger attribute identifies that this connection instance is a producer, then this attribute identifies the maximum amount of I/O data that may be produced as a single message across this connection. The amount of I/O data to be transmitted shall be less than or equal to the produced_connection_size attribute.

This attribute defaults to 0 within an I/O connection. If this attribute is set to a value greater than 8 in an I/O connection, then the connection shall use the fragmentation protocol.

I/O messages that contain no application I/O data and were configured to contain data indicate a no data event for the receiving application object(s). The behaviour of an application object upon detection of the no data event is application object-specific.

- 8) **consumed_connection_size**: The meaning of this attribute is different for explicit messaging connections and for I/O connections.

For explicit messaging connections:

This attribute defines the maximum number of message body bytes that this node is able to receive across this connection.

Nodes that do not support the reception of the fragmentation protocol initialise this attribute to the value 7 (message header (1 byte) + message body (7 bytes) = 8 bytes, which is the maximum length of a non-fragmented explicit message). Nodes that cannot or do not predefine a receive length limit place the value FFFF_{hex} into this attribute. Nodes that support the fragmentation protocol, but place a limit on the amount of message body bytes that can be received in a single fragmented series initialise this attribute with this limit.

For I/O connections:

If the transportclass_trigger attribute identifies that this connection instance is a consumer, then this attribute identifies the maximum amount of data that may be received as a single message across this connection. The amount of I/O data received shall be less than or equal to the consumed_connection_size attribute. If an I/O connection object receives a message whose length is greater than the length specified by this attribute value, then it discards the message and discontinues any subsequent processing.

This attribute defaults to 0 within an I/O connection. If this attribute is set to a value greater than 8 in an I/O connection, then the connection shall use the fragmentation protocol.

- 9) **expected_packet_rate**: This attribute is used to generate the values loaded into the transmission trigger timer and the inactivity/watchdog timer. This attribute determines the value(s) loaded into the timers in milliseconds. See 5.3.2.5 for a description of the transmission trigger and inactivity/watchdog timers.

Une demande de configuration de cet attribut peut aboutir à la spécification d'une valeur de temporisation qu'un produit ne peut satisfaire. Lorsqu'une demande de modification de cet attribut est reçue, les étapes suivantes doivent être effectuées:

- si la valeur spécifiée n'est pas un multiple exact de la fréquence d'horloge disponible, elle est arrondie à l'unité supérieure utilisable. Cette valeur est chargée dans l'attribut `expected_packet_rate` et est renvoyée dans le champ des données de service d'un message réponse `set_attribute_single`;
- si la valeur requise est égale à un incrément de la fréquence d'horloge, elle est chargée dans l'attribut `expected_packet_rate` et renvoyée dans le champ des données de service d'un message réponse `set_attribute_single`.

Les étapes suivantes doivent être effectuées par un objet connexion dans l'état établi lorsqu'une demande de modification de l'attribut `expected_packet_rate` est reçue:

- les temporisateurs courants de déclenchement de transmission et d'inactivité/surveillance sont annulés;
- un nouveau temporisateur de déclenchement de transmission et un nouveau temporisateur d'inactivité/surveillance sont activés avec la nouvelle valeur de l'attribut `expected_packet_rate`.

La valeur par défaut de cet attribut est 2 500 (2 500 ms) pour les connexions de messagerie explicite et zéro (0 ms) pour une connexion d'E/S.

10) Réservé.

11) Réservé.

12) **watchdog_timeout_action**: Cet attribut définit l'action à entreprendre par l'objet connexion lorsque le temporisateur d'inactivité/surveillance (chien de garde) expire (voir tableau 17);

Tableau 17 – Valeurs pour `watchdog_timeout_action`

Valeur	Signification
0	Transition vers l'état de délai expiré. La connexion passe à l'état de délai expiré et y demeure jusqu'à ce qu'elle soit réinitialisée ou supprimée. C'est la valeur par défaut pour cet attribut pour une connexion d'E/S
1	Suppression automatique. La classe connexion supprime automatiquement la connexion dans le cas d'expiration du délai d'inactivité/de surveillance. C'est la valeur non modifiable pour cet attribut pour une connexion de messagerie explicite
2	Réinitialisation automatique. La connexion reste dans l'état établi et relance le temporisateur d'inactivité/surveillance
3 – 255	Réservé

13) **produced_connection_path_length**: Indique le nombre d'octets d'information dans l'attribut `produced_connection_path`. Cette valeur est mise à jour en même temps que l'attribut `produced_connection_path`. La valeur par défaut de cet attribut est 0.

14) **produced_connection_path**: L'attribut `produced_connection_path` comprend une suite d'octets qui définit le ou les objets applicatifs dont les données vont être produites via cet objet connexion. Le format de cette suite d'octets est indiqué à l'annexe C. Cet attribut se met par défaut dans l'état non existant à la création de la connexion. Il reste dans cet état pour les connexions de messagerie explicite et pour les objets connexion qui ne produisent pas.

15) **consumed_connection_path_length**: Indique le nombre d'octets d'information dans l'attribut `consumed_connection_path`. Cette valeur est mise à jour en même temps que l'attribut `consumed_connection_path`. La valeur par défaut de cet attribut est 0.

A request to configure this attribute may result in the specification of a time value that a product cannot meet. When a request to modify this attribute is received, the following steps shall be performed:

- if the specified value is not equal to a multiple of the available clock resolution, then the value is rounded up to the next useable value. This useable value is loaded into the `expected_packet_rate` attribute and is reported in the service data field of a `set_attribute_single` response message;
- if the requested value is equal to an increment of the clock resolution, then the requested value is loaded into the `expected_packet_rate` attribute and is reported in the service data field of a `set_attribute_single` response message.

The following steps shall be performed by a connection object in the established state when a request to modify the `expected_packet_rate` attribute is received:

- the current transmission trigger timer and inactivity/watchdog timer are cancelled;
- a new transmission trigger timer and a new inactivity/watchdog timer are activated based on the new value in the `expected_packet_rate` attribute.

This attribute defaults to 2 500 (2 500 ms) within explicit messaging connections and to zero (0 ms) within an I/O connection.

10) Reserved.

11) Reserved.

12) **watchdog_timeout_action:** This attribute defines the action the connection object shall perform when the inactivity/watchdog timer expires (see table 17);

Table 17 – Values for the `watchdog_timeout_action`

Value	Meaning
0	Transition to timed-out state. The connection transitions to the timed-out state and remains in this state until it is reset or deleted. This is the default value for this attribute in the case of I/O connections
1	Auto delete. The connection class automatically deletes the connection in the event of an inactivity/watchdog time-out. This is the non-modifiable value for this attribute in the case of explicit messaging connections
2	Auto reset. The connection remains in the established state and restarts the inactivity/watchdog timer
3 – 255	Reserved

13) **produced_connection_path_length:** Identifies the number of bytes of information within the `produced_connection_path` attribute. This is set when the `produced_connection_path` attribute is set. This attribute defaults to the value 0.

14) **produced_connection_path:** The `produced_connection_path` attribute is made up of a byte stream which defines the application object(s) whose data is to be produced by this connection object. The format of this byte stream is specified in annex C. This attribute defaults to non-existent upon creation of the connection. It remains non-existent within explicit messaging connections and within connection objects that do not produce.

15) **consumed_connection_path_length:** Identifies the number of bytes of information within the `consumed_connection_path` attribute. This is set when the `consumed_connection_path` attribute is set. This attribute defaults to the value 0.

- 16) **consumed_connection_path**: L'attribut `consumed_connection_path` comprend une suite d'octets qui définit le ou les objets applicatifs qui vont recevoir les données consommées via cet objet connexion. Le format de cette suite d'octets est indiqué à l'annexe C. Cet attribut se met par défaut dans l'état non existant à la création de la connexion. Il reste dans cet état pour les connexions de messagerie explicite et pour les objets connexion qui ne consomment pas.
- 17) **production_inhibit_time**: Cet attribut est utilisé pour configurer le temps minimal entre les productions successives de données. Il détermine la valeur chargée dans le temporisateur en millisecondes. Une valeur de 0 (valeur par défaut de cet attribut) indique l'absence de temps d'inhibition.

Une demande de configuration de cet attribut peut aboutir à la spécification d'une valeur de temporisation qu'un produit ne peut pas satisfaire. Lorsqu'une demande de modification de cet attribut est reçue, les étapes suivantes doivent être effectuées:

- si la valeur spécifiée n'est pas un multiple exact de la fréquence d'horloge, la valeur est arrondie à l'unité supérieure utilisable. Cette valeur est chargée dans l'attribut `production_inhibit_time` et est renvoyée dans le champ des données de service d'un message réponse `set_attribute_single`;
- si la valeur requise est égale à un incrément de la fréquence d'horloge, cette valeur est chargée dans l'attribut `production_inhibit_time` et est renvoyée dans le champ des données de service d'un message réponse `set_attribute_single`.

La valeur de `production_inhibit_time` est rechargée dans le temporisateur d'inhibition de la production chaque fois qu'il y a production de nouvelles données.

Lorsqu'un objet connexion est dans l'état établi, toute modification de l'attribut `production_inhibit_time` reste sans effet sur un temporisateur d'inhibition de production en cours de décomptage. La nouvelle valeur de `production_inhibit_time` est chargée dans le temporisateur d'inhibition de production lors de la production des données suivantes.

Lorsque le service `apply_attributes` est reçu, `production_inhibit_time` doit être vérifié par rapport à l'attribut `expected_packet_rate`. Si la valeur de `expected_packet_rate` est supérieure à zéro mais inférieure à la valeur de `production_inhibit_time`, une erreur doit être renvoyée. Le code d'erreurs supplémentaire renvoyé au service `apply_attributes` est `11hex`, indiquant l'attribut `production_inhibit_time`.

1.3.2.5 Temporisation d'une connexion

1.3.2.5.1 Généralités

Trois types de temporisateurs sont impliqués dans une connexion:

- temporisateur de déclenchement de transmission;
- temporisateur d'inactivité/surveillance (chien de garde);
- temporisateur d'inhibition de production.

1.3.2.5.2 Temporisateur de déclenchement de transmission

L'expiration du délai de déclenchement de transmission est une indication de la nécessité éventuelle pour l'objet connexion de transmettre un message. S'il n'y a eu aucune production depuis l'activation du temporisateur, l'objet connexion doit alors produire, afin d'éviter l'expiration du délai d'inactivité/surveillance (chien de garde) à la ou aux extrémités serveur.

- 16) **consumed_connection_path:** The consumed_connection_path attribute is made up of a byte stream which defines the application object(s) that are to receive the data consumed by this connection object. The format of this byte stream is specified in annex C. This attribute defaults to non-existent upon creation of the connection. It remains non-existent within explicit messaging connections and within connection objects that do not consume.
- 17) **production_inhibit_time:** This attribute is used to configure the minimum time between successive data production. This attribute determines the value loaded into the timer in milliseconds. A value of 0 (the default value for this attribute) indicates no inhibit time.

A request to configure this attribute may result in the specification of a time value that a product cannot meet. When a request to modify this attribute is received, the following steps shall be performed:

- if the specified value is not equal to a multiple of the available clock resolution, then the value is rounded up to the next useable value. This useable value is loaded into the production_inhibit_time attribute and is reported in the service data field of a set_attribute_single response message;
- if the requested value is equal to an increment of the clock resolution, then the requested value is loaded into the production_inhibit_time attribute and is reported in the service data field of a set_attribute_single response message.

The production_inhibit_time value is loaded into the production inhibit timer each time new data production occurs.

When a connection object is in the established state, any modification to the production_inhibit_time attribute has no effect on a currently running production inhibit timer. The new production_inhibit_time value is loaded into the production inhibit timer when the next data production occurs.

When the apply_attributes service is received, production_inhibit_time shall be verified against the expected_packet_rate attribute. If the expected_packet_rate value is greater than zero, but less than the production_inhibit_time value, then an error shall be returned. The additional error code returned to the apply_attributes service is 11_{hex}, indicating the production_inhibit_time attribute.

5.3.2.5 Connection timing

5.3.2.5.1 General

Three types of timers are involved in a connection:

- transmission trigger timer;
- inactivity/watchdog timer;
- production inhibit timer.

5.3.2.5.2 Transmission trigger timer

Expiry of the transmission trigger time is an indication that the associated connection object may need to transmit a message. If a production has not occurred since the timer was activated, then the connection object shall produce, in order to avoid an inactivity/watchdog time-out at the server end-point(s).

1.3.2.5.3 Temporisateur d'inactivité/surveillance (chien de garde)

Ce temporisateur est utilisé par les objets connexion consommateurs, au nombre desquels:

- les objets connexion des extrémités client dont l'attribut `transportclass_trigger` indique une classe de transport 2 ou 3;
- tous les objets connexion des extrémités serveur.

Ce temporisateur est activé lorsque la connexion passe à l'état établi. Les tâches dont la liste figure ci-dessous doivent être effectuées par un objet connexion après avoir constaté la consommation d'un message valide:

- le temporisateur d'inactivité/surveillance en cours doit être arrêté;
- un nouveau temporisateur d'inactivité/surveillance doit être activé.

Ces actions doivent être accomplies avant de traiter le message reçu.

L'expiration du temps d'inactivité/surveillance est une indication que l'objet connexion a expiré pendant l'attente de consommation. L'objet connexion doit alors procéder comme suit:

- fournir une indication de cet événement à l'application;
- exécuter l'action indiquée par l'attribut `watchdog_timeout_action`.

La valeur initiale chargée dans le temporisateur d'inactivité/surveillance est égale à 10 000 ms ou à la valeur de `expected_packet_rate` multipliée par 4, la valeur la plus élevée étant retenue. Toutes les activations ultérieures du temporisateur d'inactivité/surveillance utilisent la valeur de `expected_packet_rate` multipliée par 4 comme nombre de millisecondes à charger dans le temporisateur d'inactivité/surveillance. Si l'attribut `expected_packet_rate` contient la valeur 0, le temporisateur d'inactivité/surveillance n'est alors ni activé, ni utilisé par l'objet connexion.

1.3.2.5.4 Temporisateur d'inhibition de production

Ce temporisateur doit être géré par l'objet connexion se trouvant dans l'extrémité client d'une connexion lorsque la valeur de l'attribut `production_inhibit_time` est différente de zéro. Ce temporisateur est lancé lorsque des données sont produites par l'objet applicatif. L'objet connexion ne doit pas produire de nouvelles données tant que le temporisateur décompte, bien qu'une réémission puisse être envoyée. L'expiration du temps d'inhibition de production autorise l'objet connexion à envoyer de nouvelles données.

Le temporisateur d'inhibition de production est initialisé avec la valeur de l'attribut `production_inhibit_time`. Si l'attribut `production_inhibit_time` contient la valeur 0, le temporisateur d'inhibition de production n'est alors ni activé, ni utilisé par l'extrémité client.

1.3.2.6 Services d'instance d'un objet connexion

L'instance d'un objet connexion accepte les services DeviceNet communs (comme décrits dans le tableau 18).

5.3.2.5.3 Inactivity/watchdog timer

This timer is used by consuming connection objects including:

- client end-point connection objects whose transportclass_trigger attribute indicates either transport class 2 or 3;
- all server end-point connection objects.

This timer is activated when the connection transitions to the established state. The tasks listed below shall be performed by a connection object upon detecting that a valid message has been consumed:

- the current inactivity/watchdog timer shall be stopped;
- a new inactivity/watchdog timer shall be activated.

These actions shall take place before the received message is processed.

Expiry of the inactivity/watchdog time is an indication that the connection object has timed-out while waiting to consume. The connection object shall then perform the following steps:

- issue an indication of this event to the application;
- perform the action indicated by the watchdog_timeout_action attribute.

The initial value loaded into the inactivity/watchdog timer is either 10 000 ms or expected_packet_rate multiplied by 4, depending on which value is numerically the greater. All subsequent activations of the inactivity/watchdog timer use expected_packet_rate multiplied by 4 as the number of milliseconds to load into the inactivity/watchdog timer. If the expected_packet_rate attribute contains the value 0, then the inactivity/watchdog timer is neither activated nor used by the connection object.

5.3.2.5.4 Production inhibit timer

This timer shall be managed by the connection object within the client end-point of a connection when the production_inhibit_time attribute value is non-zero. This timer is started when data is produced by the application object. The connection object shall not produce new data if this timer is running, although a retry may be sent. Expiry of the production inhibit time allows the connection object to send new data.

The production inhibit timer is initialised with the value in the production_inhibit_time attribute. If the production_inhibit_time attribute contains the value 0, then the production inhibit timer is neither activated nor used by the client end-point.

5.3.2.6 Connection object instance services

The connection object instance supports the following DeviceNet common services (as described in table 18).

Tableau 18 – Services d'instance des objets connexion

Mise en oeuvre	Nom du service	Description du service
Requis	Get_attribute_single	Lit l'attribut d'un objet connexion
En option	Set_attribute_single	Modifie l'attribut d'un objet connexion. L'objet connexion renvoie les informations dans le champ des données de service d'une réponse set_attribute_single lorsque l'attribut expected_packet_rate est modifié comme indiqué en 1.3.2.4
En option	Reset	Remet à zéro le temporisateur d'inactivité/surveillance associé à un objet connexion. Lorsqu'une connexion en état de délai expiré reçoit une demande de remise à zéro, elle repasse également en l'état établi
En option	Delete	Supprime un objet connexion et libère toutes les ressources associées
En option	Apply_attributes	Fournit l'objet connexion à l'application qui accomplit l'ensemble des tâches nécessaires à la création de la connexion spécifiée. Une instance de connexion renvoie les attributs produced_connection_id et consumed_connection_id dans le champ des données de service d'un message réponse apply_attributes, comme indiqué dans le tableau 19
NOTE Les codes des services sont définis à l'annexe A.		

Le tableau 19 décrit les informations qui doivent être spécifiées par une instance d'objet connexion dans le champ des données de service d'une réponse apply_attributes positive.

Tableau 19 – Données de service pour la réponse apply_attributes de l'objet connexion

Nom	Type de données	Description du paramètre
CID produite	UINT	Contient la valeur de l'attribut produced_connection_id de l'instance de connexion
CID consommée	UINT	Contient la valeur de l'attribut consumed_connection_id de l'instance de connexion

L'objet connexion accepte également les services internes suivants:

- **send_message**: utilisé en interne pour déclencher la transmission d'un message;
- **receive_data**: utilisé en interne pour remettre une trame reçue à l'objet connexion. L'objet connexion détermine s'il doit ou non réassembler une série de fragments de données en un message complet avant de traiter le message.

1.3.2.7 Comportement d'une instance de connexion

La figure 24 donne un aperçu général du comportement associé à un objet connexion d'E/S (attribut instance_type = E/S).

Table 18 – Connection object instance services

Need in implementation	Service name	Service description
Required	Get_attribute_single	Reads a connection object attribute
Optional	Set_attribute_single	Modifies a connection object attribute. The connection object returns information in the service data field of a set_attribute_single response when the expected_packet_rate attribute is modified as indicated in 5.3.2.4
Optional	Reset	Resets the inactivity/watchdog timer associated with a connection object. When a connection in the timed-out state receives a reset request, it also transitions back to the established state
Optional	Delete	Deletes a connection object and releases all associated resources
Optional	Apply_attributes	Delivers the connection object to the application which performs the set of tasks necessary to create the specified connection. A connection instance returns the produced_connection_id and consumed_connection_id attributes within the service data field of an apply_attributes response message, as indicated in table 19
NOTE Service codes are defined in annex A.		

The information to be specified by a connection object instance within the service data field of a successful apply_attributes response is shown in table 19.

Table 19 – Service data for connection object apply_attributes response

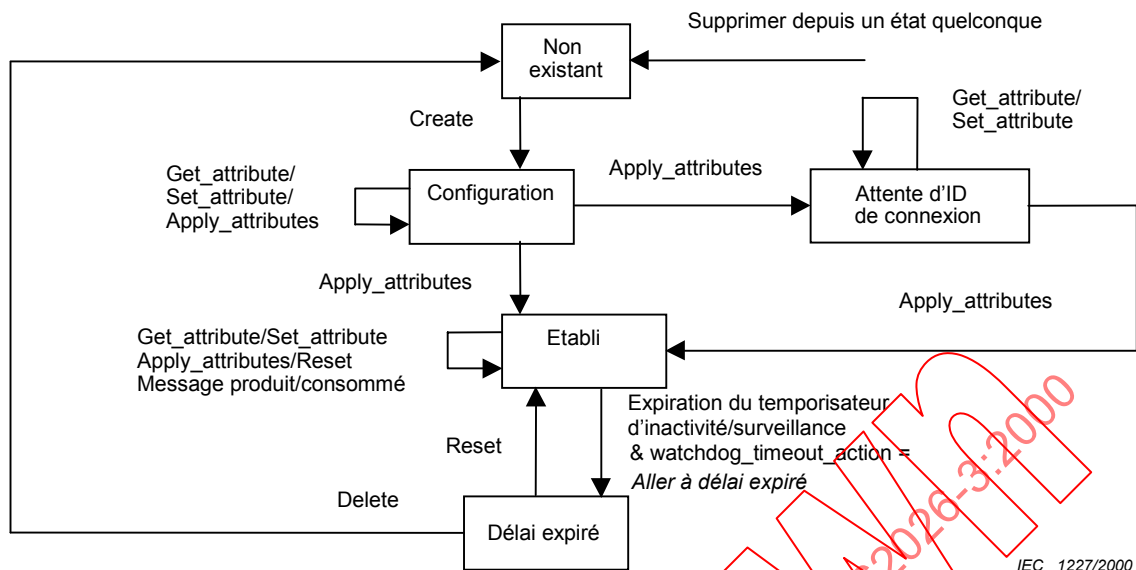
Name	Data type	Description of parameter
Produced CID	UINT	Contains the value within the connection instance's produced_connection_id attribute
Consumed CID	UINT	Contains the value within the connection instance's consumed_connection_id attribute

The connection object also supports the following internal services:

- **send_message**: used internally to trigger the transmission of a message;
- **receive_data**: used internally to deliver a received frame to the connection object. The connection object determines whether or not it needs to reassemble a series of data fragments into a complete message before processing the message.

5.3.2.7 Connection instance behaviour

Figure 24 provides a general overview of the behaviour associated with an I/O connection object (instance_type attribute = I/O).



IEC 1227/2000

Figure 24 – Diagramme de la machine d'état d'un objet connexion d'E/S

Le tableau 20 détaille la table états-événements d'une connexion d'E/S.

Tableau 20 – Table états-événements d'une connexion d'E/S

Etat de l'objet connexion d'E/S					
Événement	Non existant	Configuration	Attente de CID	Etabli	Délai expiré
La classe connexion reçoit une demande de création	La classe crée un objet connexion. Fixer instance_type à E/S. Fixer les autres attributs aux valeurs par défaut. Transition à configuration	Sans objet	Sans objet	Sans objet	Sans objet
La classe connexion reçoit une demande de suppression	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Libérer toutes les ressources associées. Transition à non existant	Libérer toutes les ressources associées. Transition à non existant	Libérer toutes les ressources associées. Transition à non existant	Libérer toutes les ressources associées. Transition à non existant
Set_attribute_single	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Valider/exécuter la demande selon 1.3.2.8. Renvoyer la réponse	En cas de demande de modification de produced_ ou consumed_ connection_id, valider la valeur, exécuter la demande et renvoyer une réponse. S'il s'agit d'une demande d'accès à un autre attribut, renvoyer une réponse d'erreur dont le code d'erreur général est mis à 0C _{hex}	Valider/exécuter la demande selon 1.3.2.8. Renvoyer la réponse	Valider/exécuter la demande selon 1.3.2.8. Renvoyer la réponse

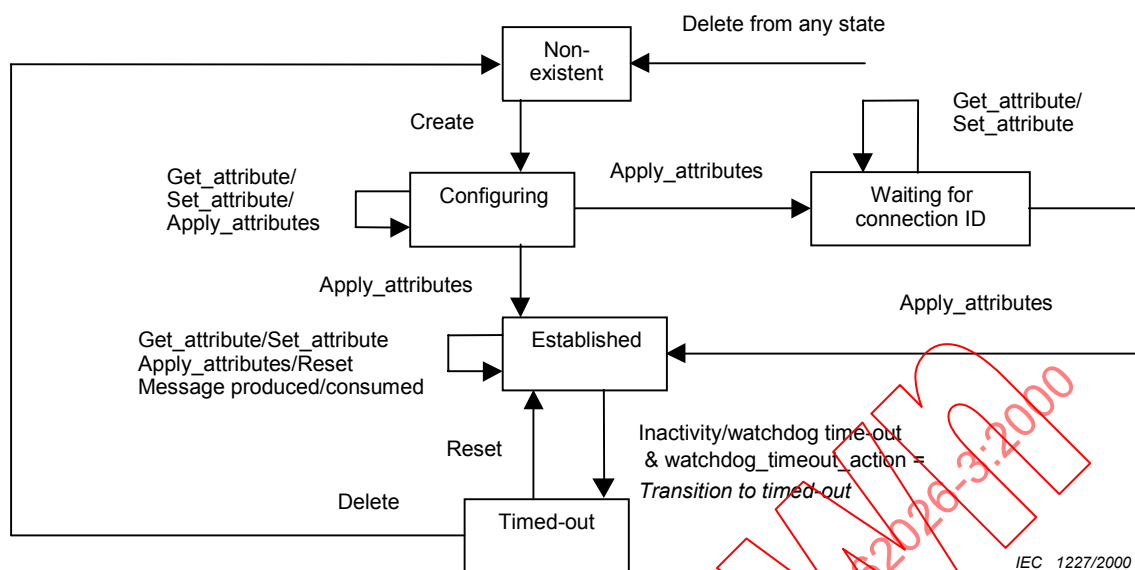


Figure 24 – I/O connection object state transition diagram

Table 20 provides a detailed state-event matrix for an I/O connection.

Table 20 – I/O connection state-event matrix

I/O connection object state					
Event	Non-existent	Configuring	Waiting for CID	Established	Timed-out
Connection class receives a create request	Class creates a connection object. Set instance_type to I/O. Set other attributes to default values. Transition to configuring	Not applicable	Not applicable	Not applicable	Not applicable
Connection class receives a delete request	Error: object does not exist (general error code 16 _{hex})	Release all associated resources Transition to non-existent	Release all associated resources Transition to non-existent	Release all associated resources Transition to non-existent	Release all associated resources Transition to non-existent
Set_attribute_single	Error: object does not exist (general error code 16 _{hex})	Validate/service the request according to 5.3.2.8 Return response	If request to modify produced_ or consumed_ connection_id, then validate the value, service the request and return response If this is a request to access any other attribute, then return an error response whose general error code is set to 0C _{hex}	Validate/service the request according to 5.3.2.8 Return response	Validate/service the request according to 5.3.2.8 Return response

Tableau 20 (suite)

Etat de l'objet connexion d'E/S					
Evénement	Non existant	Configuration	Attente de CID	Etabli	Délai expiré
Get_attribute_single	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Valider/exécuter la demande selon 1.3.2.8 Renvoyer la réponse	Valider/exécuter la demande selon 1.3.2.8 Renvoyer la réponse	Valider/exécuter la demande selon 1.3.2.8 Renvoyer la réponse	Valider/exécuter la demande selon 1.3.2.8 Renvoyer la réponse
Reset	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Erreur: l'objet ne peut pas exécuter le service demandé dans son mode/état actuel (valeur de code d'erreur général = 0C _{hex})	Erreur: l'objet ne peut pas exécuter le service demandé dans son mode/état actuel (valeur de code d'erreur général = 0C _{hex})	Si le temporisateur d'inactivité/surveillance fonctionne, l'arrêter et le relancer en utilisant la valeur de l'attribut expected_packet_rate	En utilisant la valeur de l'attribut expected_packet_rate, lancer le temporisateur d'inactivité/surveillance (si l'attribut expected_packet_rate n'a pas été mis à zéro) et repasser à l'état établi
Apply_attributes	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Fournir l'objet connexion à l'application qui valide les informations des attributs. Si l'un des attributs connection_id (produced ou consumed) a besoin d'être configuré et ne peut pas être généré par cette station, effectuer les autres étapes nécessaires pour configurer la connexion, renvoyer une réponse positive et passer à l'état attente de CID. Si tous les attributs sont configurés, effectuer les étapes nécessaires pour satisfaire cette connexion, lancer tous les temporisateurs requis et passer à l'état établi. Si une erreur est détectée, une réponse d'erreur est renvoyée et la connexion reste à l'état configuration ¹⁾	Si l'un des attributs connection_id (produced ou consumed) a toujours besoin d'être configuré et ne peut être généré par cette station, renvoyer une réponse positive et rester à l'état attente de CID. Si les attributs produced_ et/ou consumed_ connection_id sont configurés, passer à l'état établi et renvoyer une réponse positive ¹⁾	Toutes les modifications se produisent immédiatement, dès que la connexion est passée à l'état établi. Renvoyer une erreur: l'objet ne peut exécuter le service demandé dans son mode/état actuel (valeur de code d'erreur général = 0C _{hex})	Erreur: l'objet ne peut exécuter le service demandé dans son mode/état actuel (valeur de code d'erreur général = 0C _{hex})

Table 20 (continued)

I/O connection object state					
Event	Non-existent	Configuring	Waiting for CID	Established	Timed-out
Get_attribute_single	Error: object does not exist (general error code 16 _{hex})	Validate/service the request according to 5.3.2.8 Return response	Validate/service the request according to 5.3.2.8 Return response	Validate/service the request according to 5.3.2.8 Return response	Validate/service the request according to 5.3.2.8 Return response
Reset	Error: object does not exist (general error code 16 _{hex})	Error: the object cannot perform the requested service in its current mode/state (general error code value = 0C _{hex})	Error: the object cannot perform the requested service in its current mode/state (general error code value = 0C _{hex})	If the inactivity/watchdog timer is running, cancel it and re-start using the value in the expected_packet_rate attribute	Using the value in the expected_packet_rate attribute, start the inactivity/watchdog timer (if the expected_packet_rate attribute has not been set to zero) and transition back to the established state
Apply_attributes	Error: object does not exist (general error code 16 _{hex})	Deliver connection object to the application which validates the attribute information. If either of the connection_id attributes (produced or consumed) needs to be configured and cannot be generated by this node, then perform the other steps necessary to configure the connection, return a success response and transition to the waiting for CID state. If all attributes are configured, then perform the steps necessary to satisfy this connection, start all required timers, and transition to the established state. If an error is detected, then an error response is returned and the connection remains in the configuring state ¹⁾	If either of the connection_id attributes (produced or consumed) still needs to be configured and cannot be generated by this node, then return a success response and remain in the waiting for CID state. If the produced_ and/or consumed_ connection_id attributes are configured, transition to the established state and return a success response ¹⁾	All modifications take place immediately, once the connection has transitioned to the established state. Return error: the object cannot perform the requested service in its current mode/state (general error code value = 0C _{hex})	Error: the object cannot perform the requested service in its current mode/state (general error code value = 0C _{hex})

Tableau 20 (suite)

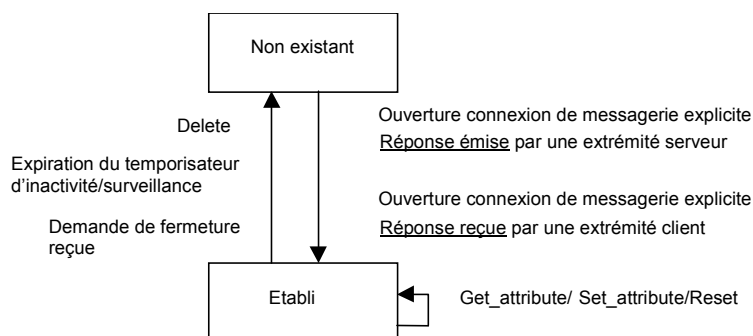
Etat de l'objet connexion d'E/S					
Événement	Non existant	Configuration	Attente de CID	Etabli	Délai expiré
Receive_data	Sans objet	Ne pas tenir compte de ce message	Ne pas tenir compte de ce message	Si un message ²⁾ complet et valide a été reçu, remettre à zéro le temporisateur d'inactivité/surveillance ³⁾ et passer le message d'E/S à l'application	Ne pas tenir compte de ce message
Send_message	Sans objet	Renvoyer une erreur interne Ne pas envoyer le message	Renvoyer une erreur interne Ne pas envoyer le message	Examiner l'attribut connection_size de la connexion et transmettre le message, fragmenté si nécessaire	Renvoyer une erreur interne Ne pas envoyer le message
Expiration du temporisateur d'inactivité/surveillance	Sans objet	Sans objet	Sans objet	Examiner l'attribut watchdog_timeout de la connexion et accomplir l'action indiquée	Sans objet

1) Si la configuration indique qu'un ID de message doit être alloué et qu'il n'existe aucun ID de message disponible dans le groupe spécifié, une réponse d'erreur dont le code d'erreur général indique ressource non disponible (02_{hex}) est renvoyée. Si la valeur d'un attribut d'objet connexion a satisfait à la vérification des limites de validité lors de sa configuration initiale mais se retrouve en conflit avec un autre élément d'information dans la station quand la demande apply est exécutée, une réponse d'erreur est renvoyée avec le code d'erreur général valeur d'attribut incorrecte (09_{hex}); le code d'erreur supplémentaire est chargé avec l'ID de l'attribut erroné de l'objet connexion.

2) L'objet connexion vérifie que la longueur du message d'E/S reçu est inférieure ou égale à l'attribut consumed_connection_size avant d'exécuter le message. Si la longueur du message reçu est inférieure ou égale à l'attribut consumed_connection_size, l'objet connexion d'E/S remet à zéro le temporisateur d'inactivité/surveillance, réalise le comportement externe indiqué par le contenu de ses attributs et remet le message à l'application. Si la longueur du message reçu est supérieure à l'attribut consumed_connection_size, l'objet connexion d'E/S écarte immédiatement le message et arrête tout traitement ultérieur. C'est la seule validation du contenu d'un message effectuée par un objet connexion d'E/S. Toute validation supplémentaire doit être effectuée par l'application.

3) Si un message fragmenté est reçu, le temporisateur d'inactivité/surveillance n'est pas remis à zéro tant que la totalité du message n'a pas été reçue.

La figure 25 donne un aperçu général du comportement associé à une connexion de messagerie explicite.



IEC 1228/2000

Figure 25 – Diagramme de la machine d'état d'un objet connexion de messagerie explicite

Table 20 (continued)

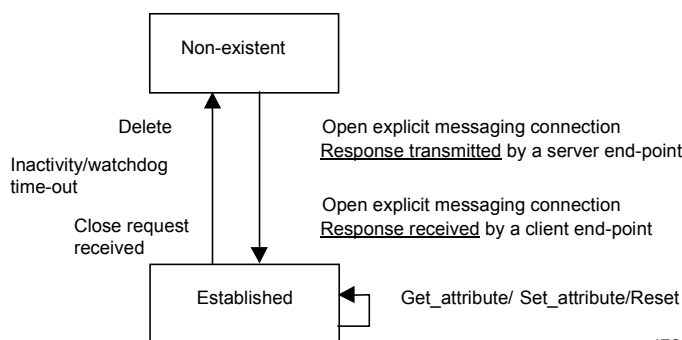
I/O connection object state					
Event	Non-existent	Configuring	Waiting for CID	Established	Timed-out
Receive_data	Not applicable	Discard the message	Discard the message	If a complete, valid message ²⁾ has been received, reset the inactivity/watchdog timer ³⁾ and deliver the I/O message to the application	Discard the message
Send_message	Not applicable	Return internal error – do not send the message	Return internal error – do not send the message	Examine the connection_size attribute of the connection and transmit the message, fragmented if necessary	Return internal error – do not send the message
Inactivity/watchdog timer expires	Not applicable	Not applicable	Not applicable	Examine the watchdog_timeout_action attribute of the connection and perform the indicated action	Not applicable

1) If the configuration indicates that a message ID needs to be allocated and an available message ID does not exist in the specified message group, then an error response whose general error code indicates resource unavailable (02_{hex}) is returned. If a connection object attribute value passed the range check when it was initially configured but the attribute value conflicts with another piece of information in the node when the apply request is processed, then an error response is returned whose general error code is set to invalid attribute value (09_{hex}) and whose additional code is set to the attribute ID of the offending connection object attribute ID.

2) The connection object verifies that the length of the received I/O message is less than or equal to the consumed_connection_size attribute prior to processing the message. If the length of the received message is less than or equal to the consumed_connection_size attribute, then the I/O connection object resets the inactivity/watchdog timer, exhibits the externally visible behaviour indicated by its attribute settings, and delivers the message to the application. If the length of the received message is greater than the consumed_connection_size attribute, then the I/O connection object immediately discards the message and discontinues any subsequent processing. This is the only message content validation performed by an I/O connection object. Any subsequent validation shall be performed by the application.

3) If a fragmented message is being received, then the inactivity/watchdog timer is not reset until the entire message has been received.

Figure 25 provides a general overview of the behaviour associated with an explicit messaging connection.



IEC 1228/2000

Figure 25 – Explicit messaging connection object state transition diagram

Le tableau 21 détaille la table états-événements d'une connexion de messagerie explicite.

Tableau 21 – Table états-événements d'une connexion de messagerie explicite

Etat de l'objet connexion de messagerie explicite		
Evénement	Non existant	Etabli
L'UCMM reçoit une demande/réponse d'ouverture de connexion de messagerie explicite et fait appel au service create de la classe connexion	Si possible, la classe crée un objet connexion avec l'attribut instance_type mis à messagerie explicite ¹⁾ . Les autres attributs sont automatiquement configurés en utilisant les informations de la demande/réponse d'ouverture de connexion de messagerie explicite. Passer à établi Si une demande est reçue, transmettre une réponse d'ouverture de connexion de messagerie explicite	Sans objet
UCMM reçoit une demande de fermeture	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Libérer toutes les ressources associées Transition vers non existant
La classe connexion reçoit une demande delete	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Libérer toutes les ressources associées Transition vers non existant
Set_attribute_single	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Valider/exécuter la demande selon 1.3.2.8 Renvoyer la réponse
Get_attribute_single	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Valider/exécuter la demande selon 1.3.2.8 Renvoyer la réponse
Reset	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Arrêter le temporisateur d'inactivité/surveillance en cours. En utilisant la valeur de l'attribut expected_packet_rate, relancer le temporisateur d'inactivité/surveillance
Apply_attributes	Erreur: l'objet n'existe pas (code d'erreur général 16 _{hex})	Erreur: l'objet ne peut pas exécuter le service demandé dans son mode/état actuel (valeur de code d'erreur général = 0C _{hex})
Receive_data	Sans objet	Si un message ou fragment de message valide a été reçu, remettre à zéro le temporisateur d'inactivité/surveillance ²⁾ . Traiter/stocker le fragment ou exécuter le message explicite
Send_message	Sans objet	Examiner la longueur du message à transmettre et le transmettre, fragmenté si nécessaire
Expiration du temporisateur d'inactivité/surveillance	Sans objet	Libérer toutes les ressources associées Transition à non existant

¹⁾ Si la configuration indique qu'un ID de message doit être alloué et qu'il n'existe pas d'ID de message disponible dans le groupe de messages spécifié, une réponse d'erreur dont le code général indique ressource non disponible (02_{hex}) est renvoyée.

²⁾ Le champ MAC ID de l'en-tête de message de tous les messages explicites et/ou fragments de messages est examiné. Si le MAC ID de destination est spécifié dans le CID (champ de l'identificateur de CAN), le MAC ID source de l'autre extrémité doit être spécifié dans l'en-tête du message. Si le MAC ID source est spécifié dans le CID, le MAC ID du module de réception doit être spécifié dans l'en-tête du message. Si l'une ou l'autre de ces vérifications échoue, le temporisateur d'inactivité/surveillance n'est pas remis à zéro et le message/fragment de message est ignoré.

1.3.2.8 Règles d'accès aux attributs de l'objet connexion

Pendant la configuration d'une instance de connexion à l'aide du service set_attribute, une station doit vérifier séparément la valeur de chaque attribut devant être modifié. Si une erreur est détectée, une réponse d'erreur doit être renvoyée.

Table 21 provides a detailed state-event matrix for an explicit messaging connection.

Table 21 – Explicit messaging connection state-event matrix

Explicit messaging connection object state		
Event	Non-existent	Established
UCMM receives an open explicit messaging connection request/response and invokes the create service of the connection class	If possible, class creates connection object with the instance_type attribute set to explicit messaging ¹⁾ Other attributes are automatically configured using the information in the open explicit messaging connection request/response. Transition to established If request received, transmit open explicit messaging connection response	Not applicable
UCMM receives a close request	Error: object does not exist (general error code 16 _{hex})	Release all associated resources Transition to non-existent
Connection class receives a delete request	Error: object does not exist (general error code 16 _{hex})	Release all associated resources Transition to non-existent
Set_attribute_single	Error: object does not exist (general error code 16 _{hex})	Validate/service the request according to 5.3.2.8 Return response
Get_attribute_single	Error: object does not exist (general error code 16 _{hex})	Validate/service the request according to 5.3.2.8 Return response
Reset	Error: object does not exist (general error code 16 _{hex})	Cancel the current inactivity/watchdog timer. Using the value in the expected_packet_rate attribute, restart the inactivity/watchdog timer
Apply_attributes	Error: object does not exist (general error code 16 _{hex})	Error: the object cannot perform the requested service in its current mode/state (general error code value = 0C _{hex})
Receive_data	Not applicable	If a valid message or message fragment has been received, then reset the inactivity/watchdog timer ²⁾ . Either process/store the fragment or handle the explicit message
Send_message	Not applicable	Examine the length of the message to transmit and transmit the message, fragmented if necessary
Inactivity/watchdog timer expires	Not applicable	Release all associated resources Transition to non-existent
¹⁾ If the configuration indicates that a message ID needs to be allocated and an available message ID does not exist in the specified message group, then an error response whose general error code indicates resource unavailable (02 _{hex}) is returned. ²⁾ The MAC ID field within the message header of all explicit messages and/or message fragments is examined. If the destination MAC ID is specified in the CID (CAN identifier field), 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 node's MAC ID shall be specified in the message header. If either of these checks fail, then the inactivity/watchdog timer is not reset and the message/message fragment is discarded.		

5.3.2.8 Connection object attribute access rules

During the configuration of a connection instance using the set_attribute service, a node shall check the value of each separate attribute which is to be modified. If an error is detected, then an error response shall be returned.

Si les attributs `produced_connection_id` et/ou `consumed_connection_id` contiennent une valeur autre que par défaut à la réception de la demande `apply`, la partie correspondante dans l'attribut `initial_comm_characteristics` est ignorée et la valeur de l'ID est validée et utilisée.

Le tableau 22 et le tableau 23 indiquent quand un attribut peut être lu ou écrit via une opération `get` (lecture) ou `set` (écriture), en fonction de l'état de la connexion et de `instance_type`.

Si une demande `get/set` est reçue pour un attribut existant mais que l'état actuel et/ou le type de connexion ne permettent pas de valider l'accès requis, l'état d'erreur renvoyé indique que l'objet ne peut pas exécuter le service demandé dans son mode/état actuel (valeur de code d'erreur général = `0Chex`). Les codes d'erreurs sont spécifiés à l'annexe B.

Tableau 22 – Accès aux attributs d'un objet connexion d'E/S

Etat de la connexion d'E/S					
Attribut	Non existant	Configuration	Attente de CID	Etabli	Délai expiré
State	Non disponible	Get only	Get only	Get only	Get only
Instance_type	Non disponible	Get only	Get only	Get only	Get only
Transportclass_trigger	Non disponible	Get/set	Get only	Get only	Get only
Produced_connection_id	Non disponible	Get/set	Get/set	Get/set	Get/set
Consumed_connection_id	Non disponible	Get/set	Get/set	Get/set	Get/set
Initial_comm_characteristics	Non disponible	Get/set	Get only	Get only	Get only
Produced_connection_size	Non disponible	Get/set	Get only	Get only	Get only
Consumed_connection_size	Non disponible	Get/set	Get only	Get only	Get only
Expected_packet_rate ¹⁾	Non disponible	Get/set	Get only	Get/set	Get/set
Watchdog_timeout_action	Non disponible	Get/set	Get only	Get/set	Get/set
Produced_connection_path_length	Non disponible	Get only	Get only	Get only	Get only
Produced_connection_path	Non disponible	Get/set	Get only	Get only	Get only
Consumed_connection_path_length	Non disponible	Get only	Get only	Get only	Get only
Consumed_connection_path	Non disponible	Get/set	Get only	Get only	Get only
Production_inhibit_time	Non disponible	Get/set	Get only	Get/set	Get/set
1) Lorsqu'un objet connexion est à l'état établi, toute modification de l'attribut <code>expected_packet_rate</code> a un effet immédiat sur le temporisateur d'inactivité/surveillance. Les étapes suivantes sont effectuées par un objet connexion à l'état établi lorsqu'une demande de modification de l'attribut <code>expected_packet_rate</code> est reçue: – le temporisateur d'inactivité/surveillance en cours est arrêté; – un nouveau temporisateur d'inactivité/surveillance est activé avec la nouvelle valeur de l'attribut <code>expected_packet_rate</code>.					

If the `produced_connection_id` and/or `consumed_connection_id` attributes contain a non-default value upon reception of the apply request, then the relevant portion of the `initial_comm_characteristics` attribute is ignored and the ID value is validated and used.

Table 22 and table 23 indicate when an attribute can be read or written via a get or set operation based on the connection's state and instance_type.

If a request to get/set a supported attribute is received but the current state and/or type of connection dictates that the requested access is invalid, then the returned error status indicates that the object cannot perform the requested service in its current mode/state (general error code value = $0C_{hex}$). Error codes are specified in annex B.

Table 22 – I/O connection object attribute access

I/O connection state					
Attribute	Non-existent	Configuring	Waiting for CID	Established	Timed-out
State	Not available	Get only	Get only	Get only	Get only
Instance_type	Not available	Get only	Get only	Get only	Get only
Transportclass_trigger	Not available	Get/set	Get only	Get only	Get only
Produced_connection_id	Not available	Get/set	Get/set	Get/set	Get/set
Consumed_connection_id	Not available	Get/set	Get/set	Get/set	Get/set
Initial_comm_characteristics	Not available	Get/set	Get only	Get only	Get only
Produced_connection_size	Not available	Get/set	Get only	Get only	Get only
Consumed_connection_size	Not available	Get/set	Get only	Get only	Get only
Expected_packet_rate ¹⁾	Not available	Get/set	Get only	Get/set	Get/set
Watchdog_timeout_action	Not available	Get/set	Get only	Get/set	Get/set
Produced_connection_path_length	Not available	Get only	Get only	Get only	Get only
Produced_connection_path	Not available	Get/set	Get only	Get only	Get only
Consumed_connection_path_length	Not available	Get only	Get only	Get only	Get only
Consumed_connection_path	Not available	Get/set	Get only	Get only	Get only
Production_inhibit_time	Not available	Get/set	Get only	Get/set	Get/set
¹⁾ When a connection object is in the established state, any modifications to the <code>expected_packet_rate</code> attribute have immediate effect on the inactivity/watchdog timer. The following steps are performed by a connection object in the established state when a request is received to modify the <code>expected_packet_rate</code> attribute: – the current inactivity/watchdog timer is cancelled; – a new inactivity/watchdog timer is activated based on the new value in the <code>expected_packet_rate</code> attribute.					

Tableau 23 – Accès aux attributs d'un objet connexion de messagerie explicite

Etat d'une connexion de messagerie explicite		
Attribut	Non existant	Etabli
State	Non disponible	Get only
Instance_type	Non disponible	Get only
Transportclass_trigger	Non disponible	Get only
Produced_connection_id	Non disponible	Get only
Consumed_connection_id	Non disponible	Get only
Initial_comm_characteristics	Non disponible	Get only
Produced_connection_size	Non disponible	Get only
Consumed_connection_size	Non disponible	Get only
Expected_packet_rate ¹⁾	Non disponible	Get/set ¹⁾
Watchdog_timeout_action	Non disponible	Get only
Produced_connection_path_length	Non disponible	Get only
Produced_connection_path	Non disponible	Get only
Consumed_connection_path_length	Non disponible	Get only
Consumed_connection_path	Non disponible	Get only
Production_inhibit_time	Non disponible	Get only
¹⁾ Lorsqu'un objet connexion est dans l'état établi, toute modification de l'attribut expected_packet_rate a un effet immédiat sur le temporisateur d'inactivité/surveillance. Les étapes suivantes sont effectuées par un objet connexion à l'état établi lorsqu'une demande de modification de l'attribut expected_packet_rate est reçue: – le temporisateur d'inactivité/surveillance en cours est arrêté; – un nouveau temporisateur d'inactivité/surveillance est activé avec la nouvelle valeur de l'attribut expected_packet_rate.		

1.3.2.9 Gestion dynamique des ID de messages

L'établissement dynamique des connexions d'E/S et de messagerie explicite nécessite que toutes les extrémités mettent en œuvre des procédures internes d'attribution d'ID de message. Ces procédures doivent aussi noter qu'un ID de message précédemment alloué est de nouveau disponible lorsque cet ID de message n'est plus utilisé.

Pour réduire la possibilité d'erreurs de CID, les règles qui suivent doivent être observées:

- la procédure d'attribution d'un CID doit s'assurer qu'il n'y a pas deux stations DeviceNet avec des configurations conduisant à la transmission de séquences binaires identiques dans le champ identificateur de CAN;
- la procédure de libération d'un CID doit s'assurer que toutes les extrémités d'une connexion sont à l'état «délai expiré» avant de réutiliser un ID de message:
 - si l'ID de message est associé à une connexion qui active un temporisateur d'inactivité/surveillance, un nouveau temporisateur d'inactivité/surveillance est alors activé. A l'expiration du temporisateur, l'ID de message est marqué comme étant disponible;
 - si l'ID de message est associé à une connexion qui n'active pas un temporisateur d'inactivité/surveillance, il peut être immédiatement noté comme disponible.

1.3.3 Définition de la classe d'objet DeviceNet (code d'ID de classe: 3)

1.3.3.1 Généralités

L'objet DeviceNet est utilisé pour fournir la configuration et l'état d'une liaison physique à DeviceNet. Un appareil doit mettre en œuvre un objet DeviceNet par liaison physique.

Table 23 – Explicit messaging connection object attribute access

Explicit messaging connection state		
Attribute	Non-existent	Established
State	Not available	Get only
Instance_type	Not available	Get only
Transportclass_trigger	Not available	Get only
Produced_connection_id	Not available	Get only
Consumed_connection_id	Not available	Get only
Initial_comm_characteristics	Not available	Get only
Produced_connection_size	Not available	Get only
Consumed_connection_size	Not available	Get only
Expected_packet_rate ¹⁾	Not available	Get/set ¹⁾
Watchdog_timeout_action	Not available	Get only
Produced_connection_path_length	Not available	Get only
Produced_connection_path	Not available	Get only
Consumed_connection_path_length	Not available	Get only
Consumed_connection_path	Not available	Get only
Production_inhibit_time	Not available	Get only
¹⁾ When a connection object is in the established state, any modifications to the expected_packet_rate attribute have immediate effect on the inactivity/watchdog timer. The following steps are performed by a connection object in the established state when a request is received to modify the expected_packet_rate attribute: – the current inactivity/watchdog timer is cancelled; – a new inactivity/watchdog timer is activated based on the new value in the expected_packet_rate attribute.		

5.3.2.9 Dynamic management of message IDs

Dynamically establishing both I/O and explicit messaging connections requires all end-points to implement internal message ID allocation procedures. These procedures shall also mark a previously allocated message ID as available when that message ID is no longer in use.

To reduce the likelihood of CID errors, the following rules shall be followed:

- a) the CID allocation process shall ensure that no two nodes on DeviceNet have configurations that allow transmission of an identical bit pattern within the CAN identifier field;
- b) the CID deallocation process shall ensure that all end-points of a connection have timed-out prior to reusing a message ID:
 - if the message ID is associated with a connection that activates an inactivity/watchdog timer, then a new inactivity/watchdog timer is activated. Upon expiration of this timer, the message ID is marked as available;
 - if the message ID is associated with a connection that does not activate an inactivity/watchdog timer, then the message ID can be immediately marked as available.

5.3.3 DeviceNet object class definition (class ID code: 3)

5.3.3.1 General

The DeviceNet object is used to provide the configuration and status of a physical attachment to DeviceNet. A product shall support one DeviceNet object per physical link attachment.

1.3.3.2 Attributs de la classe d'objet DeviceNet

Les attributs de classe pour l'objet DeviceNet sont définis dans le tableau 24.

Tableau 24 – Attributs de classe pour l'objet DeviceNet

ID d'attribut	Mise en oeuvre	Règle d'accès	Nom de l'attribut	Type de données	Description de l'attribut	Valeur
1	Requis	Get	Revision	UINT	Version de l'objet	2

1.3.3.3 Services communs de la classe d'objet DeviceNet

La classe d'objet DeviceNet doit accepter les services communs DeviceNet indiqués au tableau 25.

Tableau 25 – Services communs de la classe d'objet DeviceNet

Mise en oeuvre	Nom du service	Description du service
Requis	Get_attribute_single	Lecture d'un attribut de la classe d'objet DeviceNet
NOTE Les codes des services sont définis à l'annexe A.		

1.3.3.4 Attributs d'instance d'un objet DeviceNet

Le tableau 26 définit les attributs d'instance de l'objet DeviceNet.

Tableau 26 – Attributs d'instance pour l'objet DeviceNet

ID d'attribut	Mise en oeuvre	Règle d'accès	Nom	Type de données	Brève description de l'attribut	Valeurs
1	En option	Get/set	MAC ID	USINT	Adresse de station	Plage 0-63
2	En option	Get/set	Vitesse de transmission	USINT	Vitesse de transmission	Plage 0-2
3	En option	Get/set	BOI	BOOL	Interruption bus-off	
4	En option	Get/set	Compteur bus-off	USINT	Nombre de passages de CAN à l'état bus-off	Plage 0-255
5	En option	Get	Informations d'attribution	STRUCT		
			Choix d'attribution	BYTE	Voir 5.3.3.5.2.1.2	
			MAC ID du maître	USINT	MAC ID du maître (depuis allocate)	Plage 0-63 et 255 Modifié via allocate seulement
6	En option	Get	Commutateurs de MAC ID modifiés	BOOL	Le ou les commutateurs d'adresse de station ont changé depuis la dernière mise sous tension/réinitialisation	0 = pas de changement 1 = changement depuis la dernière réinitialisation ou mise sous tension
7	En option	Get	Commutateurs de vitesse de transmission modifiés	BOOL	Le ou les commutateurs de vitesse de transmission ont changé depuis la dernière mise sous tension/réinitialisation	0 = pas de changement 1 = changement depuis la dernière réinitialisation ou mise sous tension

5.3.3.2 DeviceNet object class attributes

The class attributes for the DeviceNet object are defined in table 24.

Table 24 – Class attributes for the DeviceNet object

Attribute ID	Need in implementation	Access rule	Attribute name	Data type	Attribute description	Value
1	Required	Get	Revision	UINT	Revision of the object	2

5.3.3.3 DeviceNet object class common services

The DeviceNet object class shall support the DeviceNet common services shown in table 25.

Table 25 – DeviceNet object class common services

Need in Implementation	Service name	Service description
Required	Get_attribute_single	Reads a DeviceNet object class attribute
NOTE Service codes are defined in annex A.		

5.3.3.4 DeviceNet object instance attributes

Table 26 defines the instance attributes for the DeviceNet object.

Table 26 – Instance attributes for the DeviceNet object

Attribute ID	Need in implementation	Access rule	Name	Data type	Brief description of attribute	Values
1	Optional	Get/set	MAC ID	USINT	Node address	Range 0-63
2	Optional	Get/set	Bit rate	USINT	Bit rate	Range 0-2
3	Optional	Get/set	BOI	BOOL	Bus-off interrupt	
4	Optional	Get/set	Bus-off counter	USINT	Number of times CAN went to the bus-off state	Range 0-255
5	Optional	Get	Allocation information	STRUCT		
			Allocation choice byte	BYTE	See 5.3.3.5.2.1.2	
			Master's MAC ID	USINT	MAC ID of master (from allocate)	Range 0-63 and 255 Modified via allocate only
6	Optional	Get	MAC ID switch changed	BOOL	Node address switch(es) have changed since last power-up/reset	0 = no change 1 = change since last reset or power-up
7	Optional	Get	Bit rate switch changed	BOOL	Bit rate switch(es) have changed since last power-up/reset	0 = no change 1 = change since last reset or power-up

Tableau 26 (suite)

ID d'attribut	Mise en oeuvre	Règle d'accès	Nom	Type de données	Brève description de l'attribut	Valeurs
8	En option	Get	Valeur des commutateurs de MAC ID	USINT	Valeur réelle du ou des commutateurs d'adresse de station	Plage 0-63
9	En option	Get	Valeur des commutateurs de vitesse de transmission	USINT	Valeur réelle du ou des commutateurs de vitesse de transmission	Plage 0-2
1) L'attribut informations d'attribution doit être présent si le jeu de connexions maître/esclave prédéfini est accepté.						

Des attributs sont décrits ci-après.

1) **MAC ID**

Cet attribut contient le MAC ID de l'appareil. Un appareil qui utilise, pour définir le MAC ID, des commutateurs accessibles par l'utilisateur lorsque le produit est installé doit renvoyer une réponse d'erreur dont le code général est 0E_{hex} (attribut non modifiable), en réponse à une demande set_attribute_single spécifiant l'attribut MAC ID. L'attribut MAC ID est considéré comme permanent en ce sens qu'une fois configuré, il ne doit pas être oublié après une mise hors et sous tension ou une réinitialisation de l'appareil. La valeur par défaut du MAC ID doit être égale à 63.

La modification du MAC ID nécessite qu'un appareil supprime tous les objets connexion et ré-exécute la machine d'état d'accès à la liaison définie en 1.4.

2) **Vitesse de transmission**

L'attribut vitesse de transmission indique la vitesse sélectionnée. Les valeurs sont indiquées au tableau 27.

Tableau 27 – Valeurs de l'attribut vitesse de transmission

Valeur	Signification
0	125 kbit/s
1	250 kbit/s
2	500 kbit/s
3-255	Réservé

Un appareil qui utilise des commutateurs accessibles par l'utilisateur après installation du produit pour régler la vitesse de transmission doit renvoyer une réponse d'erreur dont le code général est 0E_{hex} (attribut non modifiable) en réponse à une demande set_attribute_single spécifiant l'attribut vitesse de transmission. L'attribut vitesse de transmission est considéré comme permanent en ce sens qu'une fois configuré, il ne doit pas être oublié après une mise hors et sous tension ou une réinitialisation de l'appareil. La vitesse de transmission par défaut doit être de 125 kbit/s.

Aucune modification de la vitesse de transmission ne doit prendre effet tant que l'appareil n'est pas physiquement réinitialisé ou remis à zéro par l'envoi du service reset à l'objet identifié.

3) **Interruption bus-off** (comme décrit dans le tableau 28)

L'attribut interruption bus-off (BOI) consiste en un bit qui définit comment un dispositif CAN traite l'interruption bus-off.

Table 26 (continued)

Attribute ID	Need in implementation	Access rule	Name	Data type	Brief description of attribute	Values
8	Optional	Get	MAC ID switch value	USINT	Actual value of node address switch(es)	Range 0-63
9	Optional	Get	Bit rate switch value	USINT	Actual value of bit rate switch(es)	Range 0-2
1) The allocation information attribute shall be supported if the predefined master/slave connection set is supported.						

Some attributes are described below.

1) **MAC ID**

This attribute contains the MAC ID of this device. A device that uses switches which are user accessible when the product is installed to set the MAC ID shall return an error response whose general error code is set to 0E_{hex} (attribute not settable) in response to a set_attribute_single request specifying the MAC ID attribute. The MAC ID attribute is considered non-volatile in that, once configured, the attribute shall be remembered after a power cycle or device reset. The default value of the MAC ID shall be 63.

Modification of the MAC ID requires that a device deletes all connection objects and re-executes the link access state machine defined in 5.4.

2) **Bit rate**

The bit rate attribute indicates the selected bit rate. Values are shown in table 27.

Table 27 – Bit rate attribute values

Value	Meaning
0	125 kbit/s
1	250 kbit/s
2	500 kbit/s
3-255	Reserved

A device that uses switches which are user accessible when the product is installed to set the bit rate shall return an error response whose general error code is set to 0E_{hex} (attribute not settable) in response to a set_attribute_single request specifying the bit rate attribute. The bit rate attribute is considered non-volatile in that, once configured, the attribute shall be remembered after a power cycle or device reset. The default bit rate shall be 125 kbit/s.

Modification of the bit rate shall not take effect until the device is either physically reset or reset by sending the reset service to the identity object.

3) **Bus-off interrupt (BOI)** (as described in table 28)

The bus-off interrupt attribute consists of one bit that defines how a CAN device processes the bus-off interrupt.

Tableau 28 – Valeurs de l'attribut BOI (interruption bus-off)

Valeur	Signification
FAUX	Maintien du composant CAN dans son état bus-off (remis à zéro) sur détection d'une indication bus-off
VRAI	Si possible, remise à zéro totale du composant CAN et poursuite de la communication sur détection d'une indication bus-off

Lorsque l'attribut interruption bus-off est FAUX (mis à 0) et qu'un événement bus-off du composant CAN est détecté:

- le transcepteur doit être maintenu dans son état remis à zéro/bus-off;
- l'appareil doit se mettre dans l'état défaut de communication (voir 5.4).

Lorsque l'attribut interruption bus-off est VRAI (mis à 1) et qu'un événement bus-off du composant CAN est détecté, il est éventuellement possible de renvoyer le MAC et le transcepteur à un mode de fonctionnement normal et de continuer à communiquer suivant la machine d'état d'accès à la liaison détaillée en 1.4.

Si l'attribut interruption bus-off n'est pas géré, l'appareil doit appliquer le comportement indiqué par la valeur FAUX de l'attribut dans le tableau 28.

4) Compteur bus-off

Le compteur bus-off est initialisé à 0 à la mise sous tension ou à l'initialisation de l'appareil.

Le compteur bus-off arrête de compter lorsqu'il atteint sa valeur maximale.

L'objet DeviceNet remet à zéro le compteur bus-off chaque fois qu'il reçoit une demande set_attribute_single spécifiant l'attribut compteur bus-off. Les données de set_attribute_single ne sont pas utilisées; elles peuvent prendre n'importe quelles valeurs.

5) Informations d'attribution

L'attribut informations d'attribution doit être géré si le jeu de connexions maître/esclave prédéfini est géré. Il indique si ce jeu de connexions maître/esclave défini en 1.5 a été affecté ou non. Dans l'affirmative, cet attribut indique l'appareil responsable de l'affectation et la ou les connexion(s) actuellement attribuées.

Cet attribut est modifié lorsqu'une réponse positive associée à un service allocate_master/slave_connection_set (défini en 1.5) est générée. Il n'est pas modifiable par le service set_attribute_single. Une réponse d'erreur dont le champ de code général est 0E_{hex} (attribut non modifiable) est renvoyée si une demande set_attribute_single spécifie cet attribut.

L'attribut informations d'attribution comprend les éléments suivants:

– choix d'attribution

L'octet choix d'attribution indique quelles connexions maître/esclave prédéfinies sont actives (dans l'état configuration ou établi). Son format est spécifié en 5.3.3.5.2.1.2.

L'octet choix d'attribution est initialisé à 00 à la mise sous tension ou à la remise à zéro de l'appareil.

– MAC ID du maître

Le MAC ID du maître contient le MAC ID de l'appareil qui a attribué le jeu de connexions maître/esclave prédéfini via le service allocate_master/slave_connection_set. Cela comprend le champ MAC ID de l'appareil responsable de l'affectation, copié depuis la demande allocate_master/slave_connection_set.

Table 28 – BOI attribute values

Value	Meaning
FALSE	Hold the CAN chip in its bus-off (reset) state upon detection of a bus-off indication
TRUE	If possible, fully reset the CAN chip and continue communicating upon detection of a bus-off indication

When the bus-off interrupt attribute is FALSE (set to 0) and a CAN chip bus-off event is detected:

- the transceiver shall be held in its reset/bus-off state;
- the device shall enter the communication fault state (see 5.4).

When the bus-off interrupt attribute is TRUE (set to 1) and a CAN chip bus-off event is detected, it may be possible to return the MAC and transceiver implementation to its normal operating mode and continue communicating based on the link access state machine detailed in 5.4.

If the bus-off interrupt attribute is not supported, the device shall implement the behaviour indicated by attribute value FALSE in table 28.

4) **Bus-off counter**

The bus-off counter is initialised to 0 at power-up or device initialisation.

The bus-off counter stops counting when it reaches maximum count.

The DeviceNet object resets the bus-off counter to 0 whenever it receives a set_attribute_single request specifying the bus-off counter attribute. The set_attribute_single data is not used and can be any value.

5) **Allocation information**

The allocation information attribute shall be supported if the predefined master/slave connection set is supported. It indicates whether or not the predefined master/slave connection set defined in 5.5 has been allocated. If it has been allocated, then this attribute indicates the device that has performed the allocation and the connection(s) that are currently allocated.

This attribute is modified when a success response associated with an allocate_master/slave_connection_set service (defined in 5.5) is generated. This attribute is not modifiable by the set_attribute_single service. An error response whose general error code field is set to 0E_{hex} (attribute not settable) is returned if a set_attribute_single request specifies this attribute.

The allocation information attribute consists of the following:

– **allocation choice byte**

The allocation choice byte indicates which of the predefined master/slave connections are active (in the configuring or established state). Its format is specified in 5.3.3.5.2.1.2.

The allocation choice byte is initialised to 00 at device power-up or reset.

– **master's MAC ID**

The master's MAC ID contains the MAC ID of the device that has allocated the predefined master/slave connection set via the allocate_master/slave_connection_set service. This contains the allocator's MAC ID field copied from the allocate_master/slave_connection_set request.

La plage de valeurs est de 0-63, plus 255, en format décimal. Une valeur située dans la plage 0-63 indique que le jeu de connexions maître/esclave prédéfini est actuellement attribué et indique le MAC ID de l'appareil responsable de l'attribution. La valeur 255 signifie que le jeu de connexions maître/esclave prédéfini n'a pas été attribué.

L'attribut MAC ID du maître est initialisé à 255 à la mise sous tension/remise à zéro de l'appareil.

1.3.3.5 Services d'instance de l'objet DeviceNet

Les paragraphes qui suivent décrivent les services communs et les services spécifiques à la classe acceptés par une instance d'objet DeviceNet.

1.3.3.5.1 Services communs

L'instance d'objet DeviceNet accepte les services communs indiqués au tableau 29.

Tableau 29 – Services communs de l'instance d'objet DeviceNet

Mise en oeuvre	Nom du service	Description du service
Requis	Get_attribute_single	Lit la valeur d'un attribut de l'objet DeviceNet
En option	Set_attribute_single	Modifie la valeur d'un attribut de l'objet DeviceNet
NOTE Les codes des services sont définis à l'annexe A.		

1.3.3.5.2 Services spécifiques à la classe

L'instance d'objet DeviceNet doit accepter les services spécifiques à la classe indiqués dans le tableau 30.

Tableau 30 – Services spécifiques à une classe de l'objet DeviceNet

Code de service	Mise en oeuvre	Nom du service	Description du service
4B _{hex}	En option	Allocate_master/slave_connection_set	Demande l'utilisation du jeu de connexions maître/esclave prédéfini
4C _{hex}	En option	Release_master/slave_connection_set	Indique que les connexions spécifiées dans le jeu de connexion maître/esclave prédéfini doivent être supprimées

Ces services sont utilisés pour accorder et retirer l'attribution du jeu de connexions maître/esclave prédéfini décrit en 1.5.

Un appareil qui se comporte comme un client pour le jeu de connexions maître/esclave prédéfini est appelé maître. Un appareil qui se comporte comme un serveur pour le jeu de connexions maître/esclave prédéfini est appelé esclave.

Avant qu'un appareil puisse fonctionner en tant que maître, il doit d'abord attribuer le jeu de connexions maître/esclave prédéfini de l'esclave. Un seul maître peut se voir attribuer le jeu de connexions maître/esclave prédéfini à un moment donné. Lorsqu'un maître n'a plus besoin de son esclave, il doit libérer toutes les connexions; l'esclave libère alors le jeu de connexions maître/esclave prédéfini.

The range of values is 0-63 and 255 decimal. A value in the range 0-63 indicates that the predefined master/slave connection set is currently allocated and denotes the MAC ID of the device that performed the allocation. The value 255 means the predefined master/slave connection set has not been allocated.

The master's MAC ID attribute is initialised to 255 at device power-up/reset.

5.3.3.5 DeviceNet object instance services

The subclauses that follow describe the common services and object class-specific services supported by the DeviceNet object instance.

5.3.3.5.1 Common services

The DeviceNet object instance shall support the common services shown in table 29.

Table 29 – DeviceNet object instance common services

Need in implementation	Service name	Service description
Required	Get_attribute_single	Reads a DeviceNet object attribute value
Optional	Set_attribute_single	Modifies a DeviceNet object attribute value
NOTE Service codes are defined in annex A.		

5.3.3.5.2 Object class-specific services

The DeviceNet object instance shall support the object class-specific services shown in table 30.

Table 30 – DeviceNet object class-specific services

Service code	Need in implementation	Service name	Service description
4B _{hex}	Optional	Allocate_master/slave_connection_set	Requests the use of the predefined master/slave connection set
4C _{hex}	Optional	Release_master/slave_connection_set	Indicates that the specified connections within the predefined master/slave connection set shall be deleted

These services are used to allocate and deallocate the predefined master/slave connection set described in 5.5.

A device that behaves as the client across the predefined master/slave connection set is referred to as a master. A device that behaves as the server across the predefined master/slave connection set is referred to as a slave.

Before a device can function as a master, it shall first allocate the predefined master/slave connection set within the slave. Only one master shall have the predefined master/slave connection set allocated at any given time. When a master no longer requires its slave, it shall release all connections, causing the slave to deallocate the predefined master/slave connection set.

1.3.3.5.2.1 Allocate_master/slave_connection_set (code de service: 4B_{hex})

1.3.3.5.2.1.1 Généralités

Ce service attribue le jeu de connexions maître/esclave prédéfini. Les codes d'erreurs généraux sont définis à l'annexe B. Les valeurs des codes d'erreurs supplémentaires spécifiques à l'objet DeviceNet sont définies en 1.3.3.6. Ce service doit être transmis au moyen du message de demande explicite non connectée du jeu de connexions maître/esclave (voir 1.5.2) ou bien via une connexion de messagerie explicite.

Le service allocate_master/slave_connection_set effectue les tâches suivantes:

- création d'un objet connexion;
- configuration d'un objet connexion.

Voir 5.3.4 et 5.3.5 pour les spécifications d'essais concernant l'attribution du jeu de connexions maître/esclave pour les connexions d'E/S et de messagerie explicite.

1.3.3.5.2.1.2 Paramètres du champ des données de service d'une demande

Les informations du tableau 31 sont placées dans le champ des données de service d'une demande allocate_master/slave_connection_set.

Tableau 31 – Paramètres du champ des données de service d'une demande allocate_master/slave_connection_set

Nom	Type de données	Description du paramètre
Choix d'attribution	BYTE	Indique quelles connexions du jeu maître/esclave prédéfini doivent être attribuées/configurées pour une utilisation par le maître
MAC ID de l'appareil demandeur	USINT	Contient le MAC ID associé à la station demandant l'attribution

Le paramètre choix d'attribution est spécifié dans un seul octet (voir figure 26). Chaque bit correspond ou bien à une ou à des connexions de messagerie explicite et/ou d'E/S du jeu de connexions maître/esclave prédéfini qui peuvent être attribuées, ou bien, dans le cas de la suppression d'acquittement, à une commande. Si le bit est mis à 1, une demande d'attribution est faite pour la connexion correspondante. Si le bit est mis à 0, le demandeur ne veut pas attribuer la connexion correspondante.

7	6	5	4	3	2	1	0
Réservé	Suppression d'acquittement	Cyclique	Changement d'état	Réservé	Requête multiple	Interrogation	Message explicite

IEC 1229/2000

Figure 26 – Contenu de l'octet de choix d'attribution

Les bits 3 et 7 doivent être mis à 0 et l'esclave doit vérifier cette exigence lorsqu'il reçoit la demande allocate_master/slave_connection_set.

La figure 27 indique le format de ce message explicite.

5.3.3.5.2.1 Allocate_master/slave_connection_set (service code: 4B_{hex})

5.3.3.5.2.1.1 General

This service allocates the predefined master/slave connection set. General error codes are defined in annex B. DeviceNet object-specific additional error code values are defined in 5.3.3.6. This service shall be transmitted using either the unconnected explicit request message of the master/slave connection set (see 5.5.2) or an explicit messaging connection.

The allocate_master/slave_connection_set service performs the following:

- connection object create;
- connection object configure.

See 9.3.4 and 9.3.5 for test specifications regarding the allocation of the master/slave connection set for both explicit and I/O messaging connections.

5.3.3.5.2.1.2 Request service data field parameters

The information in table 31 is specified within the service data field of an allocate_master/slave_connection_set request.

Table 31 – Allocate_master/slave_connection_set request service data field parameters

Name	Data type	Description of parameter
Allocation choice	BYTE	Indicates which connections from the predefined master/slave connection set are to be allocated/configured for use by the master
Allocator's MAC ID	USINT	Contains the MAC ID associated with the node requesting the allocation

The allocation choice parameter is specified within a single byte (see figure 26). Each bit denotes an explicit message and/or I/O connection(s) from the predefined master/slave connection set that are to be allocated, or, in the case of an acknowledgement suppression, a command. If the bit is set to 1, then a request is being made to allocate the corresponding connection. If a bit is set to 0, then the requester does not want to allocate the corresponding connection.

7	6	5	4	3	2	1	0
Reserved	Acknowledgement suppression	Cyclic	Change of state	Reserved	Bit strobed	Polled	Explicit message

IEC 1229/2000

Figure 26 – Allocation choice byte contents

Bits 3 and 7 shall be set to 0 and the slave shall verify this requirement when the allocate_master/slave_connection_set request is received.

Figure 27 shows the format of this explicit message.

Contenu								
Rang de l'octet	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (0)	Service (4B _{hex})						
	ID de classe (3)							
	ID d'instance (1)							
	Choix d'attribution							
	0	0	MAC ID de l'appareil demandeur					

IEC 1230/2000

Figure 27 – Message de demande allocate_master/slave_connection_set

Contenu du message de demande allocate_master/slave_connection_set:

- **frag (0)/ID de transaction/MAC ID:** comme défini en 1.2.1.2;
- **bit R/R (0):** indique qu'il s'agit d'un message de demande;
- **code de service (4B_{hex}):** identifie ce message comme étant le service allocate_master/slave_connection_set;
- **ID de classe:** définit la classe d'objet vers laquelle cette demande est dirigée. Lorsque ce message est transmis via l'accès du message de demande explicite non connectée du jeu de connexions maître/esclave (voir 1.5.2), la valeur de l'ID de classe est spécifiée par un nombre entier de 8 bits. L'ID de classe doit être mis à 3;
- **ID d'instance:** définit l'instance particulière dans la classe d'objet vers laquelle cette demande est dirigée. Lorsque ce message est transmis via l'accès du message de demande explicite non connectée du jeu de connexions maître/esclave (voir 1.5.2), la valeur de l'ID d'instance est spécifiée par un nombre entier de 8 bits. L'ID d'instance doit être mis à 1;
- **choix d'attribution:** spécifié dans l'octet qui suit le champ de l'ID d'instance;
- **MAC ID de l'appareil demandeur:** spécifié dans l'octet qui suit le champ choix d'attribution.

1.3.3.5.2.1.3 Paramètres du champ des données de service d'une réponse positive

Les informations contenues dans le champ des données de service d'une réponse positive allocate_master/slave_connection_set sont décrites dans le tableau 32.

Tableau 32 – Paramètres de la réponse allocate_master/slave_connection_set

Nom	Type de données	Description du paramètre
Format du corps du message	Défini ci-dessous	Ce paramètre équivaut sémantiquement au paramètre de format réel de corps du message renvoyé avec une réponse d'ouverture de connexion de messagerie explicite (comme décrit en 1.2.1.6). Cet argument est significatif lorsque la demande allocate_master/slave_connection_set a été reçue par l'accès du message de demande explicite non connectée du jeu maître/esclave (voir 1.5.2). Il indique le format du corps du message associé aux messages ultérieurs transmis via la connexion de messagerie explicite du jeu de connexions maître/esclave prédéfini. Si la demande allocate_master/slave_connection_set a été reçue via une connexion de messagerie explicite d'un appareil ayant la capacité UCMM, ce paramètre est fixé au format réel de corps du message associé à cette connexion de messagerie explicite.

Contents								
Byte offset	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (0)	Service (4B _{hex})						
	Class ID (3)							
	Instance ID (1)							
	Allocation choice							
	0	0	Allocator's MAC ID					

IEC 1230/2000

Figure 27 – Allocate_master/slave_connection_set request message

Allocate_master/slave_connection_set request message contents:

- **frag (0)/transaction ID/MAC ID:** as defined in 5.2.1.2;
- **R/R bit (0):** indicates this is a request message;
- **service code (4B_{hex}):** identifies this message as an allocate_master/slave_connection_set service;
- **class ID:** defines the object class to which this request is directed. When this message is transmitted across the unconnected explicit request message port of the master/slave connection set (see 5.5.2), the class ID value is specified within an 8-bit integer field. The class ID shall be set to 3;
- **instance ID:** defines the particular instance within the object class to which this request is directed. When this message is transmitted across the unconnected explicit request message port of the master/slave connection set (see 5.5.2), the instance ID value is specified within an 8-bit integer field. The instance ID shall be set to 1;
- **allocation choice:** specified by the byte following the instance ID field;
- **allocator's MAC ID:** specified within the byte following the allocation choice field.

5.3.3.5.2.1.3 Success response service data field parameters

The information contained within the service data field of a successful allocate_master/slave_connection_set response is described in table 32.

Table 32 – Allocate_master/slave_connection_set response parameters

Name	Data type	Description of parameter
Message body format	Defined below	This parameter is semantically equivalent to the actual message body format parameter returned with an open explicit message connection response message (as described in 5.2.1.6). This argument is significant when the allocate_master/slave_connection_set request was received across the unconnected explicit request message port of the master/slave connection set (see 5.5.2). It indicates the message body format associated with subsequent messages transmitted across the explicit messaging connection within the predefined master/slave connection set. If the allocate_master/slave_connection_set request was received across an explicit messaging connection within a UCMM capable device, then this parameter is set to the actual message body format associated with that explicit messaging connection.

La figure 28 indique le format d'une réponse positive à la demande `allocate_master/slave_connection_set`.

Contenu								
Rang de l'octet	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (1)	Code de service (4B _{hex})						
2	Réservé (tous les bits = 0)				Format du corps du message			

IEC 1231/2000

Figure 28 – Réponse positive à une demande `allocate_master/slave_connection_set`

Contenu de la réponse positive à la demande `allocate_master/slave_connection_set`:

- **frag (0)/ ID de transaction/MAC ID**: comme indiqué en 1.2.1.2;
- **bit R/R (1)**: indique qu'il s'agit d'un message réponse;
- **code de service (4B_{hex})**: identifie ce message comme étant le service `allocate_master/slave_connection_set`;
- **bits réservés**: ces bits ne sont pas utilisés par le récepteur de la réponse et doivent être mis à 0 par l'émetteur du message réponse;
- **format du corps du message**: comme décrit dans le tableau 32.

1.3.3.5.2.2 Comportement du serveur `allocate_master/slave_connection_set`

- 1) Si une erreur se produit, aucune des connexions demandées ne doit être attribuée. Si cette demande ne peut être entièrement satisfaite, aucune des attributions demandées ne doit être effectuée.
- 2) Si l'appareil de réception ne dispose pas d'un jeu de connexions maître/esclave prédéfini, une réponse d'erreur est renvoyée. Le code d'erreur général de la réponse d'erreur doit être mis à 08 pour indiquer un service non accepté.
- 3) L'appareil de réception (esclave) valide le paramètre MAC ID de l'appareil demandeur dans la demande, de la façon suivante:
 - si le jeu de connexions maître/esclave prédéfini est déjà attribué et si cette demande ne provient pas du maître en cours, l'esclave retourne une erreur. Le code d'erreur général pour la réponse doit être 0C_{hex}, et le code d'erreur supplémentaire doit prendre la valeur 01, spécifique à l'objet;
 - si le jeu de connexions maître/esclave prédéfini n'est pas attribué, l'esclave valide le paramètre choix d'attribution comme indiqué au point 4);
 - si le jeu de connexions maître/esclave prédéfini est déjà attribué et si cette demande provient du maître en cours, l'esclave valide le paramètre choix d'attribution comme indiqué au point 4).
- 4) L'esclave doit valider le paramètre choix d'attribution contenu dans la demande. Si l'esclave ne dispose pas de l'une des connexions spécifiées dans l'argument choix d'attribution, une réponse d'erreur doit être renvoyée. Le code d'erreur général pour la réponse doit être 02, et le code d'erreur supplémentaire mis à la valeur 02, spécifique à l'objet.

Si une ou plusieurs des connexions demandées sont disponibles dans cet esclave mais ont déjà été attribuées au maître indiqué par l'argument MAC ID de l'appareil demandeur, l'esclave doit renvoyer une réponse d'erreur avec le code d'erreur général mis à 0B_{hex}, et le code d'erreur supplémentaire à la valeur 02, spécifique à l'objet. Si la connexion d'E/S demandée est dans l'état délai expiré, l'esclave doit réattribuer la connexion d'E/S, la mettant dans l'état configuration.

Figure 28 shows the format of a success response to the `allocate_master/slave_connection_set` request.

Byte offset	Contents							
	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (1)	Service code (4B _{hex})						
2	Reserved (all bits = 0)				Message body format			

IEC 1231/2000

Figure 28 – Success response to `allocate_master/slave_connection_set` request

`Allocate_master/slave_connection_set` request success response contents:

- **frag (0)/transaction ID/MAC ID:** as defined in 5.2.1.2;
- **R/R bit (1):** indicates this is a response message;
- **service code (4B_{hex}):** identifies this message as an `allocate_master/slave_connection_set` service;
- **reserved bits:** these bits are not used by the receiver of the response and shall be set to 0 by the transmitter of the response message;
- **message body format:** as described in table 32.

5.3.3.5.2.2 `Allocate_master/slave_connection_set` server behaviour

- 1) If an error is encountered, then none of the requested connections shall be allocated. If this request cannot be fully serviced, then none of the requested allocations shall take place.
- 2) If the receiving device does not support the predefined master/slave connection set, then an error response is returned. The general error code within the error response shall be set to 08 to indicate service not supported.
- 3) The receiving device (slave) validates the allocator's MAC ID parameter within the request as follows:
 - if the predefined master/slave connection set is allocated and this request is not from the current master, then the slave returns an error. The general error code within the error response shall be set to 0C_{hex}, with the additional error code set to an object-specific value of 01;
 - if the predefined master/slave connection set is not allocated, then the slave validates the allocation choice parameter as specified in item 4);
 - if the predefined master/slave connection set is allocated and this request is from the current master, then the slave shall validate the allocation choice parameter as specified in item 4).
- 4) The slave shall validate the allocation choice parameter within the request. If the slave does not support one of the connections specified in the allocation choice argument, then an error response shall be returned. The general error code within the error response shall be set to 02, with the additional error code set to an object-specific value of 02.

If any of the connections being requested are supported by this slave and have already been allocated to the master denoted by the allocator's MAC ID argument, the slave shall return an error response with the general error code set to 0B_{hex}, with the additional error code set to an object-specific value of 02. If the requested I/O connection is in the timed-out state, the slave shall reallocate the I/O connection, setting it to the configuring state.

Si l'octet choix d'attribution n'a aucun bit mis à 1, l'esclave doit renvoyer une réponse d'erreur avec le code d'erreur général mis à 09, et le code d'erreur supplémentaire à la valeur 02, spécifique à l'objet.

Si une ressource dont l'utilisation est requise pour les connexions demandées n'est pas disponible, une réponse d'erreur doit être renvoyée avec le code d'erreur général mis à 02, et le code d'erreur supplémentaire à la valeur 04, spécifique à l'objet.

Les choix d'attribution cyclique et de changement d'état s'excluent l'un l'autre. Si une demande d'attribution implique la mise à 1 simultanée des bits d'affectation cyclique et de changement d'état, une réponse d'erreur doit être renvoyée. Le code d'erreur général de la réponse d'erreur doit être mis à 09 (valeur d'attribut incorrecte) et le code d'erreur supplémentaire à 02.

Si un maître a attribué le jeu de connexions cyclique/changement d'état et si une demande d'attribution ultérieure est reçue avec le bit d'attribution interrogation mis à 1, une erreur doit être renvoyée. Le code d'erreur général doit être mis à 02 pour indiquer que la ressource n'est pas disponible. Si le bit de suppression d'acquittement de l'octet est mis à 1 mais pas le bit changement d'état ni le bit cyclique, une réponse d'erreur doit être renvoyée. Le code d'erreur général de la réponse d'erreur doit être mis à 09 et le code d'erreur supplémentaire à 02.

- 5) L'esclave prend note du fait que le jeu de connexions maître/esclave prédéfini a été attribué au MAC ID indiqué dans le champ MAC ID de l'appareil demandeur, en mettant à jour l'attribut informations d'affectation de l'objet DeviceNet. Si nécessaire, les attributs `produced_connection_id` et/ou `consumed_connection_id` de l'objet ou des objets connexions peuvent maintenant être initialisés.

L'octet choix d'attribution de l'attribut informations d'attribution indique quels objets connexions du jeu de connexions maître/esclave prédéfini sont actifs (dans l'état configuration ou établi). Cet octet est mis à jour chaque fois qu'un objet connexion maître/esclave change d'état.

- 6) Les connexions d'E/S attribuées passent à l'état configuration. Pour ce qui est des objets connexions maître/esclave prédéfinis, un service `apply_attributes` implicite accompagne un `set_attribute_single` pour l'attribut `expected_packet_rate` effectué pendant que la connexion est à l'état configuration. Un `set_attribute_single` pour l'attribut `expected_packet_rate` déclenche l'exécution des étapes prévues pour un service `apply_attributes` et entraîne le passage de l'objet connexion maître/esclave prédéfini à l'état établi. Si une erreur se produit lors de l'exécution de `apply_attributes`, l'attribut `expected_packet_rate` doit être remis à sa valeur précédente et une erreur doit être renvoyée, avec le code d'erreur général mis à 09 et le code d'erreur supplémentaire chargé avec l'ID de l'attribut erroné de l'objet connexion.
- 7) Si elle est attribuée, la connexion de messagerie explicite passe à l'état établi. Le temporisateur d'inactivité/surveillance est lancé à l'aide de la valeur initiale spécifiée en 1.3.2.5.3.

Les appareils à capacité UCMM qui disposent du jeu de connexions maître/esclave prédéfini doivent accepter l'attribution et l'utilisation de la connexion de messagerie explicite du jeu de connexions maître/esclave prédéfini.

- 8) La connexion de messagerie explicite du jeu de connexions maître/esclave, ou bien une connexion de messagerie explicite dynamiquement établie, agit comme parent de la ou des connexions d'E/S attribuées jusqu'à ce que cette ou ces connexions d'E/S passent à l'état établi. Le terme parent est utilisé pour indiquer qu'en cas de suppression de la connexion de messagerie explicite alors qu'aucune des connexions d'E/S attribuées n'est à l'état établi, le jeu de connexions maître/esclave prédéfini est automatiquement libéré par l'esclave. La logique qui définit quelle connexion de messagerie explicite doit agir comme parent est définie à la figure 29.

If the allocation choice byte has no bits set, the slave shall return an error response with the general error code set to 09, with the additional error code set to an object-specific value of 02.

If a resource that is required for use with the requested connections is not available, then an error response shall be returned with the general error code set to 02, and the additional error code set to an object-specific value of 04.

The change of state and cyclic allocation choices are mutually exclusive. If an allocation request would result in both the cyclic and change of state bits being set, an error response shall be returned. The general error code within the error response shall be set to 09 (invalid attribute value) with an additional error code of 02.

If a master has allocated the change of state/cyclic connection set and a subsequent allocation request is received with the polled allocation bit set, an error shall be returned. The general error code within the error response shall be set to 02 to indicate resource unavailable. If the allocation choice byte has the acknowledgement suppression bit set and neither the change of state bit nor the cyclic bit are set, an error response shall be returned. The general error code within the error response shall be set to 09 with an additional error code of 02.

- 5) The slave notes the fact that the predefined master/slave connection set has been allocated to the MAC ID within the allocator's MAC ID field by updating the DeviceNet object's allocation information attribute. If necessary, the `produced_connection_id` and/or `consumed_connection_id` attributes of the connection object(s) may now be initialised.

The allocation choice byte of the allocation information attribute indicates which connection objects from the predefined master/slave connection set are active (in the configuring, or established state). This byte is updated whenever a master/slave connection object changes state.

- 6) Any allocated I/O connection(s) transition to the configuring state. With respect to the predefined master/slave connection objects, an implied `apply_attributes` service accompanies a `set_attribute_single` of the `expected_packet_rate` attribute while the connection is in the configuring state. A `set_attribute_single` of the `expected_packet_rate` attribute triggers the execution of the steps performed with an `apply_attributes` service and causes the predefined master/slave I/O connection object to transition to the established state. If an error is encountered when performing the `apply_attributes` portion, then the `expected_packet_rate` attribute shall be reset to its previous value, and an error shall be returned whose general error code is set to 09 and whose additional error code is set to the offending connection object's attribute ID.

- 7) If allocated, the explicit messaging connection transitions to the established state. The inactivity/watchdog timer is started using the initial value specified in 5.3.2.5.3.

UCMM capable devices which support the predefined master/slave connection set shall support the allocation and use of the predefined master/slave connection set explicit messaging connection.

- 8) Either the master/slave connection set's explicit messaging connection or a dynamically established explicit messaging connection acts as the allocated I/O connection(s) parent until the I/O connection(s) are in the established state. The term parent is used to indicate that, if the explicit messaging connection is deleted and none of the allocated I/O connections exist in the established state, then the predefined master/slave connection set is automatically released by the slave. The logic describing which explicit messaging connection is to act as the parent is defined in figure 29.

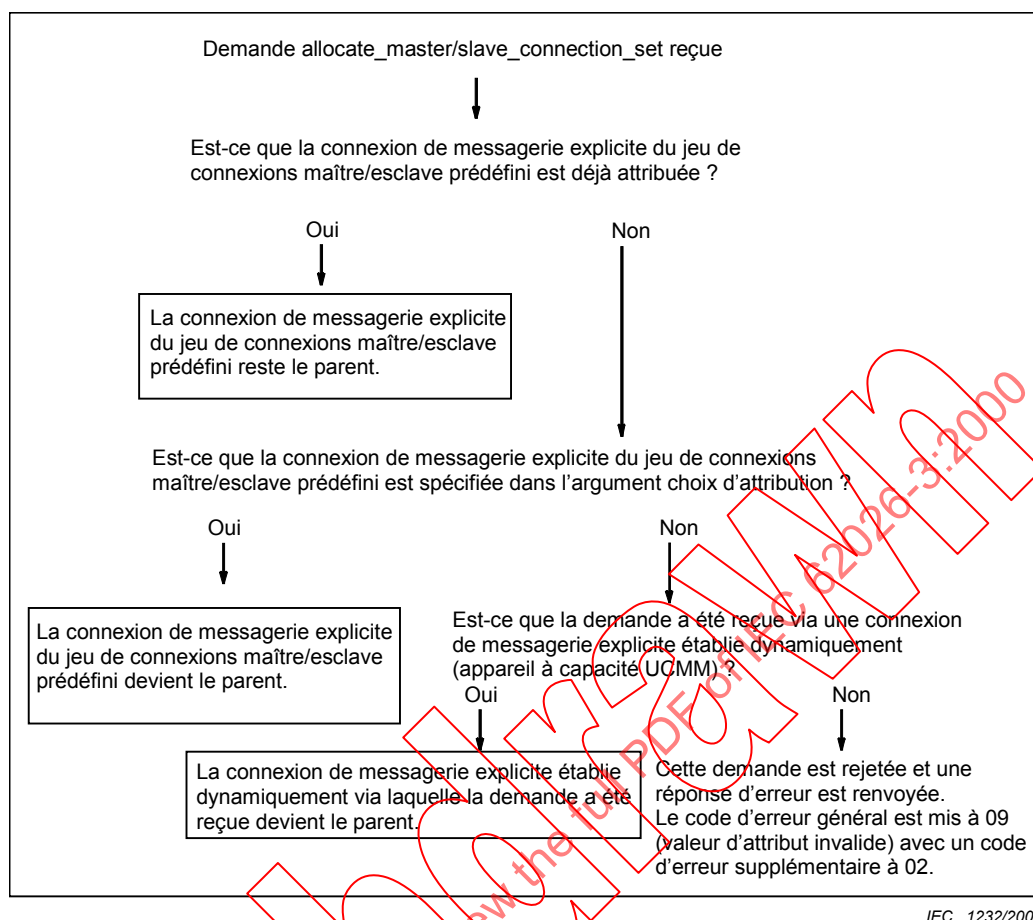


Figure 29 – Choix d'une connexion de messagerie explicite parent

L'esclave libère automatiquement le jeu de connexions maître/esclave prédéfini et toutes les connexions repassent à l'état non existant si toutes les conditions suivantes sont vraies:

- aucun des objets connexion associés au jeu de connexions maître/esclave prédéfini n'existe à l'état établi,
- l'objet connexion de messagerie explicite parent n'est pas à l'état établi.

1.3.3.5.2.3 Release_master/slave_connection_set (code de service: 4C_{hex})

1.3.3.5.2.3.1 Généralités

Ce service est utilisé pour libérer le jeu de connexions maître/esclave prédéfini dans un esclave. Les codes d'erreurs généraux sont définis à l'annexe B. Les codes d'erreurs supplémentaires, spécifiques aux objets DeviceNet, sont définis en 5.3.3.6. Ce service peut être transmis via l'accès du message de demande explicite non connectée du jeu de connexions maître/esclave (voir 5.5.2), de même que via une connexion de messagerie explicite.

1.3.3.5.2.3.2 Paramètres du champ des données de service d'une demande

Les informations du tableau 33 sont spécifiées dans le champ des données de service d'une demande release_master/slave_connection_set.

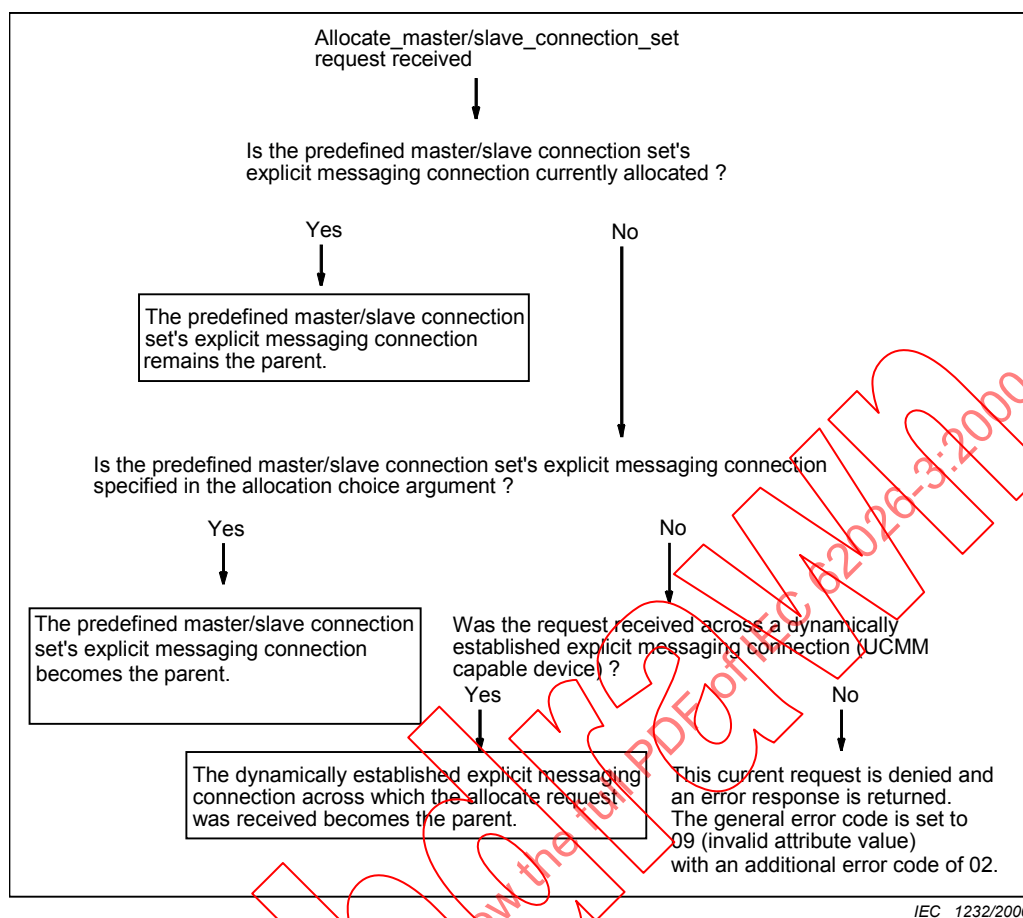


Figure 29 – Parent explicit messaging connection logic

The slave automatically releases the predefined master/slave connection set and all connections return to the non-existent state if all of the following conditions are true:

- none of the connection objects associated with the predefined master/slave connection set exist in the established state;
- the parent explicit messaging connection object is not in the established state.

5.3.3.5.2.3 Release_master/slave_connection_set (service code: 4C_{hex})

5.3.3.5.2.3.1 General

This service is used to deallocate the predefined master/slave connection set within a slave. General error codes are defined in annex B. DeviceNet object-specific additional error code values are defined in 5.3.3.6. This service can be transmitted across the unconnected explicit request message port of the master/slave connection set (see 5.5.2) as well as across an explicit messaging connection.

5.3.3.5.2.3.2 Request service data field parameters

The information in table 33 is specified within the service data field of a release_master/slave_connection_set request.

Tableau 33 – Paramètres du champ des données de service d'une demande release_master/slave_connection_set

Nom	Type de données	Description du paramètre
Choix de libération	BYTE	Indique quelles connexions maître/esclave prédéfinies doivent être libérées

Le paramètre choix de libération est spécifié dans un seul octet (voir figure 30). Chaque bit indique une ou des connexions de messagerie explicite et/ou d'E/S à libérer. Si le bit est mis à 1, une demande de libération est faite pour la connexion correspondante. Si le bit est mis à 0, le demandeur ne veut pas libérer la connexion correspondante.

7	6	5	4	3	2	1	0
Réservé	Suppression d'acquittement	Cyclique	Changement d'état	Réservé	Requête multiple	Interrogation	Message explicite

IEC 1233/2000

Figure 30 – Contenu de l'octet de choix de libération

Les bits 3 et 7 doivent être mis à 0 et l'esclave doit vérifier cette exigence lorsqu'il reçoit la demande release_master/slave_connection_set.

La valeur 00 n'est pas valide.

La figure 31 indique le format de ce message explicite.

	Contenu							
Rang de l'octet	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (0)	Service (4C _{hex})						
	ID de classe (3)							
	ID d'instance (1)							
	Choix de libération							

IEC 1234/2000

Figure 31 – Message de demande release_master/slave_connection_set

Contenu du message de demande release_master/slave_connection_set:

- **frag (0)/ID, ID de transaction/MAC ID:** voir 1.2.1.2;
- **bit R/R (0):** indique qu'il s'agit d'un message de demande;
- **code de service (4C_{hex}):** identifie ce message comme étant le service release_master/slave_connection_set;
- **ID de classe:** définit la classe de l'objet vers laquelle cette demande est dirigée. Lorsque ce message est transmis via l'accès du message de demande explicite non connectée du jeu de connexions maître/esclave (voir 1.5.2), la valeur de l'ID de classe est spécifiée dans un nombre entier de 8 bits. L'ID de classe doit être sur 3;
- **ID d'instance:** définit l'instance particulière dans la classe d'objet vers laquelle cette demande est dirigée. Lorsque ce message est transmis via l'accès du message de demande explicite non connectée du jeu de connexions maître/esclave (voir 1.5.2), la valeur de l'ID d'instance est spécifiée dans un nombre entier de 8 bits. L'ID d'instance doit être mis à 1;
- **choix de libération:** spécifié dans l'octet qui suit le champ de l'ID d'instance.

Table 33 – Release_master/slave_connection_set request service data field parameters

Name	Data type	Description of parameter
Release choice	BYTE	Indicates which predefined master/slave connections are to be released

The release choice parameter is specified within a single byte (see figure 30). Each bit denotes an explicit message and/or I/O connection(s) to be released. If the bit is set to 1, then a request is being made to release the corresponding connection. If a bit is set to 0, then the requester does not want to release the corresponding connection.

7	6	5	4	3	2	1	0
Reserved	Acknowledgement suppression	Cyclic	Change of state	Reserved	Bit strobed	Polled	Explicit message

IEC 1233/2000

Figure 30 – Release choice byte contents

Bits 3 and 7 shall be set to 0 and the slave shall verify this requirement when the release_master/slave_connection_set request is received.

A value of 00 is invalid.

Figure 31 shows the format of this explicit message.

Contents	
Byte offset	7 6 5 4 3 2 1 0
0	Frag (0) XID MAC ID
1	R/R (0) Service (4C _{hex})
	Class ID (3)
	Instance ID (1)
	Release choice

IEC 1234/2000

Figure 31 – Release_master/slave_connection set request message

Release_master/slave_connection set request message contents:

- **frag (0)/transaction ID/MAC ID:** see 5.2.1.2;
- **R/R bit (0):** indicates this is a request message;
- **service code (4C_{hex}):** identifies this message as a release_master/slave_connection_set service;
- **class ID:** defines the object class to which this request is directed. When this message is transmitted across the unconnected explicit request message port of the master/slave connection set (see 5.5.2), the class ID value is specified within an 8-bit integer field. The class ID shall be set to 3;
- **instance ID:** defines the particular instance within the object class to which this request is directed. When this message is transmitted across the unconnected explicit request message port of the master/slave connection set (see 5.5.2), the instance ID value is specified within an 8-bit integer field. The instance ID shall be set to 1;
- **release choice:** specified within the byte following the instance ID field.

1.3.3.5.2.3.3 Message réponse positive

La figure 32 indique le format d'une réponse positive à la demande `release_master/slave_connection_set`.

		Contenu							
Rang de l'octet		7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID						
1	R/R (1)	Service (4C _{hex})							

IEC 1235/2000

Figure 32 – Réponse positive à une demande `release_master/slave_connection_set`

Contenu de la réponse positive à la demande `release_master/slave_connection_set`:

- **frag (0)/ID, ID de transaction MAC ID:** voir 1.2.1.2;
- **bit R/R (1):** indique qu'il s'agit d'un message réponse;
- **code de service (4C_{hex}):** identifie ce message comme étant le service `release_master/slave_connection_set`.

1.3.3.5.2.4 Comportement du serveur `release_master/slave_connection_set`

Si une erreur se produit, aucune des connexions spécifiées n'est libérée. Si cette demande ne peut être entièrement satisfaite, aucune des libérations demandées n'est effectuée.

Si l'appareil de réception ne dispose pas du jeu de connexions maître/esclave prédéfini, une réponse d'erreur doit être renvoyée. Le code d'erreur général de la réponse d'erreur doit être à 08.

Le dispositif de réception (esclave) doit valider le paramètre choix de libération dans la demande. Si l'esclave ne dispose pas de l'une des connexions spécifiées dans l'argument choix de libération, une réponse d'erreur doit être renvoyée. Le code d'erreur général de la réponse d'erreur doit être mis à 02, et le code d'erreur supplémentaire à la valeur 02, spécifique à l'objet.

Si l'octet choix de libération n'a aucun bit mis à 1 (tous les bits sont à 0), l'esclave doit renvoyer une réponse d'erreur avec le code d'erreur général mis à 09 (valeur d'attribut incorrecte), et le code d'erreur supplémentaire à la valeur 02, spécifique à l'objet.

Si l'une des connexions spécifiées est dans l'état non existant, une réponse d'erreur doit être renvoyée. Le code d'erreur général de la réponse d'erreur doit être mis à 0B_{hex}.

L'esclave doit vérifier qu'il est dans un état qui lui permet de ne plus utiliser la ou les connexions spécifiées. Si ce n'est pas le cas, une réponse d'erreur doit être renvoyée. Le code d'erreur général de la réponse d'erreur doit être mis à 0C_{hex}.

Si la demande est valide, l'esclave doit libérer toutes les ressources associées à la ou aux connexions spécifiées. Si cela a pour conséquence la libération de toutes les connexions maître/esclave prédéfinies, l'esclave doit prendre note du fait que le jeu de connexions maître/esclave prédéfini n'est plus attribué en mettant à jour l'attribut informations d'attribution.

L'esclave ne doit pas vérifier si la demande de libération vient de son maître.

Si, à la suite de cette demande, aucune des connexions maître/esclave prédéfinies n'existe à l'état établi, l'esclave doit libérer le jeu de connexions maître/esclave prédéfini et toutes les connexions doivent retourner à l'état non existant.

5.3.3.5.2.3.3 Success response message

Figure 32 shows the format of a success response to the release_master/slave_connection_set request.

Contents								
Byte offset	7	6	5	4	3	2	1	0
0	Frag (0)	XID	MAC ID					
1	R/R (1)	Service (4C _{hex})						

IEC 1235/2000

Figure 32 – Success response to release_master/slave_connection_set request

Release_master/slave_connection_set request success response contents.

- **frag (0)/transaction ID/MAC ID:** see 5.2.1.2;
- **R/R bit (1):** indicates this is a response message;
- **service code (4C_{hex}):** identifies this as a release_master/slave_connection_set service.

5.3.3.5.2.4 Release_master/slave_connection_set server behaviour

If an error is encountered, then none of the specified connections are released. If this request cannot be fully serviced, then none of the requested releases take place.

If the receiving device does not support the predefined master/slave connection set, then an error response shall be returned. The general error code within the error response shall be set to 08.

The receiving device (slave) shall validate the release choice parameter within the request. If the slave does not support one of the connections specified in the release choice argument, then an error response shall be returned. The general error code within the error response shall be set to 02, with the additional error code set to an object-specific value of 02.

If the release choice byte has no bits set (all bits are 0), the slave shall return an error response with the general error code set to 09 (invalid attribute value), with the additional error code set to an object-specific value of 02.

If one of the specified connections is in the non-existent state, then an error response shall be returned. The general error code within the error response shall be set to 0B_{hex}.

The slave shall verify that it is in a state that allows it to discontinue use of the specified connection(s). If this is not the case, then an error response shall be returned. The general error code within the error response shall be set to 0C_{hex}.

If the request is valid, then the slave shall release all resources associated with the specified connection(s). If this results in all predefined master/slave connections being released, the slave shall note the fact that the predefined master/slave connection set is no longer allocated by updating the allocation information attribute.

The slave shall not check to see that the release request came from its master.

If this request has resulted in none of the predefined master/slave connections existing in the established state, then the slave shall release the predefined master/slave connection set and all connections shall return to the non-existent state.

1.3.3.6 Codes d'erreur spécifiques à l'objet DeviceNet

Le tableau 34 donne la liste des codes d'erreurs supplémentaires spécifiques à l'objet DeviceNet.

Tableau 34 – Codes d'erreurs supplémentaires spécifiques à l'objet DeviceNet

Valeur	Signification
01	Conflit d'attribution pour le jeu de connexions maître/esclave prédéfini. Renvoyé lorsqu'une demande allocate_master/slave_connection_set est reçue et que l'esclave a déjà attribué le jeu de connexions maître/esclave prédéfini à un autre maître
02	Paramètre choix d'attribution/libération incorrect. Renvoyé lorsqu'une demande allocate/release_master/slave_connection_set est reçue et que: 1) l'esclave n'accepte pas le choix spécifié dans le paramètre choix; 2) l'esclave a reçu la demande d'attribuer/de libérer une ou des connexions déjà attribuées/libérées; 3) l'octet choix d'attribution/libération ne contenait que des zéros, une combinaison de bits incorrecte, ou ne spécifiait pas d'attribution de message explicite lorsque c'était nécessaire
03	Un serveur qui n'a pas la capacité UCMM a reçu un message qui n'était pas un message d'attribution ou de libération sur l'accès du message de demande explicite non connecté du jeu de connexions maître/esclave (voir 1.5.2)
04	La ressource dont l'utilisation est requise pour le jeu de connexions maître/esclave prédéfini n'est pas disponible

1.4 Machine d'état d'accès à la liaison

1.4.1 Généralités

Le présent paragraphe définit la machine d'état d'accès à la liaison que chaque station DeviceNet doit mettre en oeuvre. La machine d'état d'accès à la liaison est décrite par les critères suivants:

- tâches devant être effectuées avant de communiquer à travers le CDI;
- événements de la liaison affectant les possibilités de communication d'une station à travers le CDI.

1.4.2 Diagramme de la machine d'état et table états-événements

La figure 33 fournit un aperçu général de la machine d'état d'accès à la liaison.

5.3.3.6 Error codes specific to the DeviceNet object

Table 34 lists additional error codes specific to the DeviceNet object.

Table 34 – DeviceNet object-specific additional error codes

Value	Meaning
01	Predefined master/slave connection set allocation conflict. This is returned when an allocate_master/slave_connection_set request is received and the slave has already allocated the predefined master/slave connection set to another master
02	Invalid allocation/release choice parameter. This is returned when an allocate/release_master/slave_connection_set request is received and: 1) the slave does not support the choice specified in the choice parameter; 2) the slave was asked to allocate/release connection(s) already allocated/released; 3) the allocation choice/release byte contained all zeros, an invalid combination of bits, or did not contain the explicit message allocation choice when required
03	A server that does not support UCMM received a message that was not an allocate or release message on the unconnected explicit request message port of the master/slave connection set (see 5.5.2)
04	Resource required for use with the predefined master/slave connection set is unavailable

5.4 Link access state machine

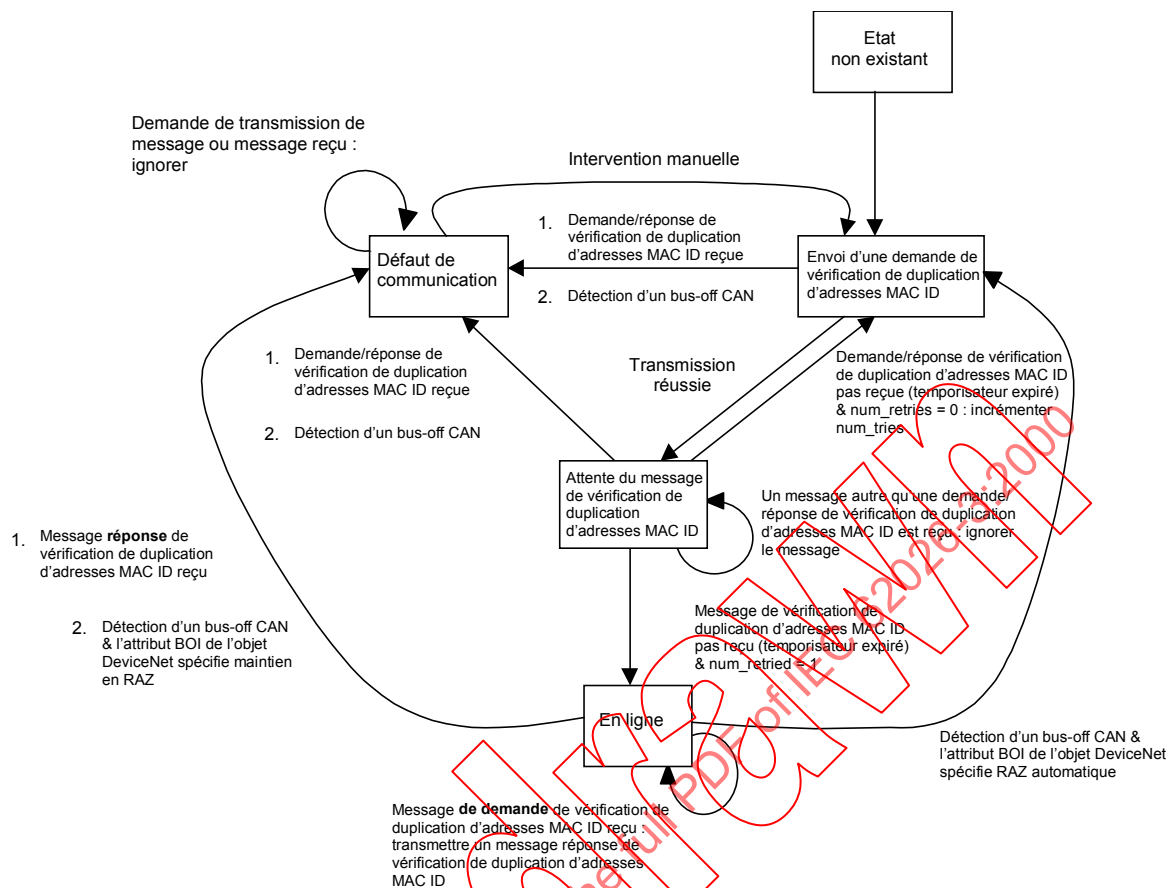
5.4.1 General

This subclause defines the link access state machine that every DeviceNet node shall implement. The link access state machine is described by the following:

- tasks that shall be performed prior to communicating through the CDI;
- link events that affect a node's ability to communicate through the CDI.

5.4.2 State transition diagram and state-event matrix

Figure 33 provides a general overview of the link access state machine.



IEC 1236/2000

Figure 33 – Diagramme de la machine d'état d'accès à la liaison

Le tableau 35 détaille la table états-événements pour la machine d'état d'accès à la liaison.

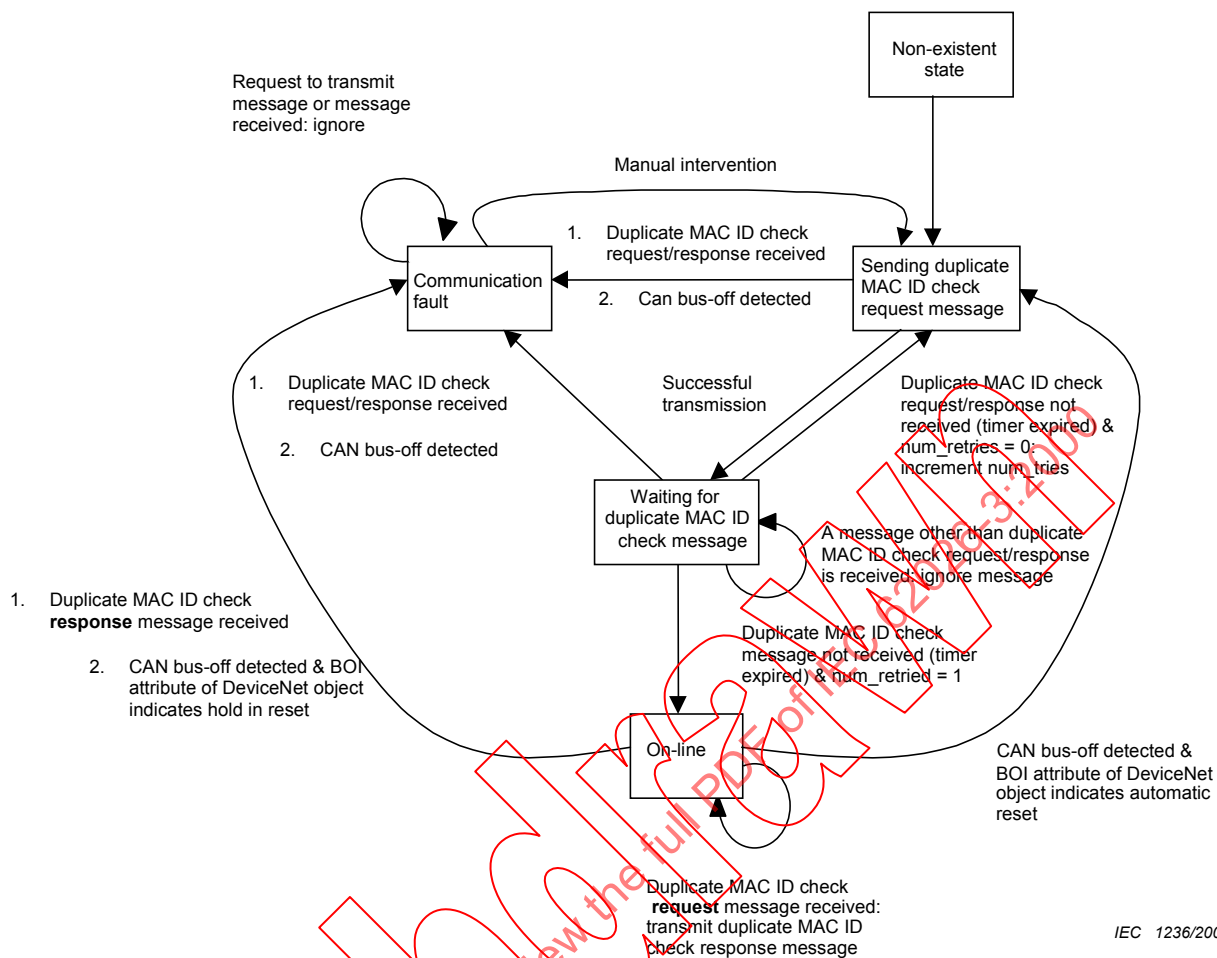


Figure 33 – Link access state transition diagram

Table 35 provides a detailed state-event matrix for the link access state machine.

Tableau 35 – Table états-événements d'accès à la liaison

Etat				
Evénement	Envoi d'une demande de vérification de duplication d'adresses MAC ID	Attente du message de vérification de duplication d'adresses MAC ID	En ligne	Défaut de communication
Transmission réussie du message de demande de vérification de duplication d'adresses MAC ID	Activer le temporisateur d'1s ¹⁾ Transition vers l'attente du message de vérification de duplication d'adresses MAC ID	Sans objet	Sans objet	Sans objet
Détection bus-off	Transcepteur maintenu en RAZ Transition vers l'état défaut de communication	Transcepteur maintenu en RAZ Transition vers l'état défaut de communication	Accéder à l'attribut BOI de l'objet DeviceNet Si l'attribut BOI indique que le transcepteur doit être maintenu en RAZ, passer en défaut de communication Si l'attribut BOI indique que le MAC et le transcepteur doivent être automatiquement remis à zéro, procéder comme suit: 1) remettre à zéro, 2) demander la transmission du message de demande de vérification de duplication d'adresses MAC ID, et 3) passer à l'état envoi d'une demande de vérification de duplication d'adresses MAC ID	Sans objet
Message de demande de vérification de duplication d'adresses MAC ID reçu	Duplication d'adresses MAC ID détectée Transition vers l'état défaut de communication	Duplication d'adresses MAC ID détectée Transition vers l'état défaut de communication	Transmettre le message réponse de vérification de duplication d'adresses MAC ID	Ne pas tenir compte du message
Message réponse de vérification de duplication d'adresses MAC ID reçu	Duplication d'adresses MAC ID détectée Transition vers l'état défaut de communication	Duplication d'adresses MAC ID détectée Transition vers l'état défaut de communication	Duplication d'adresses MAC ID détectée Transition vers l'état défaut de communication	Ne pas tenir compte du message
Le temporisateur d'1 s du message de vérification de duplication d'adresses MAC ID expire ¹⁾	Sans objet	S'il s'agit du premier délai expiré, demander de nouveau la transmission d'un message de demande de vérification de duplication d'adresses MAC ID et passer à l'envoi de la demande de vérification de duplication d'adresses MAC ID S'il s'agit du deuxième délai expiré consécutif, passer en ligne	Sans objet	Sans objet
Demande de transmission de message interne	Renvoyer une erreur interne	Renvoyer une erreur interne	Transmettre le message	Renvoyer une erreur interne
Un message autre qu'une demande/ réponse de vérification de duplication d'adresses MAC ID est reçu	Ne pas tenir compte du message	Ne pas tenir compte du message	Traiter le message reçu comme il se doit	Ne pas tenir compte du message

¹⁾ Plage de validité pour le temporisateur: de 0,9 s à 1,5 s.

Table 35 – Link access state-event matrix

Event	State			
	Sending duplicate MAC ID check request	Waiting for duplicate MAC ID check message	On-line	Communication fault
Successful transmission of the duplicate MAC ID check request message	Activate 1 s timer ¹⁾ Transition to waiting for duplicate MAC ID check message	Not applicable	Not applicable	Not applicable
Bus-off detected	Transceiver held in reset Transition to the communication fault state	Transceiver held in reset Transition to the communication fault state	Access the DeviceNet object's BOI attribute If the BOI attribute indicates that the transceiver shall be held in reset, then transition to communication fault If the BOI attribute indicates that the MAC and transceiver shall be automatically reset, then: 1) reset; 2) request the transmission of the duplicate MAC ID check request message; and 3) transition to the sending duplicate MAC ID check request state	Not applicable
Duplicate MAC ID check request message received	Duplicate MAC ID detected Transition to the communication fault state	Duplicate MAC ID detected Transition to the communication fault state	Transmit the duplicate MAC ID check response message	Discard message
Duplicate MAC ID check response message received	Duplicate MAC ID detected Transition to the communication fault state	Duplicate MAC ID detected Transition to the communication fault state	Duplicate MAC ID detected Transition to the communication fault state	Discard message
1 s duplicate MAC ID check message timer expires ¹⁾	Not applicable	If this is the first time-out, then request the transmission of the duplicate MAC ID check request message again and transition to sending duplicate MAC ID check request If this is the second consecutive time-out, then transition to on-line	Not applicable	Not applicable
Internal message transmission request	Return internal error	Return internal error	Transmit message	Return internal error
A message other than duplicate MAC ID check request/response is received	Discard message	Discard message	Process the received message as appropriate	Discard message
¹⁾ Valid timer range is 0,9 s to 1,5 s.				

La machine d'état de gestion des erreurs du protocole CAN prévoit que, pendant la mise en route/remise en route du système, une seule station puisse être présente sur la liaison. Si cette station transmet un message, elle va subir une erreur d'acquittement et répéter automatiquement le message. Dans cette situation, la station va passer dans l'état error passive mais non bus-off. Pour cette raison, le seul événement qui signifie l'échec de transmission d'un message de vérification de duplication d'adresses MAC ID (demande ou réponse) est l'événement bus-off. Si une indication error passive ou error warning est reçue pendant la transmission d'un message de vérification de duplication d'adresses MAC ID, elle doit être ignorée, car elle est sans effet sur la machine d'état de détection de duplication d'adresses MAC ID.

1.4.3 Détection de duplication d'adresses MAC ID

La principale étape de la machine d'état d'accès à la liaison est l'exécution de l'algorithme de détection de duplication d'adresses MAC ID. Chaque station DeviceNet reçoit un MAC ID unique; par mesure de protection contre les erreurs, toutes les stations DeviceNet doivent participer à l'algorithme de détection de duplication d'adresses MAC ID.

NOTE Le protocole associé à l'algorithme de détection de duplication d'adresses MAC ID est décrit en 1.2.4.

Comme indiqué en 1.2.4, un message spécial du groupe 2 est réservé pour l'exécution de la détection de duplication d'adresses MAC ID.

Une station DeviceNet doit recevoir et traiter tout message de vérification de duplication d'adresses MAC ID qui contient son propre MAC ID dans le champ identificateur du groupe de messages 2.

Après la transmission d'un message de demande de vérification de duplication d'adresses MAC ID, un appareil doit attendre 1 s avant de décréter l'expiration du délai et d'entreprendre l'action appropriée définie par la machine d'état d'accès à la liaison.

Le message de demande de vérification de duplication d'adresses MAC ID doit être transmis deux fois sans recevoir en retour aucun message de demande ou réponse de vérification de duplication d'adresses MAC ID avant de pouvoir passer en ligne.

Voir 5.2.3 et 5.3.2 pour les spécifications d'essais concernant le comportement à la mise en route et la gestion des mécanismes liés à la duplication d'adresses MAC ID.

1.5 Jeu de connexions maître/esclave prédéfini

1.5.1 Généralités

Les paragraphes précédents décrivent les règles générales d'établissement de connexions entre appareils. Les règles générales prescrivent l'utilisation d'une connexion de messagerie explicite pour créer et configurer les objets connexions à chaque extrémité de la connexion. Le présent paragraphe utilise les règles générales comme base de la définition d'un jeu de connexions qui facilite les communications typiques d'une relation maître/esclave. Ces connexions sont collectivement appelées jeu de connexions maître/esclave prédéfini.

- **Serveur du groupe 2:** Appareil ayant la capacité de gestionnaire de messages non connectés (UCMM), configuré pour agir comme serveur des connexions maître/esclave prédéfinies.
- **Client du groupe 2:** Appareil ayant la capacité UCMM et ayant obtenu la propriété du jeu de connexions maître/esclave prédéfini dans un serveur, de sorte qu'il peut agir en tant que client sur ces connexions.
- **Appareil à capacité UCMM:** Appareil qui dispose de la fonctionnalité UCMM.
- **Appareil sans capacité UCMM:** Appareil qui ne dispose pas de la fonctionnalité UCMM.

The CAN fault confinement state machine considers the possibility that, during the system start-up/wake-up, only one node may be present on the link. If this node transmits a message, it will experience an acknowledgement error and will automatically repeat the message. In this situation, the node will transition to error passive but not bus-off. For this reason, the only event that signifies an unsuccessful transmission of a duplicate MAC ID check message (request or response) is the bus-off event. If an error passive or error warning indication is received during the transmission of a duplicate MAC ID check message, then it shall be ignored, as it has no effect on the duplicate MAC ID detection state machine.

5.4.3 Duplicate MAC ID detection

The main step involved in the link access state machine is the execution of a duplicate MAC ID detection algorithm. Each node on DeviceNet is assigned a unique MAC ID, and to protect against errors, all DeviceNet nodes shall participate in the duplicate MAC ID detection algorithm.

NOTE The protocol associated with the duplicate MAC ID detection algorithm is described in 5.2.4.

As stated in 5.2.4, a special message within message group 2 is defined for performing duplicate MAC ID detection.

A DeviceNet node shall receive and process any duplicate MAC ID check message that contains its MAC ID in the message group 2 identifier field.

After transmitting a duplicate MAC ID check request message, a module shall wait 1 s before timing out and taking the appropriate action defined by the link access state machine.

The duplicate MAC ID check request message shall be transmitted twice without receiving a subsequent duplicate MAC ID check request or response message before transitioning to on-line.

See 9.2.3 and 9.3.2 for test specifications regarding the power ON behaviour and handling of the duplicate MAC ID mechanisms.

5.5 Predefined master/slave connection set

5.5.1 General

The preceding subclauses present the general rules for establishing connections between devices. The general rules call for the utilisation of an explicit messaging connection to create and configure connection objects within each connection end-point. This subclause uses the general rules as a basis for the definition of a set of connections which facilitates communications typically seen in a master/slave relationship. These connections are referred to collectively as the predefined master/slave connection set.

- **Group 2 server:** An unconnected message manager (UCMM) capable device that has been configured to act as the server for the predefined master/slave identifier connections.
- **Group 2 client:** A 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.
- **UCMM capable device:** A device that supports the UCMM.
- **UCMM incapable device:** A device that does not support the UCMM.

- **Serveur du groupe 2 seulement:** Appareil esclave sans capacité UCMM et qui utilise le jeu de connexions maître/esclave prédéfini pour établir des communications. Un appareil du groupe 2 seulement ne peut transmettre et recevoir qu'avec les identificateurs spécifiés par le jeu de connexions maître/esclave prédéfini.
- **Client de groupe 2 seulement:** Appareil agissant comme client du groupe 2 envers un serveur du groupe 2 seulement. Le client de groupe 2 seulement fournit la fonctionnalité UCMM aux serveurs du groupe 2 seulement qui lui sont alloués.

1.5.2 Messages du jeu de connexions maître/esclave prédéfini

Les champs identificateurs de CAN associés au jeu de connexions maître/esclave prédéfini sont indiqués dans le tableau 36, en même temps que les identificateurs qui doivent être utilisés pour toute messagerie connectée utilisée dans le cadre du jeu de connexions maître/esclave prédéfini.

Tableau 36 – Champs identificateurs pour le jeu de connexions maître/esclave prédéfini

Bits d'identificateur											Usage	Plage
10	9	8	7	6	5	4	3	2	1	0		
0	ID des messages du groupe 1				MAC ID source						Messages du groupe 1	000-3FF
0	1	1	0	1	MAC ID source						Message d'E/S changement d'état ou cyclique de l'esclave	
0	1	1	1	0	MAC ID source						Message d'E/S réponse requête multiple de l'esclave	
0	1	1	1	1	MAC ID source						Message d'E/S réponse interrogation ou acquittement ou changement d'état/cyclique de l'esclave	
1	0	MAC ID						ID des messages du groupe 2			Messages du groupe 2	400-5FF
1	0	MAC ID source						0	0	0	Message d'E/S de commande requête multiple du maître	
1	0	MAC ID source						0	0	1	Réservé	
1	0	MAC ID destination						0	1	0	Message d'acquiescement changement d'état/cyclique du maître	
1	0	MAC ID source						0	1	1	Message de réponse explicite de l'esclave	
1	0	MAC ID destination						1	0	0	Message de demande explicite du maître	
1	0	MAC ID destination						1	0	1	Message d'E/S de commande interrogation/ changement d'état/cyclique de l'esclave	
1	0	MAC ID destination						1	1	0	Message de demande explicite non connecté du groupe 2 seulement	
1	0	MAC ID destination						1	1	1	Message de vérification de duplication d'adresses MAC ID	

Les types de messages suivants sont inclus dans le tableau 36:

- **message d'E/S de commande/réponse requête multiple:** la commande requête multiple est un message d'E/S transmis par le maître. De nombreux esclaves peuvent recevoir et réagir à la même commande requête multiple. La réponse requête multiple est un message d'E/S renvoyé par un esclave au maître à la réception d'une commande de requête multiple;
- **message d'E/S de commande/réponse interrogation:** la commande interrogation est un message d'E/S transmis par le maître. Une commande interrogation est dirigée vers un esclave spécifique. La réponse interrogation est un message d'E/S renvoyé par un esclave au maître à la réception d'une commande interrogation;

- **Group 2 only server:** A slave device that is UCMM incapable and uses the predefined master/slave connection set to establish communications. A group 2 only device can transmit and receive only those identifiers defined by the predefined master/slave connection set;
- **Group 2 only client:** A device that is acting as a group 2 client to a group 2 only server. The group 2 only client provides the UCMM functionality for group 2 only servers that are allocated to it.

5.5.2 Predefined master/slave connection set messages

The CAN identifier fields associated with the predefined master/slave connection set are shown in table 36, together with the identifiers that shall be used with all connection-based messaging used in the predefined master/slave connection set.

Table 36 – Predefined master/slave connection set identifier fields

Identifier bits											Usage	Range
10	9	8	7	6	5	4	3	2	1	0		
0	Group 1 message ID				Source MAC ID						Group 1 messages	000-3FF
0	1	1	0	1	Source MAC ID						Slave's I/O change of state or cyclic message	
0	1	1	1	0	Source MAC ID						Slave's I/O bit-strobe response message	
0	1	1	1	1	Source MAC ID						Slave's I/O poll response or change of state/cyclic acknowledgement message	
1	0	MAC ID						Group 2 message ID			Group 2 messages	400-5FF
1	0	Source MAC ID						0	0	0	Master's I/O bit-strobe command message	
1	0	Source MAC ID						0	0	1	Reserved	
1	0	Destination MAC ID						0	1	0	Master's change of state or cyclic acknowledgement message	
1	0	Source MAC ID						0	1	1	Slave's explicit response message	
1	0	Destination MAC ID						1	0	0	Master's explicit request message	
1	0	Destination MAC ID						1	0	1	Master's I/O poll command/change of state/cyclic message	
1	0	Destination MAC ID						1	1	0	Group 2 only unconnected explicit request message	
1	0	Destination MAC ID						1	1	1	Duplicate MAC ID check message	

The following types of messages are included in table 36:

- **I/O bit-strobe command/response message:** the bit-strobe command is an I/O message that is transmitted by the master. Multiple slaves may receive and react to the same bit-strobe command. The bit-strobe response is an I/O message that a slave transmits back to the master when the bit-strobe command is received;
- **I/O poll command/response message:** the poll command is an I/O message that is transmitted by the master. A poll command is directed to a specific slave. The poll response is an I/O message that a slave transmits back to the master when the poll command is received;

- **message d'E/S changement d'état/cyclique:** il s'agit de messages transmis par le maître ou l'esclave. Un message changement d'état/cyclique est dirigé vers une station spécifique. Un message d'acquiescement doit être renvoyé en réponse à ce message sauf si la configuration prévoit la suppression des messages d'acquiescement;
- **message de demande/réponse explicite:** voir 1.2.1.6;
- **message de demande explicite non connecté du groupe 2 seulement:** l'accès de demande explicite non connectée du groupe 2 seulement est utilisé pour attribuer/libérer le jeu de connexions maître/esclave prédéfini;
- **message de vérification de duplication d'adresses MAC ID:** voir 1.2.4.

1.5.3 Caractéristiques de l'objet connexion esclave

1.5.3.1 Généralités

Le présent paragraphe présente les caractéristiques, visibles en externe, des objets connexion associés au jeu de connexions maître/esclave prédéfini dans les appareils esclaves. Les objets des connexions maître/esclave prédéfinies décrits pour les appareils esclaves comprennent:

- **la connexion requête multiple:** responsable de la réception de la commande requête multiple du maître et de l'envoi de la réponse requête multiple associée;
- **la connexion interrogation:** responsable de la réception de la commande interrogation du maître et de l'envoi de la réponse interrogation associée;
- **la connexion de messagerie explicite:** responsable de la réception des demandes explicites et de l'envoi des réponses associées;
- **la connexion changement d'état/cyclique:** responsable de l'envoi du message changement d'état/cyclique et de la réception de la réponse d'acquiescement si cette fonction n'a pas été supprimée.

Le présent paragraphe donne des informations supplémentaires sur les objets connexions au sein d'un jeu de connexions maître/esclave prédéfini. Sauf indication contraire, toutes les informations spécifiées en 1.3 s'appliquent aux objets connexions décrits dans le présent paragraphe.

1.5.3.2 ID d'instance de connexion

Chaque objet connexion existant possède un ID d'instance de connexion qui identifie l'objet dans la classe connexion. Les ID d'instance de connexion devant être utilisés par un appareil esclave afin d'identifier les objets des connexions maître/esclave prédéfinies sont indiquées au tableau 37.

Tableau 37 – ID d'instance de connexion des connexions maître/esclave prédéfinies

ID d'instance de connexion	Description
1	Désigne la connexion de messagerie explicite dans le serveur
2	Désigne la connexion d'E/S sur interrogation
3	Désigne la connexion d'E/S sur requête multiple
4	Désigne la connexion d'E/S sur changement d'état ou cyclique de l'esclave

1.5.3.3 Comportement d'une instance de connexion maître/esclave prédéfinie

La figure 34 représente le diagramme de la machine d'état des objets des connexions maître/esclave prédéfinies.

- **I/O change of state/cyclic message:** the change of state/cyclic message is transmitted by either the master or the slave. A change of state/cyclic message is directed to a specific node. An acknowledgement message shall be returned in response to this message unless the configuration includes suppression of acknowledgement messages;
- **explicit response/request message:** see 5.2.1.6;
- **group 2 only unconnected explicit request message:** the group 2 only unconnected explicit request port is used to allocate/release the predefined master/slave connection set;
- **duplicate MAC ID check message:** see 5.2.4.

5.5.3 Slave connection object characteristics

5.5.3.1 General

This subclause presents the externally visible characteristics of the connection objects associated with the predefined master/slave connection set within slave devices. The predefined master/slave connection objects described for slave devices are:

- **the bit-strobe connection:** responsible for receiving the master's bit-strobe command and returning the associated bit-strobe response;
- **the poll connection:** responsible for receiving the master's poll command and returning the associated poll response;
- **the explicit messaging connection:** responsible for the reception of explicit requests and returning associated responses;
- **the change of state/cyclic connection:** responsible for sending the change of state/cyclic message and receiving the acknowledgement response if not suppressed.

This subclause gives additional information for connection objects within the predefined master/slave connection set. Except where noted, all information specified in 5.3 applies to the connection objects described in this subclause.

5.5.3.2 Connection instance IDs

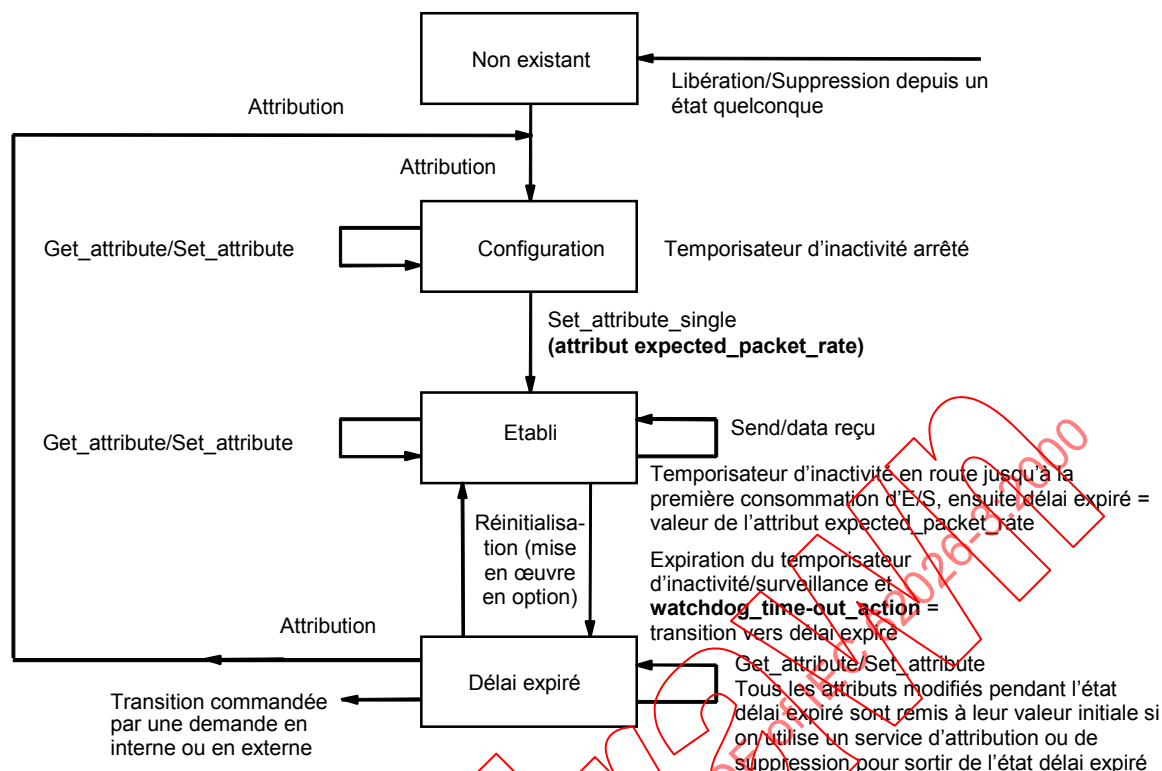
Every existing connection object has an assigned connection instance ID which identifies the connection object within the connection class. The connection instance IDs that shall be used by a slave device to identify predefined master/slave connection objects are shown in table 37.

Table 37 – Connection instance IDs for predefined master/slave connections

Connection instance ID	Description
1	Designates the explicit messaging connection into the server
2	Designates the poll I/O connection
3	Designates the bit-strobe I/O connection
4	Designates the slave's change of state or cyclic I/O connection

5.5.3.3 Predefined master/slave connection instance behaviour

Figure 34 illustrates the predefined master/slave I/O connection object state transition diagram.



IEC 1237/2000

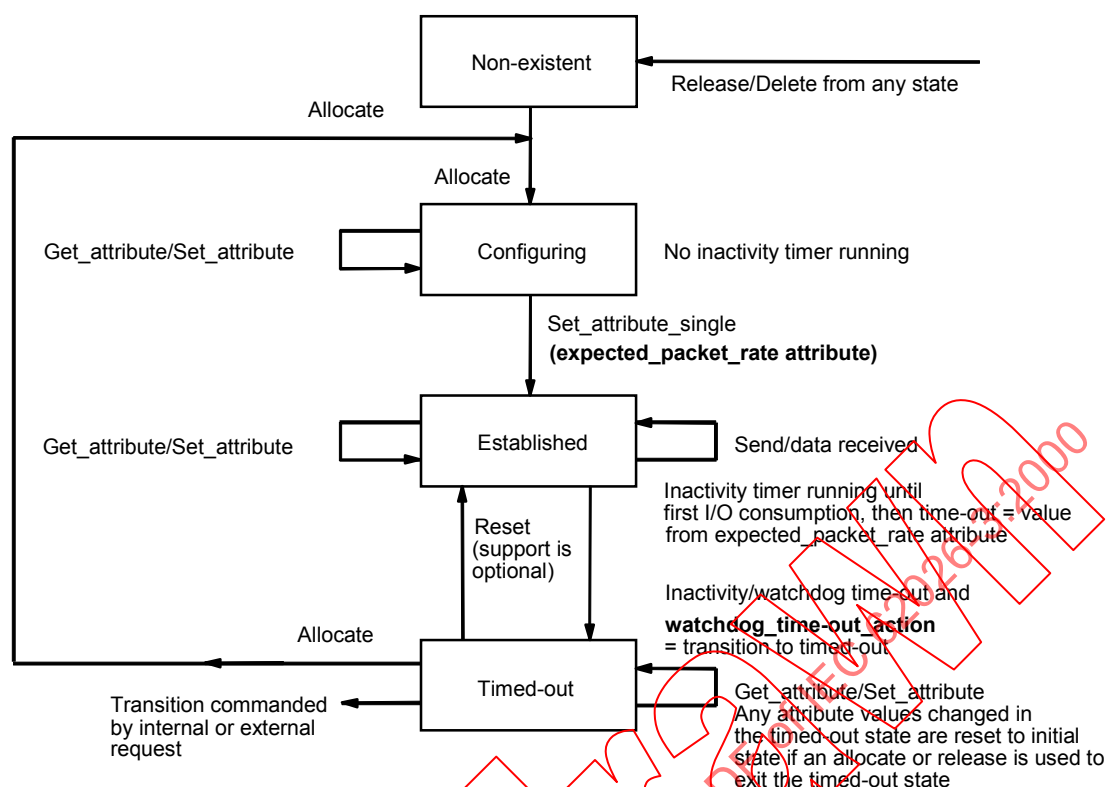
Figure 34 – Diagramme de la machine d'état des connexions d'E/S maître/esclave prédéfinies

Pour la modification des attributs, les connexions maître/esclave prédéfinies doivent au moins accepter la modification de l'attribut `expected_packet_rate`.

La table états-événements présentée ci-après donne une définition formelle du comportement des connexions d'E/S dans un jeu de connexions maître/esclave prédéfini. Cette table états-événements conserve et/ou remplace les actions décrites dans la table états-événements des objets connexion d'E/S en 1.3.

Tableau 38 – Table états-événements des connexions d'E/S maître/esclave prédéfinies

Etat de l'objet connexion d'E/S				
Événement	Non existant	Configuration	Etabli	Délai expiré
L'objet DeviceNet reçoit une demande d'attribution du jeu de connexion maître/esclave qui satisfait à toutes les vérifications d'erreur indiquées en 1.3.3.5.2.2. Cette demande spécifie l'une des connexions d'E/S maître/esclave prédéfinies	Créer un objet connexion pour chaque connexion d'E/S demandée et fixer les attributs aux valeurs par défaut spécifiées en 5.3.3.5.2.1. Transition vers configuration	Ceci est une erreur. Voir 1.3.3.5.2.2	Ceci est une erreur. Voir 1.3.3.5.2.2	Fixer les attributs aux valeurs par défaut spécifiées en 5.3.3.5.2.1. Transition vers configuration
La classe connexion reçoit une demande de suppression ou l'UCMM reçoit une demande de fermeture et la demande spécifie un objet connexion d'E/S maître/esclave prédéfini	Comme décrit dans le tableau 20	Libérer toutes les ressources associées. Transition vers non existant ¹⁾	Libérer toutes les ressources associées. Transition vers non existant ¹⁾	Libérer toutes les ressources associées. Transition vers non existant ¹⁾



The allocate and release services reset the connection instance. All connection object attributes are reset to their default values.

IEC 1237/2000

Figure 34 – Predefined master/slave I/O connection state transition diagram

For attribute modification, predefined master/slave I/O connections shall support at least the modification of the expected_packet_rate attribute.

The state-event matrix presented below provides a formal definition of the behaviour of I/O connections within the predefined master/slave connection set. This state-event matrix inherits from and/or overrides the actions presented in the I/O connection object state-event matrix in 5.3.

Table 38 – Predefined master/slave I/O connection state-event matrix

I/O connection object state				
Event	Non-existent	Configuring	Established	Timed-out
DeviceNet object receives an allocate master/slave connection set request that passes all error checks specified in 5.3.3.5.2.2. This request specifies one of the predefined master/slave I/O connections	Create a connection object for each requested I/O connection and set attributes to default values specified in 5.3.3.5.2.1. Transition to configuring	This is an error. See 5.3.3.5.2.2	This is an error. See 5.3.3.5.2.2	Set attributes to default values specified in 5.3.3.5.2.1. Transition to configuring
Connection class receives a delete request or the UCMM receives a close request and the request specifies a predefined master/slave I/O connection object	As described in table 20	Release all associated resources. Transition to non-existent ¹⁾	Release all associated resources. Transition to non-existent ¹⁾	Release all associated resources. Transition to non-existent ¹⁾

Tableau 38 (suite)

Etat de l'objet connexion d'E/S				
Événement	Non existant	Configuration	Etabli	Délai expiré
L'objet DeviceNet reçoit une demande de libération du jeu de connexions maître/esclave qui satisfait à toutes les vérifications d'erreur indiquées en 1.3.3.5.2.2. Cette demande spécifie l'une des connexions d'E/S maître/esclave prédéfinies	Ceci est une erreur Voir 1.3.3.5.2.2	Libérer toutes les ressources associées Transition vers non existant ¹⁾	Libérer toutes les ressources associées Transition vers non existant ¹⁾	Libérer toutes les ressources associées. Transition vers non existant ¹⁾
Set_attribute_single	Comme décrit dans le tableau 20	Valider/exécuter la demande en fonction des règles d'accès présentées en 1.5.3.4. S'il s'agit d'une demande valide d'écriture de l'attribut expected_packet_rate, accomplir les étapes spécifiées dans le tableau 20 pour l'événement apply_attributes dans l'état configuration et passer à l'état établi. Renvoyer la réponse appropriée	Valider/exécuter la demande en fonction des règles d'accès présentées en 1.5.3.4. Renvoyer la réponse appropriée	Valider/exécuter la demande en fonction des règles d'accès présentées en 1.5.3.4. Renvoyer la réponse appropriée
Get_attribute_single	Comme décrit dans le tableau 20	Valider/exécuter la demande en fonction des règles d'accès présentées en 1.5.3.4. Renvoyer la réponse appropriée	Valider/exécuter la demande en fonction des règles d'accès présentées en 1.5.3.4. Renvoyer la réponse appropriée	Valider/exécuter la demande en fonction des règles d'accès présentées en 1.5.3.4. Renvoyer la réponse appropriée
Reset	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20
Apply_attributes ²⁾	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20
Receive_data	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20
Send_message	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20
Expiration du temporisateur d'inactivité/surveillance	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20	Comme décrit dans le tableau 20 ¹⁾	Comme décrit dans le tableau 20
¹⁾ Chaque fois qu'un objet connexion d'un jeu de connexions maître/esclave prédéfini quitte l'état établi, il peut être nécessaire de libérer automatiquement le jeu complet de connexions maître/esclave prédéfini. Lorsque le jeu de connexions maître/esclave prédéfini est libéré, tous les objets connexions associés repassent à l'état non existant. Voir 5.3.3.5.2.3 pour plus de détails concernant la libération automatique du jeu de connexions maître/esclave prédéfini. ²⁾ Du fait qu'un service apply_attributes implicite accompagne un set_attribute_single pour l'attribut expected_packet_rate d'un objet connexion d'E/S maître/esclave prédéfini, dans l'état configuration, une demande de messagerie explicite apply_attributes réussira uniquement si expected_packet_rate attend encore d'être modifié avec succès via la demande set_attribute_single et, par conséquent, contient toujours la valeur par défaut 0.				

Le diagramme de la machine d'état (voir figure 35) montre le comportement d'une connexion de messagerie explicite maître/esclave prédéfinie.

Table 38 (continued)

I/O connection object state				
Event	Non-existent	Configuring	Established	Timed-out
DeviceNet object receives a release master/slave connection set request that passes all error checks specified in 5.3.3.5.2.2 This request specifies one of the predefined master/slave I/O connections	This is an error See 5.3.3.5.2.2	Release all associated resources Transition to non-existent ¹⁾	Release all associated resources Transition to non-existent ¹⁾	Release all associated resource Transition to non-existent ¹⁾
Set_attribute_single	As described in table 20	Validate/service the request according to the access rules presented in 5.5.3.4 If this is a valid request to set the expected_packet_rate attribute, then perform the steps specified in table 20 under the apply_attributes event in the configuring state and transition to established. Return appropriate response	Validate/service the request according to the access rules presented in 5.5.3.4 Return appropriate response	Validate/service the request according to the access rules presented in 5.5.3.4 Return appropriate response
Get_attribute_single	As described in table 20	Validate/service the request according to the access rules presented in 5.5.3.4 Return appropriate response	Validate/service the request according to the access rules presented in 5.5.3.4 Return appropriate response	Validate/service the request according to the access rules presented in 5.5.3.4 Return appropriate response
Reset	As described in table 20	As described in table 20	As described in table 20	As described in table 20
Apply_attributes ²⁾	As described in table 20	As described in table 20	As described in table 20	As described in table 20
Receive_data	As described in table 20	As described in table 20	As described in table 20	As described in table 20
Send_message	As described in table 20	As described in table 20	As described in table 20	As described in table 20
Inactivity/watchdog timer expires	As described in table 20	As described in table 20	As described in table 20 ¹⁾	As described in table 20
¹⁾ Whenever a connection object within the predefined master/slave connection set transitions out of the established state, the entire predefined master/slave connection set may need to be automatically released. When the predefined master/slave connection set is released, all associated connection objects return to the non-existent state. See 5.3.3.5.2.3 for more details concerning the automatic release of the predefined master/slave connection set. ²⁾ Since an implied apply_attributes accompanies a set_attribute_single of the expected_packet_rate attribute of a predefined master/slave I/O connection object in the configuring state, the only time an apply_attributes explicit messaging request will succeed is when expected_packet_rate has yet to be successfully modified via the set_attribute_single request and, thus, still contains the default value of 0.				

The state transition diagram (see figure 35) shows the behaviour of the predefined master/slave explicit messaging connection.

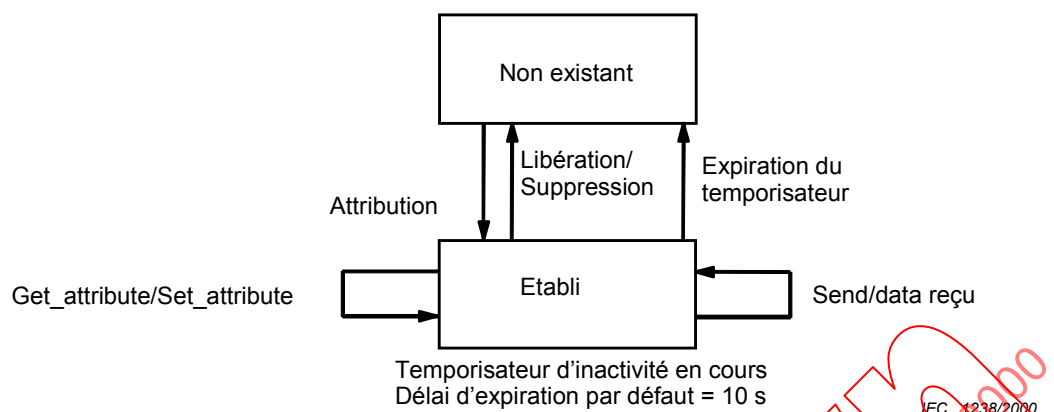


Figure 35 – Diagramme de la machine d'état pour la messagerie explicite maître/esclave prédéfinie

Le tableau 39 détaille la table états-événements pour l'objet connexion de messagerie explicite maître/esclave prédéfini. Cette table états-événements conserve et/ou remplace les actions présentées dans la table états-événements des objets connexions d'E/S de 1.3.

Tableau 39 – Table états-événements des connexions d'E/S maître/esclave prédéfinies

Etat d'un objet connexion de messagerie explicite		
Événement	Non existant	Etabli
L'objet DeviceNet reçoit une demande d'attribution du jeu de connexion maître/esclave qui satisfait à toutes les vérifications d'erreur indiquées en 1.3.3.5.2.2. Cette demande spécifie la connexion de messagerie explicite maître/esclave prédéfinie	Créer l'objet connexion de messagerie explicite maître/esclave prédéfinie pour chaque connexion d'E/S demandée et fixer les attributs aux valeurs par défaut (voir 5.3.3.5.2.1). Passer à établi.	Ceci est une erreur. Voir 1.3.3.5.2.2
L'UCMM reçoit une demande de fermeture ou la classe connexion reçoit une demande de suppression, et la demande spécifie l'objet connexion de messagerie explicite maître/esclave prédéfini	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
L'objet DeviceNet reçoit une demande de libération du jeu de connexions maître/esclave qui satisfait à toutes les vérifications d'erreur indiquées en 1.3.3.5.2.2. Cette demande spécifie la connexion de messagerie explicite maître/esclave prédéfinie	Ceci est une erreur. Voir 1.3.3.5.2.2	Libérer toutes les ressources associées Transition vers non existant ¹⁾
Set_attribute_single	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
Get_attribute_single	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
Reset	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
Apply_attributes	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
Receive_data	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
Send_message	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
Expiration du temporisateur d'inactivité/surveillance	Comme décrit dans le tableau 21	Comme décrit dans le tableau 21
¹⁾ Chaque fois qu'un objet connexion d'un jeu de connexions maître/esclave prédéfini quitte l'état établi, il peut être nécessaire de libérer automatiquement le jeu complet de connexions. Lorsque le jeu de connexions maître/esclave prédéfini est libéré, tous les objets connexions associés repassent à l'état non existant. Voir 5.3.3.5.2.3 pour plus de détails concernant la libération automatique du jeu de connexions maître/esclave prédéfini.		

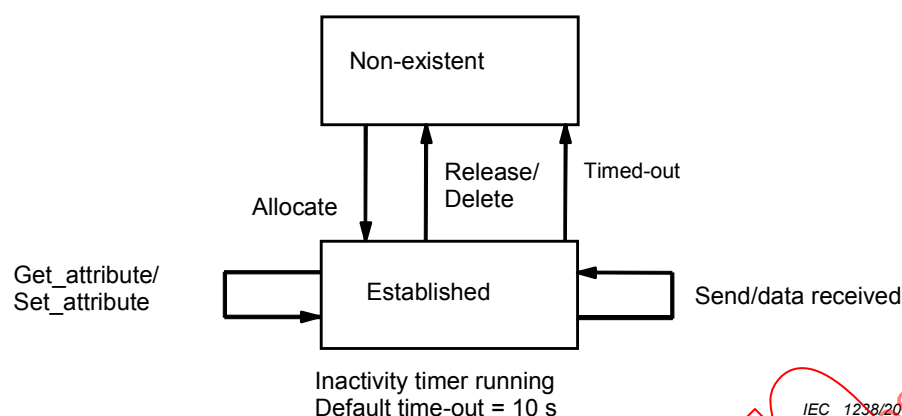


Figure 35 – Predefined master/slave explicit messaging state transition diagram

Table 39 provides a detailed state-event matrix for the predefined master/slave explicit messaging connection object. This state-event matrix inherits from and/or overrides the actions presented in the I/O connection object state-event matrix in 5.3.

Table 39 – Predefined master/slave I/O connection state-event matrix

Explicit messaging connection object state		
Event	Non-existent	Established
DeviceNet object receives an allocate master/slave connection set request that passes all error checks specified in 5.3.3.5.2.2 This request specifies the predefined master/slave explicit messaging connection	Create the predefined master/slave explicit messaging connection object for each requested I/O connection and set attributes to default values (see 5.3.3.5.2.1) Transition to established	This is an error See 5.3.3.5.2.2
UCMM receives a close request or the connection class receives a delete request, and the request specifies the predefined master/slave explicit messaging connection object	As described in table 21	As described in table 21
DeviceNet object receives a release master/slave connection set request that passes all error checks specified in 5.3.3.5.2.2 This request specifies the predefined master/slave explicit messaging connection	This is an error See 5.3.3.5.2.2	Release all associated resources Transition to non-existent ¹⁾
Set_attribute_single	As described in table 21	As described in table 21
Get_attribute_single	As described in table 21	As described in table 21
Reset	As described in table 21	As described in table 21
Apply_attributes	As described in table 21	As described in table 21
Receive_data	As described in table 21	As described in table 21
Send_message	As described in table 21	As described in table 21
Inactivity/watchdog timer expires	As described in table 21	As described in table 21
¹⁾ Whenever a connection object within the predefined master/slave connection set transitions out of the established state, the entire predefined master/slave connection set may need to be automatically released. When the predefined master/slave connection set is released, all associated connection objects return to the non-existent state. See 5.3.3.5.2.3 for more details concerning the automatic release of the predefined master/slave connection set.		

1.5.3.4 Règles d'accès aux attributs d'instance de connexion

Le paragraphe 1.3.2.8 donne une description générale des règles d'accès associées aux objets connexions. Le tableau 40 indique les règles d'accès spécifiques aux objets connexions d'E/S maître/esclave prédéfinis et conserve et/ou remplace les règles présentées en 1.3.2.8.

Tableau 40 – Accès aux attributs des objets connexions d'E/S maître/esclave prédéfinis

Etat de la connexion d'E/S				
Attribut	Non existant	Configuration	Etabli	Délai expiré
State	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Instance_type	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Transportclass_trigger	Comme décrit dans le tableau 22	Get/set ¹⁾	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Produced_connection_id	Comme décrit dans le tableau 22	Get only	Get only	Get only
Consumed_connection_id	Comme décrit dans le tableau 22	Get only	Get only	Get only
Initial_comm_characteristics	Comme décrit dans le tableau 22	Get only	Get only	Get only
Produced_connection_size	Comme décrit dans le tableau 22	Get/set ²⁾	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Consumed_connection_size	Comme décrit dans le tableau 22	Get/set	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Expected_packet_rate	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Watchdog_timeout_action	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Produced_connection_path_length	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Produced_connection_path	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Consumed_connection_path_length	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
Consumed_connection_path	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22	Comme décrit dans le tableau 22
¹⁾ L'attribut transportclass_trigger de l'objet connexion d'E/S maître/esclave prédéfini d'un appareil esclave ne doit être modifiable qu'avec l'une des valeurs suivantes: 82 _{hex} – serveur/classe de transport 2, 83 _{hex} – serveur/classe de transport 3. ²⁾ L'attribut produced_connection_size de la connexion d'E/S requête multiple ne doit pas pouvoir être fixé à une valeur supérieure à 8.				

1.5.4 Caractéristiques de l'objet connexion maître

Cette partie ne décrit pas de caractéristiques pour les objets connexions d'un maître. Le maître doit adopter un comportement externe qui lui permette d'interagir avec ses esclaves.

1.5.5 Messages de commande/réponse requête multiple

1.5.5.1 Généralités

Les messages de commande et réponse requête multiple transfèrent des trames de données d'E/S courtes entre un maître et ses esclaves en mode requête multiple.

5.5.3.4 Connection instance attribute access rules

Subclause 5.3.2.8 gives a general description of the access rules associated with connection objects. Table 40 shows access rules specific to the predefined master/slave I/O connection objects and inherits from and/or overrides the rules presented in 5.3.2.8.

Table 40 – Predefined master/slave I/O connection object attribute access

Attribute	I/O connection state			
	Non-existent	Configuring	Established	Timed-out
State	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Instance_type	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Transportclass_trigger	As described in table 22	Get/set ¹⁾	As described in table 22	As described in table 22
Produced_connection_id	As described in table 22	Get only	Get only	Get only
Consumed_connection_id	As described in table 22	Get only	Get only	Get only
Initial_comm_characteristics	As described in table 22	Get only	Get only	Get only
Produced_connection_size	As described in table 22	Get/set ²⁾	As described in table 22	As described in table 22
Consumed_connection_size	As described in table 22	Get/set	As described in table 22	As described in table 22
Expected_packet_rate	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Watchdog_timeout_action	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Produced_connection_path_length	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Produced_connection_path	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Consumed_connection_path_length	As described in table 22	As described in table 22	As described in table 22	As described in table 22
Consumed_connection_path	As described in table 22	As described in table 22	As described in table 22	As described in table 22
¹⁾ The transportclass_trigger attribute of a predefined master/slave I/O connection object within a slave device shall only be modifiable to one the following values: 82 _{hex} – server/transport class 2, 83 _{hex} – server/transport class 3. ²⁾ The produced_connection_size attribute of the bit-strobe I/O connection shall not be settable to a value greater than 8.				

5.5.4 Master connection object characteristics

This part does not present characteristics with respect to connection objects within a master. The master shall exhibit the external behaviour necessary to interface with its slaves.

5.5.5 Bit-strobe command/response messages

5.5.5.1 General

Bit-strobe command and response messages move small packets of I/O data between a master and its bit-strobed slaves.

1.5.5.2 Message de commande requête multiple

La commande requête multiple envoie un bit de données de sortie à chaque esclave.

Le message de commande requête multiple contient une chaîne de 64 bits (8 octets) de données de sortie, un bit de sortie pour chaque MAC ID possible sur la liaison.

Un appareil esclave peut être conçu pour accomplir l'une des tâches ou toutes les tâches suivantes:

- ignorer la commande requête multiple;
- consommer la commande requête multiple ainsi que ses données de sortie;
- consommer la commande requête multiple en tant que déclencheur et ignorer les données de sortie.

Par défaut, les esclaves doivent ignorer la commande requête multiple tant que la connexion requête multiple n'a pas été attribuée.

Un message de commande requête multiple transmis sans données dans le champ de données CAN est interprété comme étant un événement `receive_idle` par un objet applicatif. Un message de commande requête multiple qui contient des données est interprété comme étant un événement d'exécution par un objet applicatif. Le comportement d'un objet applicatif sur détection de `receive_idle` ou d'un événement d'exécution est spécifique à l'objet applicatif. Voir la description des objets applicatifs pour obtenir plus d'informations sur ces événements.

1.5.5.3 Message réponse requête multiple

La réponse requête multiple renvoie jusqu'à huit octets de données d'entrée au maître en provenance de chaque esclave.

Un message réponse requête multiple qui ne contient aucune donnée alors qu'il est configuré pour en contenir indique comme événement «pas de données requête multiple valides pour le maître». Le comportement du maître, sur détection de cet événement, est spécifique à une réalisation.

1.5.5.4 Caractéristiques d'un message requête multiple

Le maître met en oeuvre les ressources de communication associées à la commande requête multiple en définissant une connexion client de classe de transport 0 pour transmettre la commande et un ensemble d'objets connexions serveur de classe de transport 0 pour recevoir les réponses.

L'esclave met en oeuvre une seule connexion serveur de classe de transport 2 ou 3 pour recevoir la commande requête multiple et renvoyer la réponse associée.

1.5.6 Messages de commande/réponse interrogation

1.5.6.1 Généralités

Les messages de commande et réponse interrogation transfèrent des données d'E/S entre un maître et ses esclaves interrogés.

1.5.6.2 Message de commande interrogation

La commande interrogation envoie jusqu'à 65535 octets de données de sortie (non fragmentés ou fragmentés) à l'appareil esclave de destination.

5.5.5.2 Bit-strobe command message

The bit-strobe command sends one bit of output data to each slave.

The bit-strobe command message contains a bit string of 64 bits (8 bytes) of output data, one output bit for each possible MAC ID on the link.

A slave device may be designed to do one or all of the following:

- ignore the bit-strobe command;
- consume the bit-strobe command and its output data;
- consume the bit-strobe command as a trigger and ignore the output data.

Slaves shall default to ignoring the bit-strobe command until the bit-strobe connection is allocated.

A bit-strobe command message transmitted with no data in the CAN data field is interpreted as a receive_idle event by an application object. A bit-strobe command message that contains data is interpreted as a run event by an application object. The behaviour of an application object upon detection of either the receive_idle or run event is application object-specific. See the application object descriptions for more information concerning these events.

5.5.5.3 Bit-strobe response message

The bit-strobe response returns up to eight bytes of input data to the master from each slave.

A bit-strobe response message that contains no data and is configured to contain data indicates a master device no valid bit-strobe data event. The behaviour of the master upon detection of this event is implementation-specific.

5.5.5.4 Bit-strobe message characteristics

The master implements the communication resources associated with the bit-strobe command as a client transport class 0 connection to transmit the command and a set of server transport class 0 connection objects to receive the responses.

The slave implements a single server transport class 2 or 3 connection to receive the bit-strobe command and send the associated response.

5.5.6 Poll command/response messages

5.5.6.1 General

The poll command and response messages transfer I/O data between a master and its polled slaves.

5.5.6.2 Poll command message

The poll command sends up to 65535 bytes of output data (non-fragmented or fragmented) to the destination slave device.

Un appareil esclave a la possibilité d'accomplir l'une ou l'ensemble des tâches suivantes:

- ignorer la commande interrogation si aucune connexion interrogation n'est attribuée;
- consommer la commande interrogation ainsi que ses données de sortie;
- consommer la commande interrogation en tant que déclencheur et ignorer les données de sortie.

Par défaut, les esclaves doivent ignorer la commande interrogation tant que la connexion interrogation n'a pas été attribuée.

Un message de commande interrogation transmis sans données dans le champ de données CAN est interprété comme étant un événement `receive_idle` par un objet applicatif. Un message de commande interrogation qui contient des données est interprété comme étant un événement d'exécution par un objet applicatif. Le comportement d'un objet applicatif sur détection de `receive_idle` ou d'un événement d'exécution est spécifique à l'objet applicatif.

1.5.6.3 Message réponse interrogation

La réponse interrogation renvoie jusqu'à 65535 octets (non fragmentés ou fragmentés) de données d'entrée au maître en provenance de l'esclave.

Un message réponse interrogation qui ne contient aucune donnée alors qu'il est configuré pour en contenir indique comme événement «pas de données interrogation valides pour l'appareil maître». Le comportement du maître, sur détection de cet événement, est spécifique à une réalisation.

1.5.7 Connexions cyclique/changement d'état

1.5.7.1 Généralités

Le jeu de connexions maître/esclave prédéfini permet le transfert de données sur changement d'état ou cyclique entre le maître et l'esclave. Ce transfert de données peut être acquitté ou non acquitté.

Les jeux de connexions cyclique/changement d'état utilisent l'instance de connexion 2 pour le transfert du maître à l'esclave et l'acquittement de l'esclave envers le maître. L'instance de connexion 4 est utilisée pour le transfert de données de l'esclave au maître et l'acquittement du maître envers l'esclave. Si un appareil n'accepte pas le mode interrogation et ne prévoit aucune donnée de sortie, l'instance de connexion 2 n'a pas besoin d'être créée.

Pour garantir un fonctionnement cohérent du système, l'attribut `consumed_connection_path` de la connexion changement d'état/cyclique de l'esclave (instance 4) doit spécifier l'objet gestionnaire d'acquittement; l'instance de l'objet gestionnaire d'acquittement doit être 1.

Du fait que les jeux de connexion interrogation et changement d'état/cyclique partagent l'instance de connexion 2, l'esclave doit suivre certaines procédures basées sur la demande d'attribution afin d'assurer un comportement correct. Comme décrit ci-dessus, les instances de connexion 2 et 4 ne sont créées que lorsque le bit d'attribution changement d'état/cyclique est mis à 1. L'instance de connexion 4 produit des données d'entrée à partir du chemin par défaut et consomme les acquittements dirigés vers l'instance 1 de l'objet gestionnaire d'acquittement. L'instance de connexion 2 consomme des données de sortie dirigées vers le chemin par défaut et produit un acquittement de longueur nulle. Lorsqu'une demande d'attribution ne contient que le bit d'attribution interrogation, l'instance de connexion 2 consomme des données de sorties dirigées vers le chemin de sortie par défaut et produit des données d'entrée à partir du chemin d'entrée par défaut.

A slave device is permitted to do one or all of the following:

- ignore the poll command if no poll connection is allocated;
- consume the poll command and its output data;
- consume the poll command as a trigger and ignore the output data.

Slaves shall default to ignoring the poll command until the poll connection is allocated.

A poll command message transmitted with no data in the CAN data field is interpreted as a receive_idle event by an application object. A poll command message that contains data is interpreted as a run event by an application object. The behaviour of an application object upon detection of the receive_idle or run event is application object-specific.

5.5.6.3 Poll response message

The poll response returns up to 65535 bytes (non-fragmented or fragmented) of input data to the master from the slave.

A poll response message that contains no data and is configured to contain data indicates a master device no valid poll data event. The master behaviour upon detection of this event is implementation-specific.

5.5.7 Change of state/cyclic connections

5.5.7.1 General

The predefined master/slave connection set supports change of state or cyclic data transfer between the master and the slave. This data transfer may be either acknowledged or unacknowledged.

The change of state/cyclic connection sets use connection instance 2 for master to slave data transfer and slave to master acknowledgement. Connection instance 4 is used for slave to master data transfer and master to slave acknowledgement. If a device does not support the poll and has no support for output data, connection instance 2 need not be created.

To keep the system behaviour consistent, the slave's change of state/cyclic connection (instance 4) shall have the consumed_connection_path attribute set to the acknowledgement handler object, and the instance of the acknowledgement handler object shall be 1.

Because the polled and change of state/cyclic connection sets share connection instance 2, the slave shall follow certain procedures based on the allocation request in order to ensure the correct behaviour. As described above, it is only when the change of state/cyclic allocation bit is on that connection instances 2 and 4 are created. Connection instance 4 produces from the default input path and consumes acknowledgements to instance 1 of the acknowledgement handler object. Connection instance 2 consumes to the default output path and produces a zero length acknowledgement. When the allocation request contains only the polled allocation bit, connection instance 2 consumes to the default output path and produces from the default input path.

Si les deux bits d'attribution changement d'état/cyclique et interrogation sont mis à 1, l'instance de connexion 2 doit se comporter comme si seul le bit d'attribution «interrogation» était à 1, et l'instance de connexion 4 doit continuer à se comporter comme décrit ci-dessus. C'est aussi le comportement requis si le maître attribue le jeu de connexions «interrogation» dans un seul message, puis le jeu de connexions changement d'état/cyclique dans un message ultérieur.

Pour accomplir une production de données non acquittées, le bit de suppression d'acquiescement est mis à 1 avec le bit changement d'état ou le bit cyclique de l'octet choix d'attribution. L'instance de connexion 2 est configurée comme connexion de classe de transport 0 pour la production de données du maître vers l'esclave. L'instance de connexion 4 est configurée comme connexion de classe de transport 0 pour la production de données de l'esclave vers le maître. Lorsque le bit interrogation est mis à 1 dans le même choix d'attribution, cela entraîne la configuration de l'instance de connexion 2 pour le mode interrogation, alors que l'instance de connexion 4 continue à être configurée comme une connexion de changement d'état ou cyclique non acquittée.

1.5.7.2 Acquittement du déclencheur de messages changement d'état/cyclique

1.5.7.2.1 Généralités

Les messages changement d'état/cyclique transfèrent jusqu'à 65535 octets de données entre un maître et un esclave en utilisant des déclencheurs de production sur changement d'état ou bien cycliques. La production des données peut être acquittée ou non acquittée.

Les paragraphes suivants décrivent:

- le message changement d'état/cyclique (acquitté);
- le message changement d'état/cyclique (non acquitté);
- les caractéristiques du message changement d'état/cyclique;
- un exemple d'application changement d'état/cyclique.

1.5.7.2.2 Message changement d'état/cyclique du maître

Le message changement d'état/ cyclique du maître envoie jusqu'à 65535 octets de données (non fragmentée ou fragmentés) à l'appareil esclave de destination. La production des données est déclenchée par un changement d'état ou par l'expiration du délai du déclencheur de transmission.

Un message changement d'état/cyclique transmis sans données dans le champ de données CAN est interprété comme étant un événement receive_idle par un objet applicatif. Un message changement d'état/cyclique qui contient des données est interprété comme étant un événement d'exécution par un objet applicatif. Le comportement d'un objet applicatif sur détection de l'événement d'exécution est spécifique à l'objet applicatif.

1.5.7.2.3 Message d'acquiescement changement d'état/cyclique de l'esclave

Le message d'acquiescement changement d'état de l'esclave renvoie jusqu'à 65535 octets de données (non fragmentés ou fragmentés) au maître en provenance de l'esclave. Par défaut, le message d'acquiescement a une longueur nulle.

1.5.7.2.4 Message changement d'état/ cyclique de l'esclave

Le message changement d'état/cyclique de l'esclave envoie jusqu'à 65535 octets de données (non fragmentés ou fragmentés) au maître en provenance de l'esclave. La production de données est déclenchée par un changement d'état ou par l'expiration du délai du déclencheur de transmission.

If both the change of state/cyclic *and* the polled allocation bits are set, connection instance 2 shall behave as though only the polled allocation bit was set, and connection instance 4 shall continue to behave as described above. This is also the required behaviour if the master allocates the poll connection set in one message, and then the change of state/cyclic connection set in a following message.

To achieve unacknowledged data production, the acknowledgement suppression bit is set with either the change of state or cyclic bit in the allocation choice byte. Connection instance 2 is configured as a transport class 0 connection for master to slave data production. Connection instance 4 is configured as a transport class 0 connection for slave to master data production. When the poll bit is set in the same allocation choice, this causes connection instance 2 to be configured as the poll, while connection instance 4 continues to be configured as an unacknowledged change of state or cyclic connection.

5.5.7.2 Change of state/cyclic messages trigger acknowledgement

5.5.7.2.1 General

The change of state/cyclic messages move up to 65535 bytes of data between a master and slave using change of state or cyclic production triggers. Data production may be either acknowledged or unacknowledged.

The following subclauses describe:

- change of state/cyclic message (acknowledged);
- change of state/cyclic message (unacknowledged);
- change of state/cyclic message characteristics;
- change of state/cyclic example application.

5.5.7.2.2 Master's change of state/cyclic message

The master's change of state/cyclic sends up to 65535 bytes of data (non-fragmented or fragmented) to the destination slave device. Data production is triggered by either a change of state or transmission trigger time-out.

A change of state/cyclic message transmitted with no data in the CAN data field is interpreted as a receive_idle event by an application object. A change of state/cyclic message that contains data is interpreted as a run event by an application object. The behaviour of an application object upon detection of the run event is application object-specific.

5.5.7.2.3 Slave's change of state/cyclic acknowledgement message

The slave's change of state acknowledgement message returns up to 65535 bytes (non-fragmented or fragmented) of data to the master from the slave. By default, the acknowledgement message is zero length.

5.5.7.2.4 Slave's change of state/cyclic message

The slave's change of state/cyclic message sends up to 65535 bytes of data (non-fragmented or fragmented) to the master from the slave. Data production is triggered by either a change of state or transmitted trigger time-out.

Un message changement d'état/cyclique qui ne contient aucune donnée alors qu'il est configuré pour en contenir indique comme événement «pas de données changement d'état/cyclique valides pour l'appareil maître». Le comportement du maître, sur détection de cet événement, dépend de la réalisation.

1.5.7.2.5 Message d'acquittement changement d'état/cyclique du maître

Le message d'acquittement changement d'état/cyclique du maître renvoie jusqu'à 65535 octets de données (non fragmentés ou fragmentés) à l'esclave en provenance du maître. Par défaut, le message d'acquittement a une longueur nulle.

1.5.8 Appareils du groupe 2 seulement

Pour établir des communications avec un appareil du groupe 2 seulement, un client doit attribuer le jeu de connexions maître/esclave prédéfini. La demande d'attribution d'un appareil du groupe 2 seulement est transmise en tant que demande explicite non connectée du groupe 2 seulement. Au lieu d'utiliser l'UCMM pour établir les connexions de messagerie explicite, les appareils du groupe 2 seulement reçoivent et traitent les messages de demande explicite non connectés du groupe 2 seulement.

Les seuls services valides qui peuvent être transmis en tant que messages de demande explicite non connectés du groupe 2 seulement, sont les suivants:

- message allocate_master/slave_connection_set;
- message release_master/slave_connection_set.

Ces services sont décrits en 5.3.3.5.2.

Les réponses aux demandes explicites non connectées du groupe 2 seulement sont renvoyées par transmission d'un message du groupe 2 dont l'élément MAC ID est chargé avec le MAC ID de l'appareil qui répond (MAC ID source), sachant que l'ID de message est fixé à 3.

Les appareils du groupe 2 seulement ne doivent utiliser le champ identificateur de message réponse explicite de l'esclave que pour transmettre des messages réponses non connectés du groupe 2 seulement et des messages de réponse pour la connexion explicite maître/esclave prédéfinie.

Si un serveur du groupe 2 seulement reçoit un message de demande non connecté du groupe 2 seulement qui n'est pas une demande allocate_master/slave_connection_set ou release_master/slave_connection_set, une réponse d'erreur doit alors être renvoyée, avec le code d'erreur général fixé à 02 et le code d'erreur supplémentaire fixé à la valeur 03, spécifique à l'objet DeviceNet (voir 1.3.3.6).

1.6 Couche physique

1.6.1 Généralités

Le tableau 41 donne les spécifications générales de la couche physique qui doivent être respectées dans les systèmes DeviceNet (voir 5.2.6 pour les spécifications d'essais).

A change of state/cyclic message that contains no data and is configured to contain data indicates a master device no valid change of state/cyclic data event. The master behaviour upon detection of this event is implementation-specific.

5.5.7.2.5 Master's change of state/cyclic acknowledgement message

The master's change of state/cyclic acknowledgement message returns up to 65535 bytes (non-fragmented or fragmented) of data to the slave from the master. By default, the acknowledgement message is zero length.

5.5.8 Group 2 only devices

To establish communications with a group 2 only device, a client shall allocate the predefined master/slave connection set. The request to allocate a group 2 only device is transmitted as a group 2 only unconnected explicit request. Instead of using the UCM to establish explicit messaging connections, group 2 only devices receive and process group 2 only unconnected explicit request messages.

The only services that are valid when transmitted as group 2 only unconnected explicit request messages are:

- allocate_master/slave_connection_set message;
- release_master/slave_connection_set message.

These services are described in 5.3.3.5.2.

Responses to group 2 only unconnected explicit requests are returned by transmitting a group 2 message whose MAC ID component is set to the responding device's MAC ID (source MAC ID) and whose message ID is set to 3.

Group 2 only devices shall use the slave's explicit response message identifier field only for transmitting group 2 only unconnected response messages and predefined master/slave explicit messaging connection response messages.

If a group 2 only server receives a group 2 only unconnected request message that is not an allocate_master/slave_connection_set or release_master/slave_connection_set request, then an error response shall be returned whose general error code is set to 02, with the additional error code set to a DeviceNet object-specific value of 03 (see 5.3.3.6).

5.6 Physical layer

5.6.1 General

Table 41 shows the general physical layer specifications which shall be met in DeviceNet systems (see 9.2.6 for test specifications).

Tableau 41 – Caractéristiques générales de la couche physique

Caractéristiques	Spécifications
Vitesses de transmission	125 kbit/s, 250 kbit/s, 500 kbit/s
Longueur totale maximale de la ligne principale et longueur maximale de câble entre deux appareils quelconques	500 m à 125 kbit/s 250 m à 250 kbit/s 100 m à 500 kbit/s
Nombre de stations accepté par le transcepteur	64 minimum
Signalisation	Conformément à l'ISO 11898
Modulation	Bande de base
Codage	NRZ avec bits de remplissage
Couplage au support de transmission	Tx/Rx à couplage différentiel c.c.
Isolation (entre le circuit transcepteur et le composant CAN)	500 V c.c. Ne concerne que les appareils isolés
Impédance d'entrée différentielle typique (état récessif)	Shunt C = 5 pF Shunt R = 25 kΩ (sous tension)
Impédance d'entrée différentielle minimale (état récessif)	Shunt C = 24 pF Shunt R = 20 kΩ (sous tension)
Tension sur les conducteurs d'alimentation	11 V c.c. à 25 V c.c.
Plage maximale de tension des signaux	-25 V c.c. à +18 V c.c. (CAN_H, CAN_L) ¹⁾
¹⁾ Les tensions à CAN_H et à CAN_L doivent avoir comme référence la terre du transcepteur. Le potentiel à la terre du transcepteur peut être supérieur à celui de la borne U- d'une quantité égale à la chute de tension au travers d'une diode Schottky ou équivalent. Cette tension doit être de 0,6 V maximum relativement à U-.	

La couche physique, comprenant le transcepteur, le connecteur, le circuit de protection contre les erreurs de câblage, le régulateur de tension, le support de transmission et l'isolation (en option), doit être réalisée comme indiqué par la figure 36. Les essais permettant de vérifier la robustesse de la réalisation de la couche physique dans un appareil sont spécifiés en 5.2.9.

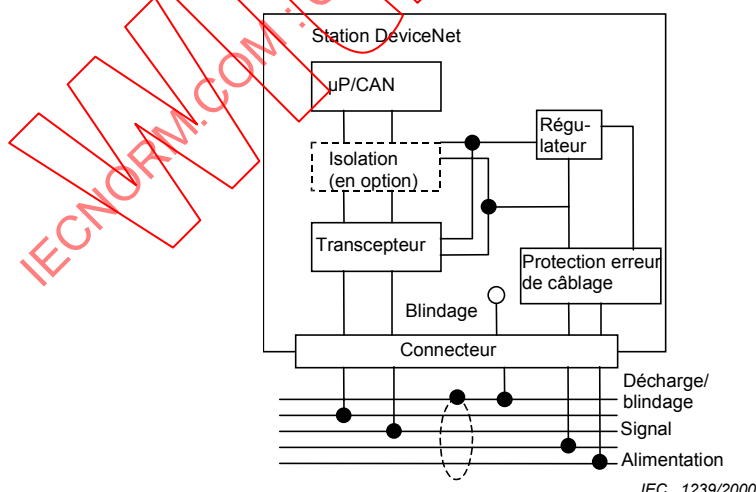


Figure 36 – Schéma fonctionnel de la couche physique

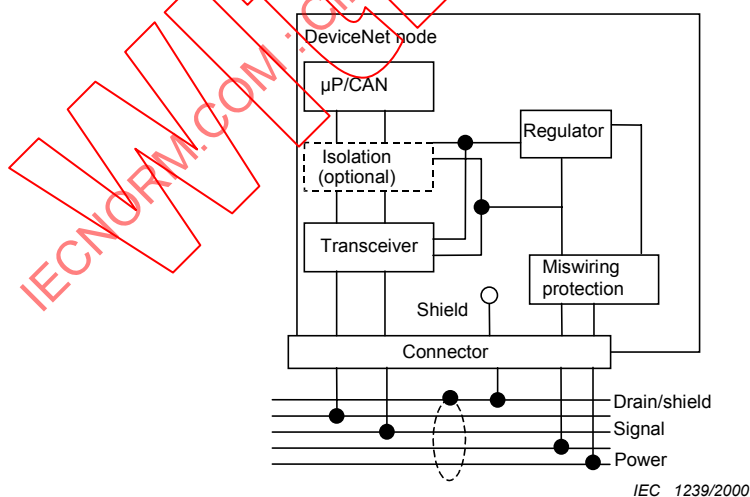
1.6.2 Transcepteur

Un transcepteur doit assurer la transmission et la réception des signaux CAN vers et depuis la liaison. Il doit recevoir les signaux différentiels de la liaison et les transmettre au processeur CAN, et il doit transmettre les signaux différentiels du processeur CAN vers la liaison.

Table 41 – General physical layer characteristics

Characteristic	Specification
Bit rates	125 kbit/s, 250 kbit/s, 500 kbit/s
Maximum total length of trunk line and maximum cable length between any two devices	500 m at 125 kbit/s 250 m at 250 kbit/s 100 m at 500 kbit/s
Number of nodes supported by the transceiver	64 minimum
Signalling	In accordance with ISO 11898
Modulation	Baseband
Encoding	NRZ with bit stuffing
Coupling to transmission medium	DC coupled differential Tx/Rx
Isolation (between transceiver circuitry and CAN chip)	500 V d.c. Applies to isolated devices only
Differential input impedance typical (recessive state)	Shunt C = 5 pF Shunt R = 25 k Ω (power on)
Differential input impedance min. (recessive state)	Shunt C = 24 pF Shunt R = 20 k Ω (power on)
Voltage on the power conductors	11 V d.c. to 25 V d.c.
Maximum signal voltage range	-25 V d.c. to +18 V d.c. (CAN_H, CAN_L) ¹⁾
¹⁾ Voltages at CAN_H and CAN_L shall be referenced to the transceiver ground. The potential at the transceiver ground may be higher than the U ₋ terminal by an amount equal to the voltage drop across a Schottky diode or equivalent. This voltage shall be 0,6 V maximum with respect to U ₋ .	

The physical layer, comprising the transceiver, connector, miswiring protection circuitry, voltage regulator, transmission medium and optional isolation, shall be as shown in figure 36. Testing of the robustness of the physical layer implementation within a device is specified in 9.2.9.

**Figure 36 – Physical layer block diagram**

5.6.2 Transceiver

A transceiver shall provide the transmission and reception of CAN signals to and from the link. The transceiver shall receive differential signals from the link and forward them to the CAN controller, and shall transmit differential signals from the CAN controller to the link.

Pour être compatibles avec la conception du système d'alimentation, les bornes CAN_H et CAN_L du transcepteur doivent permettre un fonctionnement de ± 5 V minimum en mode commun, c'est-à-dire que des différences avec la terre de ± 5 V doivent être tolérées.

Le transcepteur doit être continuellement alimenté par la liaison.

Les caractéristiques exigées pour le transmetteur et le récepteur sont données dans le tableau 42 et le tableau 43 (voir 5.2.7 pour les spécifications d'essais).

Tableau 42 – Caractéristiques du transmetteur

Caractéristiques	Spécifications
Niveau de sortie différentiel (nominal)	2,0 V crête à crête
Niveau de sortie différentiel (minimal) (mesuré au niveau du connecteur, charge de 50 Ω)	1,5 V crête à crête
Tension dominante minimale du bus à CAN_H	2,75 V ¹⁾
Tension dominante maximale du bus à CAN_H	4,5 V ¹⁾
Tension dominante minimale du bus à CAN_L	0,5 V ¹⁾
Tension dominante maximale du bus à CAN_L	2,0 V ¹⁾
Tension récessive minimale du bus à CAN_H et CAN_L	2,0 V ¹⁾
Tension récessive maximale du bus à CAN_H et CAN_L	3,0 V ¹⁾
Délai maximal du transmetteur (avec isolation)	120 ns
Protection des sorties contre les courts-circuits	Limitée en interne
¹⁾ Les tensions à CAN_H et à CAN_L doivent avoir pour référence la terre du transcepteur. Le potentiel à la terre du transcepteur peut être supérieur à celui de la borne U_- d'une quantité égale à la chute de tension au travers d'une diode Schottky ou équivalent. Cette tension doit être de 0,6 V maximum relativement à U_- .	

Tableau 43 – Caractéristiques du récepteur

Caractéristiques	Spécifications
Tension d'entrée différentielle dominante	0,95 V minimum
Tension d'entrée différentielle récessive	0,45 V maximum
Hystérésis	150 mV type
Délai maximal du récepteur (avec isolation)	130 ns
Plage de tensions de fonctionnement (CAN_H et CAN_L)	-5 à $+10$ V ¹⁾
¹⁾ Les tensions à CAN_H et à CAN_L doivent avoir pour référence la terre du transcepteur. Le potentiel à la terre du transcepteur peut être supérieur à celui de la borne U_- d'une quantité égale à la chute de tension au travers d'une diode Schottky ou équivalent. Cette tension doit être de 0,6 V maximum relativement à U_- .	

Un appareil doit avoir un délai d'acquiescement inférieur ou égal à 302 ns. Ce délai comprend les délais de propagation du transmetteur, du récepteur et du processeur CAN. Toute combinaison de ces délais est autorisée, à condition que le temps total soit inférieur ou égal à 302 ns et que les délais maximaux du transmetteur et du récepteur donnés respectivement dans le tableau 42 et le tableau 43 ne soient pas dépassés.

NOTE Le délai maximal indiqué pour le transmetteur au tableau 42 est de 120 ns et le délai maximal indiqué pour le récepteur au tableau 43 est de 130 ns. Un délai maximal de 53 ns pour le processeur CAN garantit que le délai d'acquiescement ne dépasse pas 302 ns.

Voir 5.2.8 pour les spécifications d'essais.

To be compatible with the power system design, the transceiver CAN_H and CAN_L terminals shall support a minimum of ± 5 V common mode operation, i.e. it shall tolerate ground differences of ± 5 V.

The transceiver shall be continuously powered by the link.

The required characteristics for the transmitter and the receiver are given in table 42 and table 43 (see 9.2.7 for test specifications).

Table 42 – Transmitter characteristics

Transmitter characteristic	Specification
Differential output level (nominal)	2,0 V peak to peak
Differential output level (minimum) (measured at connector, 50 Ω load)	1,5 V peak to peak
Minimum dominant bus voltage at CAN_H	2,75 V ¹⁾
Maximum dominant bus voltage at CAN_H	4,5 V ¹⁾
Minimum dominant bus voltage at CAN_L	0,5 V ¹⁾
Maximum dominant bus voltage at CAN_L	2,0 V ¹⁾
Minimum recessive bus voltage at CAN_H and CAN_L	2,0 V ¹⁾
Maximum recessive bus voltage at CAN_H and CAN_L	3,0 V ¹⁾
Maximum transmitter delay (including isolation)	120 ns
Output short-circuit protection	Internally limited
¹⁾ Voltages at CAN_H and CAN_L shall be referenced to the transceiver ground. The potential at the transceiver ground may be higher than the U ₋ terminal by an amount equal to the voltage drop across a Schottky diode or equivalent. This voltage shall be 0,6 V maximum with respect to U ₋ .	

Table 43 – Receiver characteristics

Receiver characteristic	Specification
Differential input voltage dominant	0,95 V minimum
Differential input voltage recessive	0,45 V maximum
Hysteresis	150 mV typical
Maximum receiver delay (including isolation)	130 ns
Operating voltage range (CAN_H and CAN_L)	-5 to +10 V ¹⁾
¹⁾ Voltages at CAN_H and CAN_L shall be referenced to the transceiver ground. The potential at the transceiver ground may be higher than the U ₋ terminal by an amount equal to the voltage drop across a Schottky diode or equivalent. This voltage shall be 0,6 V maximum with respect to U ₋ .	

A device shall have an acknowledgement delay which is less than or equal to 302 ns. The acknowledgement delay comprises the propagation delays of the transmitter, receiver and CAN controller. Any combination of these delays is permitted provided that the total time is less than or equal to 302 ns and that the maximum transmitter and receiver delays given in table 42 and table 43 respectively are not exceeded.

NOTE The maximum transmitter delay time in table 42 is 120 ns and the maximum receiver delay time in table 43 is 130 ns. A maximum CAN controller delay time of 53 ns ensures that the acknowledgement delay does not exceed 302 ns.

See 9.2.8 for test specifications.

NOTE La conformité électromagnétique la meilleure peut être obtenue en gardant un conducteur très court le long de ce trajet et avec un boîtier à structure fermée, fait d'un matériau conducteur. Si l'appareil n'est pas doté d'un tel boîtier, la broche du blindage du connecteur peut être laissée non connectée.

5.6.3 Grounding

To prevent ground loops, the link shall be grounded in only one location, preferably at the power supply. The physical layer circuitry in all devices shall be referenced to U_- . A device shall not cause current to flow between U_- and ground.

5.6.4 Isolation

5.6.4.1 General

Devices shall be either non-isolated devices or isolated devices.

NOTE Isolated devices and non-isolated devices may co-exist and communicate through CDIs.

5.6.4.2 Non-isolated devices

Within a device that contains a non-isolated physical layer, components which are ground referenced to U_- may connect to other external devices. These external devices shall be ground isolated, and this requirement shall be stated in the manufacturer's documentation.

The shield connection on the DeviceNet connector should be connected through a parallel RC circuit ($R = 1\text{ M}\Omega$, $C = 0,01\text{ }\mu\text{F}$ (500 V)) to the device enclosure.

NOTE The best electromagnetic conformance may be achieved with the conductor kept very short along this path and with the enclosure being a closed structure of conductive material. If the device does not have such an enclosure, the shield pin of the connector may be left unconnected.

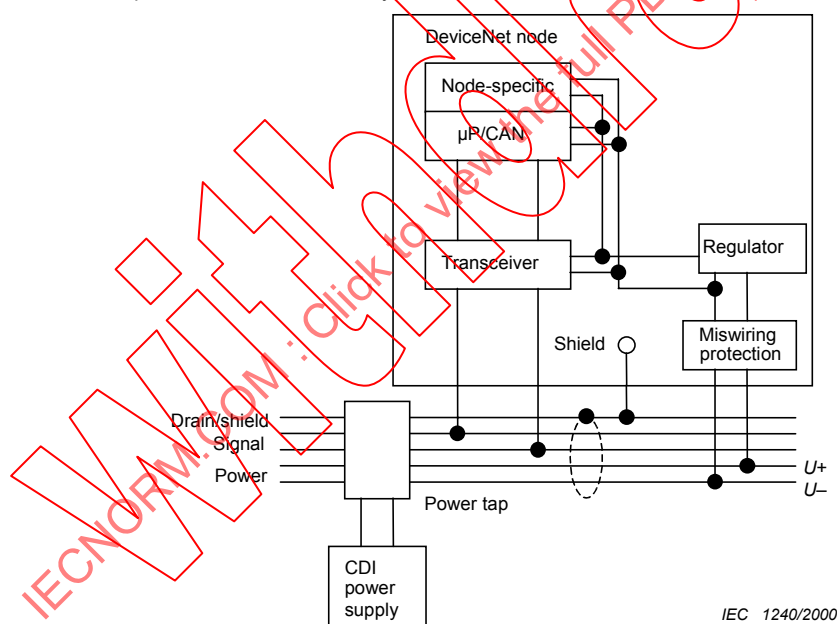


Figure 37 – Device containing a non-isolated physical layer

5.6.4.3 Isolated device

These devices shall obtain power from the link for the transceiver and isolation circuitry (see figure 38). Link power may be used for other circuitry provided that this link-powered circuitry is ground referenced to U_- or is otherwise ground isolated.

Pour un appareil isolé, le processeur CAN peut être encore actif alors que la ligne d'alimentation n'est plus sous tension. Les appareils isolés doivent être capables de détecter que leurs transcepteurs sont hors tension.

Dans un appareil isolé, les composants ayant U_- comme référence de terre peuvent se connecter à d'autres appareils externes par l'intermédiaire d'accès série, d'accès parallèles ou de connexions d'E/S. Ces appareils externes doivent être isolés de la terre et cet impératif doit être indiqué dans la documentation du fabricant.

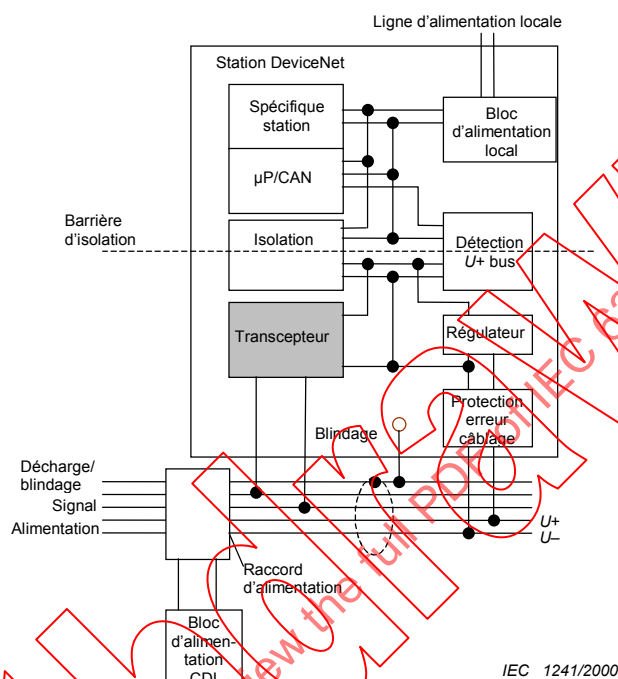


Figure 38 – Appareil contenant une couche physique isolée

Il convient que la connexion du blindage de la liaison au boîtier de l'appareil s'effectue via un circuit RC parallèle.

1.6.5 Support de transmission

DeviceNet utilise deux paires torsadées blindées dans un même câble. L'une de ces paires fournit un support de communication différentiel et l'autre alimente les appareils.

DeviceNet offre deux types de câbles: épais et fin. Le câble épais permet de fournir une plus grande quantité de courant et permet de communiquer sur de plus grandes distances. Les deux câbles, épais et fin, peuvent être utilisés pour les lignes principales et/ou les lignes de dérivation, dans n'importe quelle combinaison.

1.6.6 Topologie

1.6.6.1 Généralités

Le support de communication DeviceNet doit utiliser une topologie linéaire. Quand elles sont utilisées, les lignes de dérivation permettent le raccordement d'une ou plusieurs stations. Ces structures de branchement ne sont permises que sur les lignes de dérivation.

Des résistances de terminaison doivent être connectées à chaque extrémité de la ligne principale. La distance entre deux points quelconques du câble ne doit pas dépasser la longueur maximale de câble autorisée pour la vitesse de transmission.

For an isolated device, the CAN controller may still be active during a de-energized power line condition. Isolated devices shall be capable of sensing when their transceivers have become de-energized.

Within an isolated device, components which are ground referenced to U_- may connect to other external devices through serial ports, parallel ports or I/O connections. These external devices shall be ground isolated and this requirement shall be stated in the manufacturer's documentation.

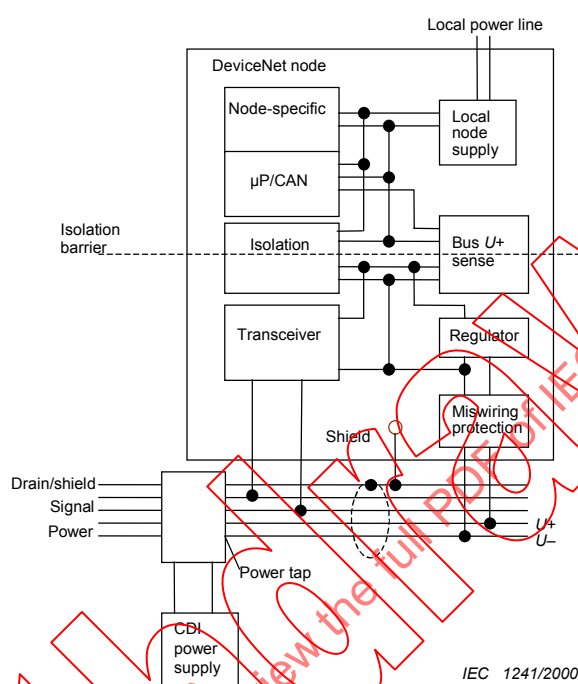


Figure 38 – Device containing an isolated physical layer

The shield connection of the link should be connected via a parallel RC circuit to the device enclosure.

5.6.5 Transmission medium

DeviceNet uses two twisted shielded conductor pairs within one cable. One of these pairs provides a differential communication medium and the other pair provides power to the devices.

DeviceNet provides two different types of cable: thick cable and thin cable. Thick cable provides higher current capability and allows communication over longer distances. Both thick and thin cable may be used for trunk lines and/or drop lines in any combination.

5.6.6 Topology

5.6.6.1 General

The DeviceNet medium shall have a linear topology. If used, drop lines may allow one or more nodes to be attached. Branching structures are only allowed on the drop line.

Terminating resistors shall be connected at each end of the trunk line. The cable distance between any two points in the cable system shall not exceed the maximum cable distance allowed for the bit rate.

1.6.6.2 Lignes principales

Le câble épais ou le câble fin peuvent être utilisés pour construire des lignes principales. La distance maximale de câble avec une combinaison de câble épais et de câble fin doit être calculée à l'aide des formules ci-dessous:

$$l_{\text{épais}} + 5 \times l_{\text{fin}} = 500 \text{ m} \quad \text{à } 125 \text{ kbit/s}$$

$$l_{\text{épais}} + 2,5 \times l_{\text{fin}} = 250 \text{ m} \quad \text{à } 250 \text{ kbit/s}$$

$$l_{\text{épais}} + l_{\text{fin}} = 100 \text{ m} \quad \text{à } 500 \text{ kbit/s}$$

où $l_{\text{épais}}$ est la longueur de câble épais et l_{fin} est la longueur de câble fin.

1.6.6.3 Lignes de dérivation

La longueur d'une ligne de dérivation est la distance de câble mesurée entre le dérivateur sur la ligne principale et le transcepteur de chaque appareil et elle ne doit pas dépasser 6 m. La longueur totale des lignes de dérivation admise pour la liaison dépend de la vitesse de transmission et ne doit pas dépasser les valeurs spécifiées dans le tableau 44.

Tableau 44 – Longueur totale de câble des lignes de dérivation

Débit binaire kbit/s	Longueur de câble m
125	156
250	78
500	39

1.6.7 Alimentation de la liaison

1.6.7.1 Configuration de l'alimentation

L'alimentation de la liaison doit être fournie par une source nominale de 24 V c.c. et peut accepter jusqu'à 8 A sur tout segment de ligne principale à câble épais ou jusqu'à 3 A sur tout segment de ligne principale à câble fin. Plusieurs blocs d'alimentation peuvent être utilisés.

1.6.7.2 Limites de charge

Le courant maximal sur la liaison se détermine à l'aide des informations contenues au tableau 45 (voir 5.2.2 pour les spécifications d'essais).

Tableau 45 – Limites de charge

Caractéristiques	Spécifications
Chute de tension maximale sur U_- et U_+	5 V chacune
Courant maximal de la ligne principale à câble épais	8 A
Courant maximal de la ligne principale à câble fin	3 A
Plage maximale de courants des lignes de dérivation	0,75 A à 3,0 A ¹⁾
Plage de tensions à chaque station	11 V à 25 V
Courant de fonctionnement de chaque appareil ²⁾	Spécifié par le fabricant
¹⁾ Le courant maximal d'une ligne de dérivation dépend de la longueur de la dérivation (voir ci-dessous). ²⁾ Le courant de fonctionnement représente le courant moyen prélevé depuis la liaison. Le pic de courant de fonctionnement doit être spécifié s'il dépasse le courant moyen de plus de 10 %.	

5.6.6.2 Trunk lines

Either thick or thin cable may be used to construct trunk lines. The maximum cable distance with a mix of both thick and thin cable shall be calculated using the formulas below:

$$l_{\text{thick}} + 5 \times l_{\text{thin}} = 500 \text{ m} \quad \text{at } 125 \text{ kbit/s}$$

$$l_{\text{thick}} + 2,5 \times l_{\text{thin}} = 250 \text{ m} \quad \text{at } 250 \text{ kbit/s}$$

$$l_{\text{thick}} + l_{\text{thin}} = 100 \text{ m} \quad \text{at } 500 \text{ kbit/s}$$

where l_{thick} is the length of thick cable and l_{thin} is the length of thin cable.

5.6.6.3 Drop lines

The drop line length is the cable distance measured from the tap on the trunk line to the transceiver of each device and shall not exceed 6 m. The total length of drop line cable allowable on the link depends upon the bit rate and shall not exceed the values specified in table 44.

Table 44 – Total drop line cable length

Bit rate kbits	Cable length m
125	156
250	78
500	39

5.6.7 Link power

5.6.7.1 Power configuration

The link power shall be supplied by a rated 24 V d.c. source and can support up to 8 A on any section of thick cable trunk line or up to 3 A on any section of thin cable trunk line. Multiple power supplies can be used.

5.6.7.2 Load limits

The maximum link current shall be determined using the information shown in table 45 (see 9.2.2 for test specifications).

Table 45 – Load limits

Characteristic	Specification
Maximum voltage drop on U_- and U_+	5 V each
Maximum thick cable trunk line current	8 A
Maximum thin cable trunk line current	3 A
Maximum drop line current range	0,75 A to 3,0 A ¹⁾
Voltage range at each node	11 V to 25 V
Operating current of each device ²⁾	Specified by manufacturer
¹⁾ The maximum drop line current depends upon the length of the drop (see below). ²⁾ The operating current represents the average current drawn from the link. The peak operating current shall be specified if it exceeds the average current by more than 10 %.	

Le courant maximal d'une ligne de dérivation est également contraint par l'équation suivante:

$$i = 4,5 / l$$

où

i est le courant maximal acceptable pour la ligne de dérivation (A);

l est la longueur de la dérivation (m).

2 Information sur le matériel

Conforme à la CEI 62026-1.

3 Conditions normales de service, de montage et de transport

3.1 Conditions normales de service

3.1.1 Généralités

Les composants d'un CDI DeviceNet doivent pouvoir fonctionner dans les conditions suivantes.

NOTE Si les conditions de fonctionnement diffèrent de celles données dans la présente partie, il convient que l'utilisateur indique les divergences par rapport aux conditions normales et consulte le fabricant quant aux possibilités d'utilisation dans de telles conditions.

3.1.2 Température de l'air ambiant

3.1.2.1 Câble épais

Le câble doit fonctionner normalement dans une plage de températures ambiantes de -20 °C à $+60\text{ °C}$ lorsqu'il transporte un courant de 8 A. Ce courant nominal doit être corrigé linéairement à zéro à $+80\text{ °C}$.

3.1.2.2 Câble fin

Le câble doit fonctionner normalement dans une plage de températures ambiantes de -20 °C à $+70\text{ °C}$ lorsqu'il transporte un courant de 1,5 A. Ce courant nominal doit être corrigé linéairement à zéro à $+80\text{ °C}$.

3.1.2.3 Dérivateurs d'appareil

Les dérivateurs d'appareil doivent fonctionner normalement dans une plage de températures ambiantes de -40 °C à $+70\text{ °C}$ lorsqu'ils véhiculent la totalité du courant. Ce courant nominal doit être corrigé linéairement à zéro à $+80\text{ °C}$. Le courant continu maximal sur les conducteurs d'alimentation est de 3 A pour les microconnecteurs et de 8 A pour les autres connecteurs, sauf indication contraire.

3.1.2.4 Raccords d'alimentation

Les raccords d'alimentation doivent fonctionner normalement dans une plage de températures ambiantes de -40 °C à $+70\text{ °C}$ lorsqu'ils véhiculent la totalité du courant. Ce courant nominal doit être corrigé linéairement à zéro à $+80\text{ °C}$.

The maximum drop line current is also constrained by the following equation:

$$i = 4,5 / l$$

where

i is the maximum allowable drop line current (A);

l is the drop length (m).

6 Product information

In accordance with IEC 62026-1.

7 Normal service, mounting and transport conditions

1.1 Normal service conditions

7.1.1 General

Components of a DeviceNet CDI shall be capable of operating under the following conditions.

NOTE If the conditions for operation differ from those given in this part, the user should state the deviation from the standard conditions and consult the manufacturer on the suitability for use under such conditions.

7.1.2 Ambient air temperature

7.1.2.1 Thick cable

The cable shall operate normally within an ambient temperature range of $-20\text{ }^{\circ}\text{C}$ to $+60\text{ }^{\circ}\text{C}$ when carrying a current of 8 A. This current rating shall be derated linearly to zero at $+80\text{ }^{\circ}\text{C}$.

7.1.2.2 Thin cable

The cable shall operate normally within an ambient temperature range of $-20\text{ }^{\circ}\text{C}$ to $+70\text{ }^{\circ}\text{C}$ when carrying a current of 1,5 A. This current rating shall be derated linearly to zero at $+80\text{ }^{\circ}\text{C}$.

7.1.2.3 Device taps

Device taps shall operate normally within an ambient temperature range of $-40\text{ }^{\circ}\text{C}$ to $+70\text{ }^{\circ}\text{C}$ when carrying full current. This current rating shall be derated linearly to zero at $+80\text{ }^{\circ}\text{C}$. The maximum continuous current on power conductors is 3 A for microconnectors and 8 A for other connectors, unless otherwise specified.

7.1.2.4 Power taps

Power taps shall operate normally within an ambient temperature range of $-40\text{ }^{\circ}\text{C}$ to $+70\text{ }^{\circ}\text{C}$ when carrying full current. This current rating shall be derated linearly to zero at $+80\text{ }^{\circ}\text{C}$.

3.1.2.5 Autres composants du CDI

Tous les autres composants d'un CDI DeviceNet doivent fonctionner pour les températures ambiantes de -25 °C à $+70\text{ °C}$ sauf indication contraire, par exemple conjointement avec un type spécifique d'actionneur ou de capteur. Les caractéristiques de fonctionnement doivent être maintenues dans toute la plage admissible de températures ambiantes.

3.1.3 Altitude

Les composants DeviceNet doivent être capables de fonctionner à des altitudes conformes à la CEI 62026-1.

3.1.4 Conditions climatiques

3.1.4.1 Humidité

Les composants DeviceNet doivent être capables de fonctionner à $+40\text{ °C}$ avec une humidité relative de l'air ne dépassant pas 95 %.

3.1.4.2 Degré de pollution

Les composants DeviceNet doivent être capables de fonctionner dans des conditions de pollution conformes à la CEI 62026-1.

3.2 Conditions durant le transport et le stockage

Un accord spécial doit être passé entre l'utilisateur et le fabricant si les conditions de transport et de stockage sont différentes des conditions suivantes:

- humidité: humidité relative de l'air ne dépassant pas 95 % à 40 °C ;
- température de stockage: -40 °C à $+85\text{ °C}$

3.3 Montage

Les composants DeviceNet doivent être montés conformément à la CEI 62026-1.

4 Dispositions relatives à la construction et au fonctionnement

4.1 Voyants et commutateurs mécaniques de configuration

4.1.1 Voyants d'état

4.1.1.1 Généralités

Les appareils peuvent être munis des voyants d'état suivants:

- un voyant d'état pour le module et un voyant d'état pour le CDI, ou bien un voyant d'état combinant module/CDI;
- un voyant d'état d'E/S.

Tout autre voyant doit pouvoir se distinguer clairement de ceux définis ci-dessus.

4.1.1.2 Voyant d'état du module

Voyant bicolore (vert/rouge) qui doit indiquer si les circuits DeviceNet sont alimentés ou non et s'ils fonctionnent correctement. Les états du voyant sont spécifiés dans le tableau 46.

7.1.2.5 Other CDI components

All other components of a DeviceNet CDI shall operate between the ambient temperatures of $-25\text{ }^{\circ}\text{C}$ and $+70\text{ }^{\circ}\text{C}$ if not otherwise defined, for example in conjunction with a specific actuator or sensor type. The operating characteristics shall be maintained over the permissible range of ambient temperature.

7.1.3 Altitude

DeviceNet components shall be capable of operating at altitudes in accordance with IEC 62026-1.

7.1.4 Climatic conditions

7.1.4.1 Humidity

DeviceNet components shall be capable of operating at $40\text{ }^{\circ}\text{C}$, with the relative humidity of the air not exceeding 95 %.

7.1.4.2 Pollution degree

DeviceNet components shall be capable of operating in polluted conditions in accordance with IEC 62026-1.

7.2 Conditions during transport and storage

A special agreement shall be made between the user and the manufacturer if the conditions during transport and storage differ from the following:

- humidity: relative humidity of the air not exceeding 95 % at $40\text{ }^{\circ}\text{C}$;
- storage temperature: $-40\text{ }^{\circ}\text{C}$ to $+85\text{ }^{\circ}\text{C}$

7.3 Mounting

DeviceNet components shall be mounted in accordance with IEC 62026-1.

8 Constructional and performance requirements

8.1 Indicators and configuration switches

8.1.1 Status indicators

8.1.1.1 General

Devices may have the following status indicators:

- one module status indicator and one CDI status indicator, or one combined module/CDI status indicator;
- I/O status indicator.

Any other indicators shall be clearly distinguishable from those defined above.

8.1.1.2 Module status indicator

This shall be a bicolour (green/red) indicator and shall indicate whether or not the DeviceNet circuitry is powered and operating correctly. The indicator states shall be as specified in table 46.

Tableau 46 – Voyant d'état du module

Etat du voyant	Signification
Eteint	Non alimenté
Vert	Fonctionnement correct
Vert clignotant	Configuration manquante, incomplète ou incorrecte
Rouge clignotant	Défaut récupérable
Rouge	Défaut non récupérable
Clignotant rouge-vert	Auto-test en cours
NOTE Les vitesses de clignotement du voyant sont indiquées en 4.1.1.8.	

4.1.1.3 Voyant d'état du CDI

Voyant bicolore (vert/rouge) qui doit indiquer l'état de la liaison de communication, comme indiqué au tableau 47.

Tableau 47 – Voyant d'état du CDI

Etat du voyant	Signification
Eteint	La vérification de duplication d'adresses MAC ID n'a pas été effectuée ou les circuits DeviceNet ne sont pas alimentés
Vert clignotant	L'appareil est en ligne mais aucune connexion n'est établie
Vert	L'appareil est en ligne et les connexions sont établies
Rouge clignotant	Expiration du délai d'une ou de plusieurs connexions d'E/S
Rouge	L'appareil a détecté une erreur et ne peut pas communiquer sur la liaison
NOTE Les vitesses de clignotement du voyant sont indiquées en 4.1.1.8.	

4.1.1.4 Voyants d'état du module et du CDI à la mise sous tension

La séquence suivante d'essais des voyants doit être effectuée à la mise sous tension:

- voyant d'état du CDI éteint;
- voyant d'état du module allumé vert pendant $0,25 \text{ s} \pm 0,1 \text{ s}$;
- voyant d'état du module allumé rouge pendant $0,25 \text{ s} \pm 0,1 \text{ s}$;
- voyant d'état du module allumé vert;
- voyant d'état du CDI allumé vert pendant $0,25 \text{ s} \pm 0,1 \text{ s}$;
- voyant d'état du CDI allumé rouge pendant $0,25 \text{ s} \pm 0,1 \text{ s}$;
- voyant d'état du CDI éteint;
- les deux voyants se positionnent conformément au tableau 46 et au tableau 47.

4.1.1.5 Voyant d'état combiné module/CDI

Voyant bicolore (vert/rouge) qui doit fournir des informations limitées sur l'état de l'appareil et des communications.

Les états du voyant combiné module/CDI sont spécifiés dans le tableau 48.

Table 46 – Module status indicator

Indicator status	Meaning
Off	Not powered
Green	Operating correctly
Flashing green	Missing, incomplete or incorrect configuration
Flashing red	Recoverable fault
Red	Unrecoverable fault
Flashing red-green	Undergoing self-test
NOTE Indicator flash rates are given in 8.1.1.8.	

8.1.1.3 CDI status indicator

This shall be a bicolour (green/red) indicator and shall indicate the status of the communication link as specified in table 47.

Table 47 – CDI status indicator

Indicator status	Meaning
Off	The duplicate MAC ID detection test has not been completed or the DeviceNet circuitry is not powered
Flashing green	Device is on-line but has no connections established
Green	Device is on-line and has connections established
Flashing red	One or more I/O connections have timed-out
Red	The device has detected an error and cannot communicate on the link
NOTE Indicator flash rates are given in 8.1.1.8.	

8.1.1.4 Module and CDI status indicators at power-up

The following indicator test sequence shall be performed at power-up:

- CDI status indicator OFF;
- module status indicator ON green for $0,25\text{ s} \pm 0,1\text{ s}$;
- module status indicator ON red for $0,25\text{ s} \pm 0,1\text{ s}$;
- module status indicator ON green;
- CDI status indicator ON green for $0,25\text{ s} \pm 0,1\text{ s}$;
- CDI status indicator ON red for $0,25\text{ s} \pm 0,1\text{ s}$;
- CDI status indicator OFF;
- both indicators assume states according to table 46 and table 47.

8.1.1.5 Combined module/CDI status indicator

This shall be a bicolour (green/red) indicator and shall provide limited device and communication status information.

The combined module/CDI status indicator states shall be as specified in table 48.

Tableau 48 – Voyant d'état combiné module/CDI

Etat du voyant	Signification
Eteint	L'appareil n'est pas en ligne: – l'appareil n'a peut-être pas terminé la vérification de duplication d'adresses MAC ID – l'appareil n'est peut-être pas alimenté
Vert	L'appareil fonctionne de manière normale, il est en ligne avec des connexions établies
Vert clignotant	L'appareil fonctionne de manière normale et, soit il est en ligne sans connexion établie, soit la configuration est manquante, incomplète ou incorrecte
Rouge clignotant	Défaut récupérable et/ou une ou plusieurs connexions d'E/S sont en état de délai expiré
Rouge	L'appareil a un défaut non récupérable qui l'a rendu incapable de communiquer sur la liaison
NOTE Les vitesses de clignotement du voyant sont données en 4.1.1.8.	

4.1.1.6 Voyant d'état combiné module/CDI à la mise sous tension

La séquence suivante d'essais du voyant doit être effectuée à la mise sous tension:

- voyant d'état combiné module/CDI allumé vert pendant $0,25 \text{ s} \pm 0,1 \text{ s}$;
- voyant d'état combiné module/CDI allumé rouge pendant $0,25 \text{ s} \pm 0,1 \text{ s}$;
- voyant d'état combiné module/CDI éteint,
- le voyant se positionne conformément au tableau 48.

4.1.1.7 Voyant d'état des E/S

Voyant bicolore (vert/rouge) qui doit fournir des informations concernant l'état des entrées et/ou des sorties.

Les états du voyant d'état des E/S sont spécifiés dans le tableau 49.

Tableau 49 – Voyant d'état des E/S

Etat du voyant	Signification
Eteint	Toutes les sorties sont inactives Toutes les entrées sont inactives
Vert clignotant	Une ou plusieurs sorties sont en attente et aucune sortie n'est défectueuse
Vert	Une ou plusieurs sorties sont actives et sous contrôle, aucune n'est défectueuse Une ou plusieurs entrées sont actives et produisent des données; aucune entrée n'est défectueuse
Rouge clignotant	Une ou plusieurs sorties sont défectueuses Une ou plusieurs entrées sont défectueuses
Rouge	Une ou plusieurs sorties sont forcées à l'état inactif (il peut s'agir d'un défaut non récupérable) Une ou plusieurs entrées sont en état de défaut non récupérable
NOTE Les vitesses de clignotement du voyant sont données en 4.1.1.8.	

Table 48 – Combined module/CDI status indicator

Indicator status	Meaning
Off	Device is not on-line: – the device may not have completed the duplicate MAC ID test – the device may not be powered
Green	The device is operating in a normal condition and the device is on-line with connections in the established state
Flashing green	The device is operating in a normal condition and either the device is on-line with no connections in the established state or configuration is missing, incomplete or incorrect
Flashing red	Recoverable fault and/or one or more I/O connections are in the timed-out state
Red	The device has an unrecoverable fault that has rendered it incapable of communicating on the link
NOTE Indicator flash rates are given in 8.1.1.8.	

8.1.1.6 Combined module/CDI status indicator at power-up

The following indicator test sequence shall be performed at power-up:

- combined module/CDI status indicator ON green for $0,25\text{ s} \pm 0,1\text{ s}$;
- combined module/CDI indicator ON red for $0,25\text{ s} \pm 0,1\text{ s}$;
- combined module/CDI indicator OFF;
- indicator assumes state according to table 48.

8.1.1.7 I/O status indicator

This shall be a bicolour (green/red) indicator and shall provide information concerning the states of inputs and/or outputs.

The I/O status indicator states shall be as specified in table 49.

Table 49 – I/O status indicator

Indicator status	Meaning
Off	All outputs are inactive All inputs are inactive
Flashing green	One or more outputs are idle and no outputs are faulted
Green	One or more outputs are active and under control, no outputs are faulted One or more inputs are active and producing data, no inputs are faulted
Flashing red	One or more outputs are faulted One or more inputs are faulted
Red	One or more outputs are forced OFF (may be an unrecoverable fault) One or more inputs have an unrecoverable fault
NOTE Indicator flash rates are given in 8.1.1.8.	

4.1.1.8 Fréquence de clignotement des voyants

Sauf indication contraire dans la présente partie, la fréquence de clignotement d'un voyant doit être de 1 éclair par seconde $\pm 0,5$ s. Un voyant doit être allumé pendant $0,5 \text{ s} \pm 0,25 \text{ s}$ et éteint pendant $0,5 \text{ s} \pm 0,25 \text{ s}$.

4.1.2 Commutateurs de vitesse de transmission pour DeviceNet

Si des commutateurs mécaniques sont utilisés pour choisir la vitesse de transmission sur DeviceNet, ils doivent être codés comme indiqué au tableau 50.

Tableau 50 – Codage des commutateurs de vitesse de transmission pour DeviceNet

Débit binaire kbit/s	Réglage des commutateurs
125	0
250	1
500	2

4.1.3 Marquage des voyants et commutateurs

Le marquage des voyants et commutateurs mécaniques doit comporter leur nom complet ou abrégé, comme indiqué au tableau 51.

Tableau 51 – Marquage des voyants et commutateurs

Description	Nom complet	Nom abrégé
Voyant d'état du module	Module status	MS
Voyant d'état du CDI	Network status	NS
Voyant d'état combiné module/CDI	Module/network status ou Mod/net status	MNS
Voyant d'état des E/S	I/O status ou I/O	IO
Commutateurs de MAC ID	Node address	NA
Commutateurs de vitesse de transmission	Data rate	DR

4.2 Câble DeviceNet

La configuration physique des câbles épais et fins doit être comme indiqué à la figure 39.

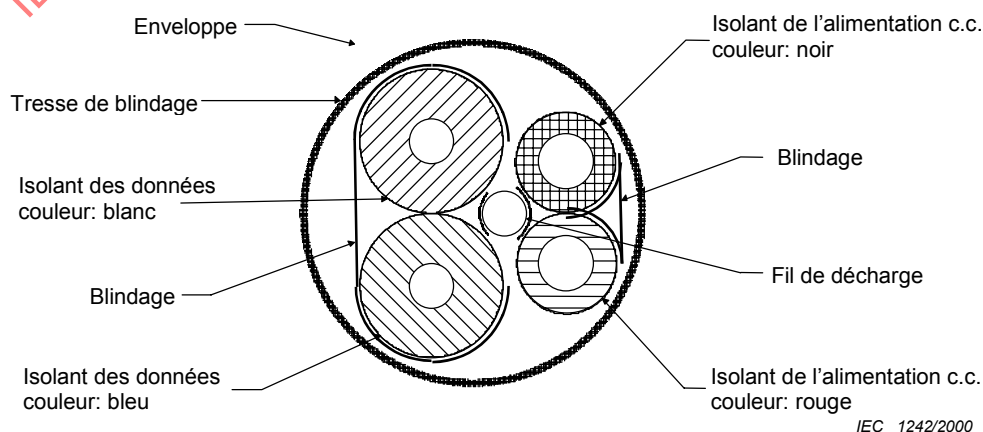


Figure 39 – Configuration physique des câbles épais et fins

8.1.1.8 Indicator flash rate

Unless otherwise specified in this part, the flash rate of an indicator shall be 1 flash per second $\pm 0,5$ s. An indicator shall be ON for $0,5$ s $\pm 0,25$ s and OFF for $0,5$ s $\pm 0,25$ s.

8.1.2 DeviceNet bit rate switches

If switches are used to set the DeviceNet bit rate, they shall be encoded as shown in table 50.

Table 50 – DeviceNet bit rate switch encoding

Bit rate kbit/s	Switch setting
125	0
250	1
500	2

8.1.3 Indicator and switch marking

Indicators and switches shall be marked with either their full names or their abbreviated names, as specified in table 51.

Table 51 – Indicator and switch marking

Description	Full name	Abbreviated name
Module status indicator	Module status	MS
CDI status indicator	Network status	NS
Combined module/CDI status indicator	Module/network status or Mod/net status	MNS
I/O status indicator	I/O status or I/O	IO
MAC ID switches	Node address	NA
Bit rate switches	Data rate	DR

8.2 DeviceNet cable

The physical configuration for both thick and thin cables shall be as shown in figure 39.

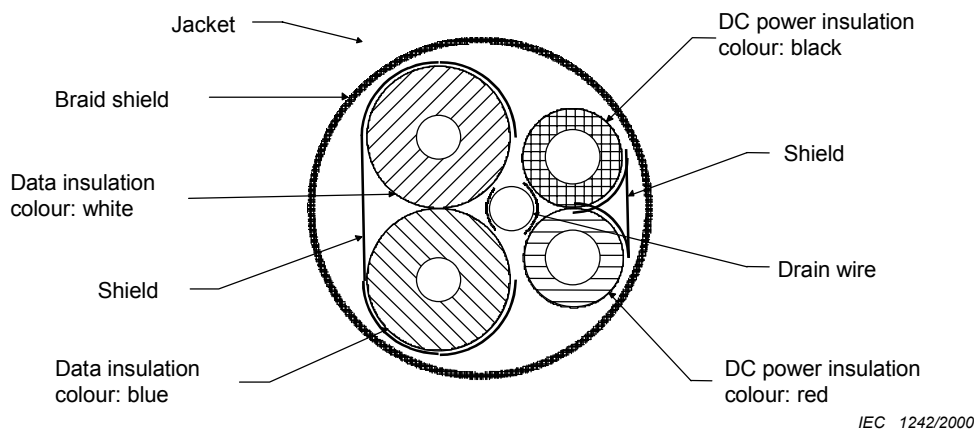


Figure 39 – Physical configuration of thick and thin cable

Le câble doit être conforme aux spécifications données dans le tableau 52, le tableau 53 et le tableau 54.

Tableau 52 – Câble: spécifications de la paire des données

Caractéristiques	Spécifications du câble épais	Spécifications du câble fin
Taille de la paire de conducteurs	0,823 mm ² en cuivre (étamé individuellement)	0,205 mm ² en cuivre (étamé individuellement)
Couleurs	Bleu, blanc	Bleu, blanc
Torsion nominale d'une paire	10 / m	16 / m
Blindage sur la paire	98 % de couverture	98 % de couverture
Impédance	120 Ω ±10 % à 1 MHz	120 Ω ±10 % à 1 MHz
Délai maximal de propagation	4,46 ns/m	4,46 ns/m
Capacité entre conducteurs	39 pF/m à 1 MHz (nominale)	39 pF/m à 1 MHz (nominale)
Capacité nominale entre un conducteur et un autre, connecté au blindage	79 pF/m à 1 MHz	79 pF/m à 1 MHz
Déséquilibre capacitif maximal	2,0 pF/m à 1 MHz	2,0 pF/m à 1 MHz
Résistance à +20 °C	Nominale 18 Ω / 1 000 m Maximale 23 Ω / 1 000 m	Nominale 69 Ω / 1 000 m Maximale 92 Ω / 1 000 m
Atténuation maximale	0,43 dB / 100 m à 125 kHz 0,82 dB / 100 m à 500 kHz 1,2 dB / 100 m à 1 MHz	0,95 dB / 100 m à 125 kHz 1,6 dB / 100 m à 500 kHz 2,3 dB / 100 m à 1 MHz

Tableau 53 – Câble: spécifications de la paire d'alimentation à courant continu

Caractéristiques physiques	Spécifications du câble épais	Spécifications du câble fin
Taille de la paire de conducteurs	1,65 mm ² en cuivre (étamé individuellement)	0,326 mm ² en cuivre (étamé individuellement)
Couleurs	Rouge, noir	Rouge, noir
Torsion nominale d'une paire	10 / m	16 / m
Blindage sur la paire	98 % de couverture	98 % de couverture
Résistance à +20 °C	Nominale 11 Ω / 1 000 m Maximale 12 Ω / 1 000 m	Nominale 45,9 Ω / 1 000 m Maximale 57,4 Ω / 1 000 m

Tableau 54 – Câble: spécifications générales

Caractéristiques	Spécifications du câble épais	Spécifications du câble fin
Deux paires blindées	Axe commun avec fil de décharge au centre	Axe commun avec fil de décharge au centre
Gaine de blindage tressée	65 % de couverture, 0,126 mm ² tresse cuivre (étamée individuellement)	65 % de couverture, 0,126 mm ² tresse cuivre (étamée individuellement)
Fil de décharge	0,823 mm ² en cuivre (étamé individuellement)	0,326 mm ² en cuivre (étamé individuellement)
Résistance à +20 °C (blindages et fil de décharge)	5,74 Ω / 1 000 m	10,5 Ω / 1 000 m

The cable shall be in accordance with the specifications given in table 52, table 53 and table 54.

Table 52 – Cable: data pair specification

Characteristic	Thick cable specification	Thin cable specification
Conductor pair size	0,823 mm ² copper (individually tinned)	0,205 mm ² copper (individually tinned)
Colours	Blue, white	Blue, white
Nominal pair twist	10 / m	16 / m
Shield over pair	98 % coverage	98 % coverage
Impedance	120 Ω ± 10 % at 1 MHz	120 Ω ± 10 % at 1 MHz
Maximum propagation delay	4,46 ns/m	4,46 ns/m
Capacitance between conductors	39 pF/m at 1 MHz (nominal)	39 pF/m at 1 MHz (nominal)
Nominal capacitance between one conductor and another conductor connected to shield	79 pF/m at 1 MHz	79 pF/m at 1 MHz
Maximum capacitive imbalance	2,0 pF/m at 1 MHz	2,0 pF/m at 1 MHz
Resistance at +20 °C	Nominal 18 Ω / 1 000 m Maximum 23 Ω / 1 000 m	Nominal 69 Ω / 1 000 m Maximum 92 Ω / 1 000 m
Maximum attenuation	0,43 dB / 100 m at 125 kHz 0,82 dB / 100 m at 500 kHz 1,2 dB / 100 m at 1 MHz	0,95 dB / 100 m at 125 kHz 1,6 dB / 100 m at 500 kHz 2,3 dB / 100 m at 1 MHz

Table 53 – Cable: d.c. power pair specification

Physical characteristics	Thick cable specification	Thin cable specification
Conductor pair size	1,65 mm ² copper (individually tinned)	0,326 mm ² (individually tinned)
Colours	Red, black	Red, black
Nominal pair twist	10 / m	16 / m
Shield over pair	98 % coverage	98 % coverage
Resistance at +20 °C	Nominal 11 Ω / 1 000 m Maximum 12 Ω / 1 000 m	Nominal 45,9 Ω / 1 000 m Maximum 57,4 Ω / 1 000 m

Table 54 – Cable: general specifications

Characteristics	Thick cable specification	Thin cable specification
Two shielded pairs	Common axis with drain wire in centre	Common axis with drain wire in centre
Overall braid shield	65 % coverage, 0,126 mm ² Cu braid (individually tinned)	65 % coverage, 0,126 mm ² Cu braid (individually tinned)
Drain wire	0,823 mm ² copper (individually tinned)	0,326 mm ² copper (individually tinned)
Resistance at +20 °C (shields and drain wire)	5,74 Ω / 1 000 m	10,5 Ω / 1 000 m

4.3 Résistances de terminaison

Une résistance à film métallique de 121 Ω , 1 %, 0,25 W doit être connectée à chaque extrémité de la ligne principale.

Les résistances de terminaison ne doivent pas être intégrées dans les appareils.

4.4 Connecteurs

4.4.1 Spécifications générales

Tous les connecteurs doivent posséder cinq bornes: une paire pour les signaux, une paire pour l'alimentation et un blindage. Les connecteurs doivent être de type ouvert ou scellé.

Les connecteurs ouverts doivent répondre aux spécifications du tableau 55.

Tableau 55 – Spécifications générales des connecteurs ouverts

Type	Spécifications
Enfichable	5 contacts type mâle ouvert (broches) à blocage mécanique empêchant une insertion à l'envers
Câblé	Comprend la soudure directe, le sertissage, des borniers à vis, des séparateurs ou autres méthodes similaires de raccordement Une connexion câblée doit permettre de retirer un appareil sans endommager la ligne principale. Le raccordement d'appareils par ces méthodes ne doit pas être effectué pendant que la liaison est sous tension

Les connecteurs scellés doivent être protégés selon le degré de protection IP67 (voir CEI 60529) et répondre aux spécifications du tableau 56.

Tableau 56 – Spécifications générales des connecteurs scellés

Type	Spécifications
Mini-connecteur mâle	5 «mini» contacts type mâle (broches) avec connexion mâle (externe) filetée 7/8-16 UN-2A. Doit répondre aux exigences ANSI/B93.55M sur la conception et le raccordement
Mini-connecteur femelle	5 «mini» contacts type femelle (prise) avec connexion femelle (interne) filetée 7/8-16 UN-2B. Doit répondre aux exigences ANSI/B93.55M sur la conception et le raccordement
Micro-connecteur mâle	Voir la CEI 60947-5-2
Micro-connecteur femelle	Voir la CEI 60947-5-2

4.4.2 Spécifications électriques et sur les contacts

Les connecteurs doivent être conformes aux spécifications indiquées dans le tableau 57 et le tableau 58.

8.3 Terminating resistors

A 121 Ω , 1 %, 0,25 W metal film resistor shall be connected at each end of the trunk line.

Terminating resistors shall not be included in devices.

8.4 Connectors

8.4.1 General specifications

All connectors shall have five terminals: a signal pair, a power pair and a shield. Connectors shall be either the open or sealed type.

Open connectors shall comply with the specifications given in table 55.

Table 55 – Open connector general specifications

Connector style	Specification
Pluggable	5 pin open style male (pin) contacts with mechanical locking to prevent reverse insertion
Hard wired	Includes direct soldering, crimping, screw terminal blocks, barrier strips or other similar attachment methods Hard wired connection shall enable a device to be removed without severing the trunk. The attachment of devices using these methods shall not be attempted while link power is applied

Sealed connectors shall be protected to IP67 (see IEC 60529) and shall comply with the specifications given in table 56.

Table 56 – Sealed connector general specifications

Connector style	Specification
Male mini-connector	5 pin "mini" style male (pin) contacts with 7/8-16 UN-2A male (outside) connection thread. Shall meet ANSI/B93.55M design and mating requirements
Female mini-connector	5 pin "mini" style female (socket) contacts with 7/8-16 UN-2B female (inside) connection thread. Shall meet ANSI/B93.55M design and mating requirements
Male micro-connector	See IEC 60947-5-2
Female micro-connector	See IEC 60947-5-2

8.4.2 Contact and electrical specifications

Connectors shall comply with the specifications given in table 57 and table 58.

Tableau 57 – Spécifications sur les contacts

Caractéristiques physiques	Spécifications
Exigences sur le placage des contacts frottants	0,76 µm minimum d'or sur 1,3 µm minimum de nickel ou 0,13 µm minimum d'or sur 0,51 µm minimum de palladium-nickel sur 1,3 µm de nickel. L'or doit être de 24 carats
Endurance des contacts frottants	1 000 insertions / extractions
NOTE Cette exigence s'applique aux contacts pouvant subir de nombreux branchements et non aux connexions câblées semi-permanentes telles que les borniers à vis.	

Tableau 58 – Spécifications électriques

Caractéristiques électriques	Spécifications
Tension de fonctionnement	25 V c.c. minimum
Courant nominal des contacts	Les contacts de connecteur utilisés avec un câble épais doivent avoir une valeur nominale minimale de 8 A. Les contacts de connecteur utilisés avec un câble fin doivent avoir une valeur nominale minimale de 3 A. La valeur nominale du courant des contacts d'un bloc d'alimentation doit être supérieure ou égale à celle du bloc d'alimentation
Résistance des contacts	Initiale: moins de 1 mΩ. Maximum: 5 mΩ pendant toute la durée de vie

4.4.3 Disposition des contacts des connecteurs montés sur appareils

Tous les connecteurs montés sur appareils doivent posséder des contacts mâles et être disposés comme indiqué au tableau 59.

Tableau 59 – Disposition des contacts d'un connecteur

Fonction	Couleur de fil	Numéros des contacts d'un connecteur enfichable et câblé	Numéros des contacts d'un connecteur de type mini et micro
U+	Rouge	5	2
U-	Noir	1	3
CAN_H	Blanc	4	4
CAN_L	Bleu	2	5
Blindage	Nu	3	1

4.5 Dérivateurs et raccords d'alimentation

4.5.1 Dérivateurs

Les dérivateurs offrent des points de raccordement sur la ligne principale. Les appareils peuvent être connectés à la liaison directement via le dérivateur ou par une ligne de dérivation. Les dérivateurs doivent également permettre le retrait d'un appareil sans déranger le fonctionnement du CDI.

Table 57 – Contact specifications

Physical characteristics	Specification
Wiping contact plating requirements	0,76 µm gold minimum over 1,3 µm nickel minimum or 0,13 µm gold minimum over 0,51 µm palladium-nickel minimum over 1,3 µm nickel. All gold shall be 24 carat
Wiping contact life	1 000 insertions / extractions
NOTE This requirement applies to contacts that may be subjected to multiple engagements and not to hard wired semi-permanent connections such as screw terminals.	

Table 58 – Electrical specifications

Electrical characteristics	Specification
Operating voltage	25 V d.c. minimum
Contact current rating	Connector contacts for use with thick cable shall be rated at 8 A minimum. Connector contacts for use with thin cable shall be rated at 3 A minimum Power supply contact current ratings shall be equal to or greater than the power supply current rating
Contact resistance	Initial: less than 1 mΩ. Maximum: 5 mΩ throughout life

8.4.3 Contact arrangement of device-mounted connectors

All device-mounted connectors shall have male contacts and shall be arranged as shown in table 59.

Table 59 – Connector contact arrangement

Function	Wire colour	Pluggable and hard wired connector contact numbers	Sealed mini and micro style connector contact numbers
U+	Red	5	2
U–	Black	1	3
CAN_H	White	4	4
CAN_L	Blue	2	5
Shield	Bare	3	1

8.5 Device taps and power taps

8.5.1 Device taps

Device taps provide points of attachment onto the trunk line. Devices may be connected to the link either directly to the device tap or with a drop line. Device taps shall also allow the removal of a device without disrupting the CDI operation.

Les dérivateurs doivent avoir une tension nominale minimale de 25 V c.c. et répondre aux spécifications indiquées dans le tableau 60 et le tableau 61.

Tableau 60 – Spécifications des conducteurs principaux internes

Description du conducteur	Spécifications
Fil conducteur de décharge	Longueur maximale de 100 mm Résistance maximale de 2,3 mΩ
Conducteurs d'alimentation U_+ et U_-	Longueur maximale de 100 mm Résistance maximale de 1,2 mΩ. Courant nominal de 8 A
Conducteurs de signaux CAN_H et CAN_L	Longueur maximale de 100 mm Résistance maximale de 2,3 mΩ

Tableau 61 – Spécifications des conducteurs de dérivation internes

Description du conducteur	Spécifications
Fil conducteur de décharge	Longueur maximale de 180 mm et résistance maximale de 10 mΩ, mesurées entre le connecteur attachant le dérivateur à la ligne principale et un accès quelconque
Conducteurs d'alimentation U_+ et U_-	Longueur maximale de 180 mm mesurée entre le connecteur attachant le dérivateur à la ligne principale et un accès quelconque. La résistance maximale est de 10 mΩ pour les conducteurs d'une valeur nominale jusqu'à 3 A, de 4 mΩ autrement, mesurée entre le connecteur attachant le dérivateur à la ligne principale et un accès quelconque
Conducteurs de signaux CAN_H et CAN_L	Longueur maximale de 180 mm et résistance maximale de 10 mΩ, mesurées entre le connecteur attachant le dérivateur à la ligne principale et le raccordement à un accès quelconque

4.5.2 Raccords d'alimentation

Un raccord d'alimentation relie un bloc d'alimentation à la ligne principale. Lorsqu'il est raccordé à la liaison, le raccord d'alimentation doit permettre une connexion continue des fils de signal, de décharge et U_- . Un raccord d'alimentation peut également fournir:

- une protection contre les surintensités dans chaque direction à partir du raccord;
- une protection du bloc d'alimentation lorsque plusieurs blocs sont utilisés;
- une connexion au fil de blindage/décharge pour la mise à la terre de la liaison.

Les raccords d'alimentation de DeviceNet doivent avoir une tension nominale minimale de 25 V c.c. et être conformes aux spécifications données dans le tableau 62 et le tableau 63.

Tableau 62 – Spécifications des conducteurs principaux internes

Description du conducteur	Spécifications
Fil conducteur de décharge	Longueur maximale de 180 mm Résistance maximale de 4,1 mΩ
Conducteur d'alimentation U_-	Longueur maximale de 180 mm Résistance maximale de 2,1 mΩ. Courant nominal de 8 A
Conducteurs de signaux CAN_H et CAN_L	Longueur maximale de 180 mm Résistance maximale de 4,1 mΩ

Device taps shall have a minimum voltage rating of 25 V d.c. and shall comply with the specifications given in table 60 and table 61.

Table 60 – Internal pass-through conductor specifications

Conductor description	Specification
Drain wire conductor	100 mm maximum length of conductor Maximum resistance 2,3 mΩ
Power conductors <i>U+</i> and <i>U-</i>	100 mm maximum length of conductor Maximum resistance 1,2 mΩ. Rated current 8 A
Signal conductors CAN_H and CAN_L	100 mm maximum length of conductor Maximum resistance 2,3 mΩ

Table 61 – Internal drop conductor specifications

Conductor description	Specification
Drain wire conductor	180 mm maximum length and 10 mΩ maximum resistance, both measured between the trunk line tap connector and any port connection
Power conductors <i>U+</i> and <i>U-</i>	180 mm maximum length measured between the trunk line tap connector and any port connection. Maximum resistance 10 mΩ for conductors rated up to 3 A, 4 mΩ otherwise, both measured between the trunk line tap connector and any port connection
Signal conductors CAN_H and CAN_L	180 mm maximum length and 10 mΩ maximum resistance, both measured between the trunk line tap connector and any port connection

8.5.2 Power taps

A power tap connects a power supply to the trunk line. When connected to the link, a power tap shall provide continuous connection for the signal, drain and *U-* wires. A power tap may also provide:

- overcurrent protection in each direction from the tap;
- power supply protection when multiple power supplies are used;
- a connection to the shield/drain wire for grounding the link.

DeviceNet power taps shall have a minimum voltage rating of 25 V d.c. and shall comply with the specifications given in table 62 and table 63.

Table 62 – Internal pass-through conductor specifications

Conductor description	Specification
Drain wire conductor	180 mm maximum length of conductor Maximum resistance 4,1 mΩ
<i>U-</i> power conductor	180 mm maximum length of conductor Maximum resistance 2,1 mΩ. Rated current 8 A
Signal conductors CAN_H and CAN_L	180 mm maximum length of conductor Maximum resistance 4,1 mΩ

Tableau 63 – Spécifications des conducteurs de dérivation internes

Description du conducteur	Spécifications
Fil conducteur de décharge vers la borne de terre	Longueur maximale de 180 mm et résistance maximale de 2,1 mΩ, mesurées entre n'importe quel connecteur du raccord d'alimentation et la borne de terre
Conducteurs d'alimentation $U+$ et $U-$	Longueur maximale de 300 mm et résistance maximale de 3,6 mΩ, mesurées entre n'importe quel connecteur du raccord d'alimentation et le connecteur du bloc d'alimentation. La valeur nominale des fils doit permettre le transport du courant de fonctionnement

Les connecteurs des raccords d'alimentation (sauf le connecteur du bloc d'alimentation) doivent se conformer aux exigences données en 4.4.

4.6 Appareils alimentés par la liaison

Les appareils peuvent être alimentés par la liaison. Ils doivent être munis d'un régulateur de tension, d'une tension d'entrée comprise entre 11 V c.c. et 25 V c.c., et comprendre une protection contre les transitoires et un système de filtrage.

Le régulateur de tension doit répondre aux spécifications du tableau 64.

Tableau 64 – Spécifications du régulateur de tension

Spécifications	Paramètre										
Plage de tensions d'entrée	11 V c.c. à 25 V c.c.										
Isolation (si requise)	500 V										
Délai de mise sous tension	Régulateurs linéaires: aucun Régulateurs en mode commutateur: <table border="0"> <tr> <td><100 mA</td> <td>2 ms à 10 ms</td> </tr> <tr> <td>0,1 A à 0,5 A</td> <td>5 ms à 15 ms</td> </tr> <tr> <td>0,5 A à 1 A</td> <td>10 ms à 20 ms</td> </tr> <tr> <td>1 A à 2 A</td> <td>15 ms à 30 ms</td> </tr> <tr> <td>>2 A</td> <td>20 ms à 40 ms</td> </tr> </table>	<100 mA	2 ms à 10 ms	0,1 A à 0,5 A	5 ms à 15 ms	0,5 A à 1 A	10 ms à 20 ms	1 A à 2 A	15 ms à 30 ms	>2 A	20 ms à 40 ms
<100 mA	2 ms à 10 ms										
0,1 A à 0,5 A	5 ms à 15 ms										
0,5 A à 1 A	10 ms à 20 ms										
1 A à 2 A	15 ms à 30 ms										
>2 A	20 ms à 40 ms										
Protection des sorties contre les courts-circuits	Limite de courant										
Protection contre l'inversion de polarité	Redresseur du trajet à la terre ¹⁾										
¹⁾ Si le redresseur est utilisé pour alimenter le transcepteur, il doit avoir une tension directe de 0,6 V au maximum.											

4.7 Protection contre les erreurs de câblage

Les appareils doivent pouvoir résister à une mauvaise connexion sans subir de dommages, dans les conditions suivantes:

- inversion des conducteurs $U+$ et $U-$;
- connexion du fil de décharge à n'importe quel autre conducteur;
- connexion de CAN_H et/ou CAN_L à une tension ne dépassant pas la plage –25 V c.c. à +18 V c.c.

Voir 5.2.4 et 5.2.5 pour les spécifications d'essais.

Table 63 – Internal power drop conductor specifications

Conductor description	Specification
Drain wire conductor to ground terminal	180 mm maximum length and 2,1 mΩ maximum resistance, both measured between any tap connector and the ground terminal
Power conductors $U+$ and $U-$	300 mm maximum length and 3,6 mΩ maximum resistance, both measured between any tap connector and the power supply port connection. Wires shall be rated to carry the operating current

Power tap connectors (except the power supply connector) shall meet the requirements given in 8.4.

8.6 Link powered devices

Devices may draw power from the link. Such devices shall use a voltage regulator with an input voltage between 11 V d.c. and 25 V d.c., and incorporate transient protection and filtering.

The voltage regulator shall comply with the specifications given in table 64.

Table 64 – Voltage regulator specifications

Specification	Parameter										
Input voltage range	11 V d.c. to 25 V d.c.										
Isolation (if required)	500 V										
Power ON delay	Linear regulators: none Switch mode regulators: <table border="0"> <tr> <td><100 mA</td><td>2 ms to 10 ms</td></tr> <tr> <td>0,1 A to 0,5 A</td><td>5 ms to 15 ms</td></tr> <tr> <td>0,5 A to 1 A</td><td>10 ms to 20 ms</td></tr> <tr> <td>1 A to 2 A</td><td>15 ms to 30 ms</td></tr> <tr> <td>>2 A</td><td>20 ms to 40 ms</td></tr> </table>	<100 mA	2 ms to 10 ms	0,1 A to 0,5 A	5 ms to 15 ms	0,5 A to 1 A	10 ms to 20 ms	1 A to 2 A	15 ms to 30 ms	>2 A	20 ms to 40 ms
<100 mA	2 ms to 10 ms										
0,1 A to 0,5 A	5 ms to 15 ms										
0,5 A to 1 A	10 ms to 20 ms										
1 A to 2 A	15 ms to 30 ms										
>2 A	20 ms to 40 ms										
Output short-circuit protection	Current limit										
Reverse polarity protection	Rectifier in ground path ¹⁾										
¹⁾ If the rectifier is used to power the transceiver, it shall have a forward voltage no greater than 0,6 V.											

8.7 Miswiring protection

Devices shall be capable of sustaining misconnection without suffering damage, under the following conditions:

- $U+$ and $U-$ conductors interchanged;
- connection of the drain wire to any other conductor;
- connection of CAN_H and/or CAN_L to a voltage not exceeding the range –25 V d.c. to +18 V d.c.

See 9.2.4 and 9.2.5 for test specifications.

4.8 Blocs d'alimentation

Les blocs d'alimentation connectés à la liaison doivent répondre aux spécifications données dans le tableau 65 (voir 5.2.1 pour les spécifications d'essais).

Tableau 65 – Spécifications des blocs d'alimentation DeviceNet

Spécifications	Paramètre
Tension nominale à +20 °C	24 V c.c. ± 1 % ou réglable à 0,2 %
Régulation de la ligne	0,3 % maximum sur la plage spécifiée par le fabricant
Régulation de la charge	0,3 % maximum
Coefficient de température	0,03 % par K maximum
Ondulation de sortie	250 V crête à crête
Capacité de charge	7 000 μ F maximum
Protection contre les surintensités	100 % à 125 % du courant nominal
Temps de montée en tension de sortie (à pleine charge)	250 ms maximum à 95 % de la valeur finale
Dépassement d'alimentation	0,2 % maximum
Stabilité	0 à 100 % de la charge (toutes conditions)
Isolation	Sortie isolée de la tension d'entrée et de la mise à la terre du châssis

4.9 Compatibilité électromagnétique (CEM)

4.9.1 Généralités

Tous les essais d'immunité et d'émissions sont des essais de type qui doivent être conduits dans des conditions représentatives, sur le plan opérationnel et sur le plan environnemental, en utilisant les pratiques de câblage recommandées et en incluant tout l'équipement nécessaire à la communication et au transfert des données sur le câble DeviceNet (voir 5.2.10 pour les spécifications d'essais).

Ces exigences doivent être satisfaites si on utilise deux appareils échangeant des communications d'E/S et un bloc d'alimentation.

4.9.2 Immunité

4.9.2.1 Critères de comportement

Les résultats d'essais sont spécifiés à l'aide des critères de comportement suivants:

- **critère A:** aucune dégradation du comportement au-delà des limites spécifiées n'est autorisée lorsque l'appareil est utilisé comme prévu (voir tableau 66);
- **critère B:** aucun changement de l'état de fonctionnement réel ou des données stockées n'est autorisé. En cours d'essai, une dégradation du comportement est admise. Aucune dégradation du comportement ou perte de fonction au-delà des limites spécifiées n'est autorisée après l'essai lorsque l'appareil est utilisé comme prévu (voir tableau 66).

8.8 Power supplies

Power supplies which are connected to the link shall comply with the specifications given in table 65 (see 9.2.1 for test specifications).

Table 65 – DeviceNet power supply specifications

Specification	Parameter
Nominal voltage at +20 °C	24 V d.c. ± 1 % or adjustable to 0,2 %
Line regulation	0,3 % max. over the range specified by the manufacturer
Load regulation	0,3 % max.
Temperature coefficient	0,03 % per K max.
Output ripple	250 mV peak to peak
Load capacitance capability	7 000 μ F max.
Overcurrent protection	100 % to 125 % of rated current
Output voltage rise time (with full load)	250 ms max. to 95 % of final value
Power ON overshoot	0,2 % max.
Stability	0 to 100 % load (all conditions)
Isolation	Output isolated from input voltage and chassis ground

8.9 Electromagnetic compatibility (EMC)

8.9.1 General

All immunity and emission tests are type tests and shall be carried out under representative conditions, both operational and environmental, using the recommended wiring practices and including all equipment necessary for communication and data transfer on the DeviceNet cable (see 9.2.10 for test specifications).

This requirement shall be met by the use of two devices with ongoing I/O communication and one power supply.

8.9.2 Immunity

8.9.2.1 Performance criteria

The test results are specified using the following performance criteria:

- **criterion A:** no degradation of performance beyond specified limits is allowed when the apparatus is used as intended, see table 66;
- **criterion B:** no change of actual operating state or stored data is allowed. During the test, degradation of performance is allowed. No degradation of performance or loss of function beyond specified limits is allowed after the test when the apparatus is used as intended, see table 66.

Tableau 66 – Critères de comportement pour les essais d'immunité

Type de fonction	Critère A	Critère B
Maîtres (processeurs et adaptateurs)	Perte de programme Défauts de mémoire Remise à zéro des E/S Corruption de la table de données	Perte de programme Défauts de mémoire Remise à zéro des E/S Corruption de la table de données
Tout module	Fonctionnement inattendu Blocage Intervention de l'opérateur Dégât matériel	Fonctionnement inattendu Blocage Intervention de l'opérateur Dégât matériel
Communications externes	Station hors ligne	Station hors ligne
Communications internes: – Rayonnées, conduites – Transitoire rapide – Décharge électrostatique – Surtensions	> 1 bit d'erreur mis à 1 pour 10 transferts	> 1 bit d'erreur mis à 1 pour 10 transferts Blocage Blocage

4.9.2.2 Immunité aux décharges électrostatiques (ESD)

Conformément à la CEI 61000-4-2, 10 décharges de chaque polarité doivent être appliquées comme suit:

- une tension d'essai de ± 8 kV doit être appliquée par la méthode de décharge à distance aux boîtiers non métalliques des appareils;
- une tension d'essai de ± 4 kV doit être appliquée par la méthode de décharge par contact aux boîtiers métalliques des appareils.

Le critère de comportement B est applicable (voir 4.9.2.1).

4.9.2.3 Immunité aux champs électromagnétiques rayonnés aux fréquences radioélectriques

Conformément à la CEI 61000-4-3:

- 10 V/m dans la plage de fréquences de 80 MHz à 1 000 MHz avec une modulation d'amplitude.

Le critère de comportement A est applicable (voir 4.9.2.1).

4.9.2.4 Immunité aux transitoires électriques rapides en sèves

Conformément à la CEI 61000-4-4:

- une tension d'essai maximale de ± 1 kV à 5 kHz doit être appliquée aux câbles qui contiennent le support de communication CDI;
- une tension d'essai maximale de ± 2 kV à 5 kHz doit être appliquée à tous les autres câbles et accès.

Le critère de comportement B est applicable (voir 4.9.2.1).

Table 66 – Immunity performance criteria

Function type	Criterion A	Criterion B
Masters (processors and adapters)	Program loss Memory faults I/O reset Data table corruption	Program loss Memory faults I/O reset Data table corruption
Any module	Unexpected operation Lockup Operator intervention Damage	Unexpected operation Lockup Operator intervention Damage
External communications	Node off-line	Node off-line
Internal communications: – Radiated, conducted – Fast transient – ESD – Surge	> 1 error bit set / 10 transfers	> 1 error bit set / 10 transfers Lockup Lockup

8.9.2.2 Electrostatic discharge (ESD) immunity

In accordance with IEC 61000-4-2, 10 discharges of each polarity shall be applied as follows:

- a test voltage of ± 8 kV shall be applied by the air gap discharge method to non-metallic device enclosures;
- a test voltage of ± 4 kV shall be applied by the contact discharge method to metallic device enclosures.

Performance criterion B applies (see 8.9.2.1).

8.9.2.3 Radiated radio-frequency electromagnetic field immunity

In accordance with IEC 61000-4-3:

- 10 V/m over the frequency range 80 MHz to 1 000 MHz, amplitude modulated.

Performance criterion A applies (see 8.9.2.1).

8.9.2.4 Electrical fast transient/burst immunity

In accordance with IEC 61000-4-4:

- a maximum test voltage of ± 1 kV at 5 kHz shall be applied to cables that contain the CDI communication medium;
- a maximum test voltage of ± 2 kV at 5 kHz shall be applied to all other cables and ports.

Performance criterion B applies (see 8.9.2.1).

4.9.2.5 Immunité aux ondes de choc

Conformément à la CEI 61000-4-5:

- cinq ondes de choc d'un maximum de ± 2 kV doivent être appliquées entre la ligne d'alimentation c.a. et la terre;
- cinq ondes de choc d'un maximum de ± 1 kV doivent être appliquées entre les lignes d'alimentation c.a.

Le critère de comportement B est applicable (voir 4.9.2.1).

4.9.2.6 Immunité aux perturbations conduites aux fréquences radioélectriques

Conformément à la CEI 61000-4-6:

- 10 V efficaces dans la plage de fréquences de 150 kHz à 80 MHz avec une modulation d'amplitude.

Le critère de comportement A est applicable (voir 4.9.2.1).

4.9.2.7 Creux de tension et interruptions

La conformité du bloc d'alimentation aux limites spécifiées, indiquées dans le tableau 65, permet de se passer de tout essai supplémentaire.

4.9.3 Emissions

4.9.3.1 Emissions rayonnées

Conformément à la CISPR 11, groupe 1, classe A.

4.9.3.2 Emissions conduites

Conformément à la CISPR 11, groupe 1, classe A.

5 Essais

5.1 Généralités

Le présent article spécifie les essais de type pour les exigences électriques et logiques. Les essais sont divisés en deux parties:

- essais électriques et de compatibilité électromagnétique (CEM);
- essais logiques (comportement d'un appareil sur DeviceNet).

NOTE Si les spécifications de DeviceNet font partie intégrante de la conception d'un produit, certains essais logiques peuvent dépendre du comportement du produit lui-même. La conformité à la présente partie de la norme ne signifie pas nécessairement la conformité à une norme spécifique relative au produit dans lequel DeviceNet se trouve intégré. La présente partie de la norme n'a pas pour objet d'effectuer des essais d'applications spécifiques utilisant les fonctions DeviceNet.

Les essais applicables à un appareil en essai (EUT) spécifique doivent être effectués dans l'ordre indiqué.

8.9.2.5 Surge immunity

In accordance with IEC 61000-4-5:

- five surges at a maximum of ± 2 kV shall be applied between a.c. mains line and earth;
- five surges at a maximum of ± 1 kV shall be applied between a.c. mains lines.

Performance criterion B applies (see 8.9.2.1).

8.9.2.6 Conducted radio-frequency disturbance immunity

In accordance with IEC 61000-4-6:

- 10 V r.m.s. over the frequency range 150 kHz to 80 MHz, amplitude modulated

Performance criterion A applies (see 8.9.2.1).

8.9.2.7 Voltage dips and interruptions

Conformity of the power supply to the specified limits given in table 65 eliminates the need for any further testing.

8.9.3 Emissions

8.9.3.1 Radiated emissions

In accordance with CISPR 11, group 1, class A.

8.9.3.2 Conducted emissions

In accordance with CISPR 11, group 1, class A.

9 Tests

9.1 General

This clause specifies type tests for electrical and logical requirements. Tests are divided into two parts:

- electrical and EMC testing;
- logical testing (behaviour of a device on DeviceNet).

NOTE If DeviceNet specifications are implemented as an integral part of a product, some logical tests may depend on the behaviour of the product itself. Conformance to this part of the standard does not necessarily mean conformance to a specific standard of the whole product in which DeviceNet is integrated. A test of specific applications of DeviceNet functions is outside the scope of this part of the standard.

The tests applicable to a specific EUT shall be performed in the sequence indicated.

5.2 Essais électriques et de compatibilité électromagnétique (CEM)

5.2.1 Essai du bloc d'alimentation DeviceNet

5.2.1.1 Objet de l'essai

Les essais suivants vérifient les exigences relatives au bloc d'alimentation qui sont spécifiques à DeviceNet.

5.2.1.2 Temps de montée en tension de sortie du bloc d'alimentation

5.2.1.2.1 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 40. Sélectionner la résistance de charge (R) pour vérifier que le bloc d'alimentation fonctionne au courant nominal.

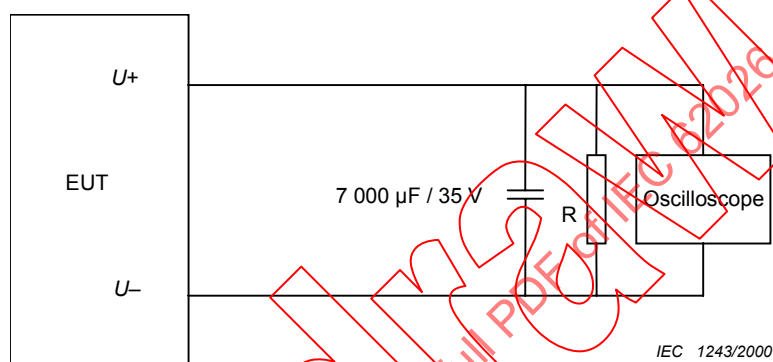


Figure 40 – Circuit d'essai du temps de montée du bloc d'alimentation

5.2.1.2.2 Procédure d'essai

L'oscilloscope connecté à travers la résistance de charge doit être réglé à 24 V, 500 ms pleine échelle. L'appareil en essai (EUT) doit être mis en route et le temps de montée, du départ jusqu'à 95 % de la tension finale d'alimentation, doit être enregistré.

5.2.1.2.3 Critères de conformité

Le temps de montée doit être inférieur ou égal à 250 ms (voir tableau 65).

5.2.1.3 Ondulation du bloc d'alimentation

5.2.1.3.1 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 40, avec le condensateur déconnecté.

5.2.1.3.2 Procédure d'essai

Le bloc d'alimentation doit être mis en route et l'ondulation c.a. crête à crête de la tension de sortie du bloc d'alimentation mesurée avec l'oscilloscope.

5.2.1.3.3 Critères de conformité

L'ondulation crête à crête doit être inférieure ou égale à 250 mV pour satisfaire aux exigences relatives au bloc d'alimentation (voir tableau 65).

9.2 Electrical and EMC testing

9.2.1 Test of the DeviceNet power supply

9.2.1.1 Test purpose

The following tests verify the power supply requirements that are specific to DeviceNet.

9.2.1.2 Power supply output voltage rise time

9.2.1.2.1 Test circuit

The test circuit shall be as shown in figure 40. Select the load resistor (R) to ensure that the power supply operates at rated current.

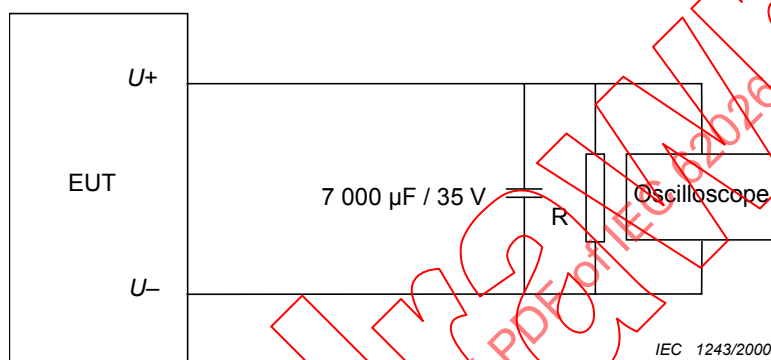


Figure 40 – Power supply rise time test circuit

9.2.1.2.2 Test procedure

The oscilloscope connected across the load resistor shall be set to 24 V, 500 ms full scale. The EUT shall be switched on and the rise time from the start of the voltage rise to 95 % of the power supply final voltage shall be recorded.

9.2.1.2.3 Criteria for compliance

The rise time shall be less than or equal to 250 ms (see table 65).

9.2.1.3 Power supply ripple

9.2.1.3.1 Test circuit

The test circuit shall be as shown in figure 40, with the capacitor disconnected.

9.2.1.3.2 Test procedure

The power supply shall be switched on and the a.c. peak-to-peak ripple of the power supply output voltage measured using the oscilloscope.

9.2.1.3.3 Criteria for compliance

In order to meet the power supply requirements, the peak-to-peak ripple shall be less than or equal to 250 mV (see table 65).

5.2.2 Consommation maximale de courant

5.2.2.1 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 41.

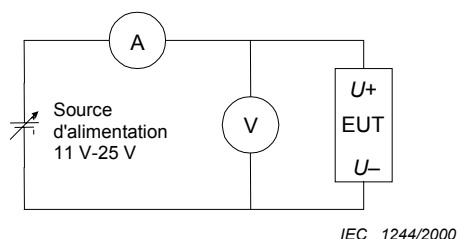


Figure 41 – Circuit d'essai de consommation de courant

5.2.2.2 Procédure d'essai

Déterminer l'état de fonctionnement qui aboutit à la consommation maximale de courant tiré de la liaison. L'appareil en essai (EUT) doit être utilisé dans cet état. Si des fluctuations de courant se produisent, le courant maximal prélevé sur la liaison doit être enregistré.

Les essais doivent être effectués aux limites supérieure et inférieure de la plage de tensions spécifiée en 1.6.7.2.

5.2.2.3 Critères de conformité

Déterminer la limite de courant applicable en inspectant l'appareil et/ou en consultant la documentation du fabricant. Les relevés de courant obtenus doivent être conformes à 1.6.7.2 et être inférieurs ou égaux à 8 A.

5.2.3 Comportement à la mise sous tension

5.2.3.1 Objet de l'essai

Cet essai a pour objet de déterminer que le premier signal transmis est un message valide de duplication d'adresses MAC ID dans les conditions suivantes:

- quand l'appareil est ajouté à un CDI existant en cours d'activité;
- quand le CDI est mis hors tension puis sous tension.

5.2.3.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 42.

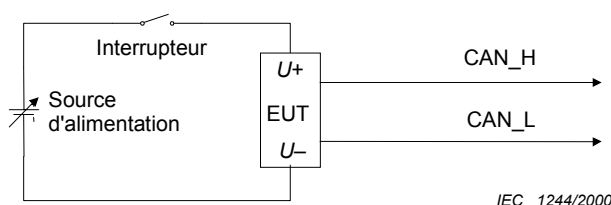


Figure 42 – Circuit d'essai de mise sous tension

9.2.2 Device peak current consumption

9.2.2.1 Test circuit

The test circuit shall be as shown in figure 41.

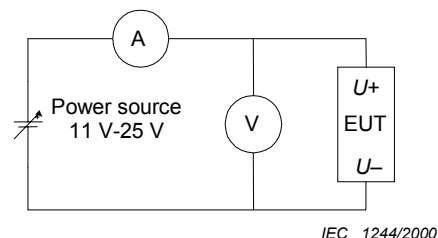


Figure 41 – Current consumption test circuit

9.2.2.2 Test procedure

To determine the operating state that results in maximum current consumption from the link. The EUT shall be operated in this state. If current fluctuations occur the peak current drawn from the link shall be recorded.

The tests shall be performed at the upper and lower limits of the voltage range specified in 5.6.7.2.

9.2.2.3 Criteria for compliance

To determine the applicable current limit by inspection of the device and/or the manufacturer's documentation. The current readings obtained shall be in accordance with 5.6.7.2 and shall be less than or equal to 8 A.

9.2.3 Power ON behaviour

9.2.3.1 Test purpose

The purpose of this test is to determine that the first transmitted signal is a valid duplicate MAC ID message under the following conditions:

- when added to an existing working CDI;
- when the power is cycled on the CDI.

9.2.3.2 Test circuit

The test circuit shall be as shown in figure 42.

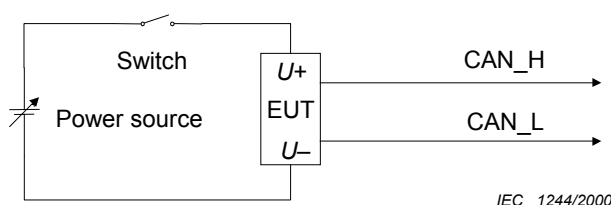


Figure 42 – Power ON test circuit

5.2.3.3 Procédure d'essai

Le commutateur d'alimentation doit être fermé et les lignes de signaux CAN_H et CAN_L doivent être surveillées.

5.2.3.4 Critères de conformité

Le premier signal transmis doit être une demande valide de vérification de duplication d'adresses MAC ID.

5.2.4 Inversion des connexions $U+$ et $U-$

5.2.4.1 Objet de l'essai

Cet essai a pour objet de vérifier que les exigences concernant la protection contre les erreurs de câblage sont satisfaites.

5.2.4.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 43.

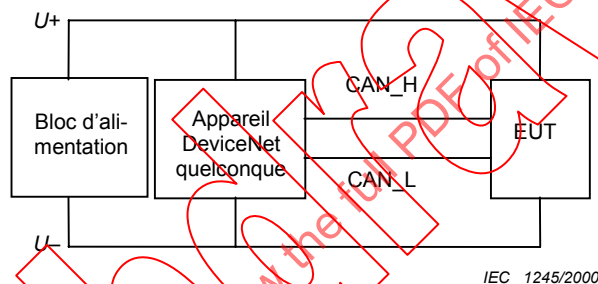


Figure 43 – Circuit d'essai pour l'inversion de $U+$ et $U-$, et aussi pour la coupure de $U-$

5.2.4.3 Procédure d'essai

La borne $U+$ de l'appareil en essai (EUT) doit être connectée au $U-$ et la borne $U-$ de l'appareil en essai au $U+$ du bloc d'alimentation et de la liaison. CAN_H et CAN_L doivent être surveillés.

5.2.4.4 Critères de conformité

Il faut vérifier que les signaux CAN_H et CAN_L restent dans les limites spécifiées dans le tableau 42 et le tableau 43.

5.2.5 Déconnexion de $U-$

5.2.5.1 Objet de l'essai

Cet essai a pour objet de vérifier que les exigences concernant la protection contre les erreurs de câblage sont satisfaites.

5.2.5.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 43.

9.2.3.3 Test procedure

The switch shall be closed and the CAN_H and CAN_L signal lines shall be monitored.

9.2.3.4 Criteria for compliance

The first transmitted signal shall be a valid duplicate MAC ID request.

9.2.4 Reverse connection of $U+$ and $U-$

9.2.4.1 Test purpose

The purpose of this test is to verify compliance with the requirements for miswiring protection.

9.2.4.2 Test circuit

The test circuit shall be as shown in figure 43.

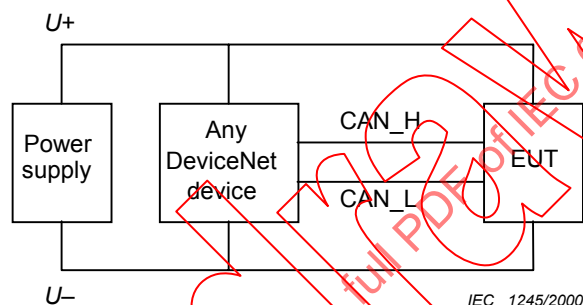


Figure 43 – Test circuit for reversal of $U+$ and $U-$, and also $U-$ interruption

9.2.4.3 Test procedure

The $U+$ terminal of the EUT shall be connected to the $U-$ of the power supply and link, and the $U-$ terminal of the EUT shall be connected to the $U+$ of the power supply and link. CAN_H and CAN_L shall be monitored.

9.2.4.4 Criteria for compliance

It shall be verified that the CAN_H and CAN_L signals remain within the limits specified in table 42 and table 43.

9.2.5 Disconnection of $U-$

9.2.5.1 Test purpose

The purpose of this test is to verify compliance with the requirements for miswiring protection.

9.2.5.2 Test circuit

The test circuit shall be as shown in figure 43.

5.2.5.3 Procédure d'essai

La connexion U_- doit être déconnectée de l'appareil en essai (EUT). CAN_H et CAN_L doivent être surveillés.

5.2.5.4 Critères de conformité

Il faut vérifier que les signaux CAN_H et CAN_L restent dans les limites spécifiées dans le tableau 42 et le tableau 43.

5.2.6 Essai d'impédance d'entrée différentielle

5.2.6.1 Objet de l'essai

Cet essai a pour objet de vérifier que l'impédance d'entrée de l'appareil en essai (EUT) satisfait aux exigences de la couche physique.

5.2.6.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 44.

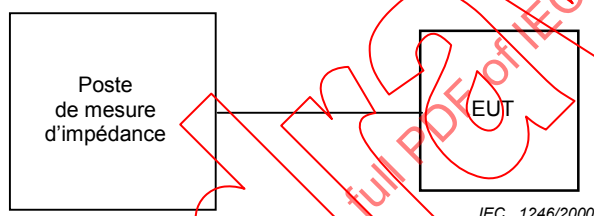


Figure 44 – Circuit d'essai d'impédance différentielle

5.2.6.3 Procédure d'essai

L'appareil en essai (EUT) doit être mis sous tension. Le transcepteur de l'appareil en essai doit être désactivé de façon à ne pas affecter l'impédance. L'impédance d'entrée de l'appareil en essai entre CAN_H et CAN_L doit être mesurée à la fréquence de $100 \text{ kHz} \pm 1 \text{ kHz}$.

5.2.6.4 Critères de conformité

L'impédance d'entrée de l'appareil en essai (EUT) entre CAN_H et CAN_L ne doit pas dépasser les valeurs spécifiées dans le tableau 41.

5.2.7 Niveaux de transmission

5.2.7.1 Objet de l'essai

Cet essai a pour objet de déterminer si les valeurs de CAN_H et de CAN_L relativement à U_- , dans les conditions de simulation d'impédance les plus défavorables, se trouvent dans les limites spécifiées au sein des exigences de la couche physique.

5.2.7.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 45.

9.2.5.3 Test procedure

The $U-$ connection shall be disconnected at the EUT. CAN_H and CAN_L shall be monitored.

9.2.5.4 Criteria for compliance

It shall be verified that the CAN_H and CAN_L signals remain within the limits specified in table 42 and table 43.

9.2.6 Differential input impedance test

9.2.6.1 Test purpose

The purpose of this test is to verify that the input impedance of the EUT meets the requirements of the physical layer.

9.2.6.2 Test circuit

The test circuit shall be as shown in figure 44.

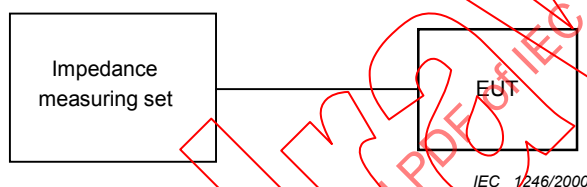


Figure 44 – Differential impedance test circuit

9.2.6.3 Test procedure

Power shall be applied to the EUT. The transceiver of the EUT shall be disabled in such a way that the impedance is not affected. The input impedance of the EUT between CAN_H and CAN_L shall then be measured at a frequency of $100\text{ kHz} \pm 1\text{ kHz}$.

9.2.6.4 Criteria for compliance

The input impedance of the EUT between CAN_H and CAN_L shall not exceed the values specified in table 41.

9.2.7 Transmit levels

9.2.7.1 Test purpose

The purpose of this test is to determine whether the values of CAN_H and CAN_L with respect to $U-$ under simulated worst case impedance conditions are within the limits specified as part of the physical layer requirements.

9.2.7.2 Test circuit

The test circuit shall be as shown in figure 45.

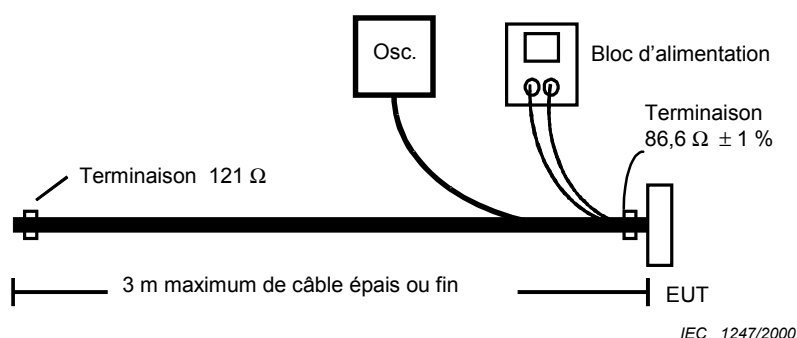


Figure 45 – Installation pour l'essai des niveaux de transmission

5.2.7.3 Procédure d'essai

Pour cet essai, l'appareil en essai (EUT) doit effectuer une transmission. En l'absence de tout autre appareil sur la liaison, l'appareil en essai commence immédiatement à envoyer des messages de demande de vérification de duplication d'adresses MAC ID dès que la séquence de mise sous tension est terminée. Le contenu réel du message provenant de l'appareil en essai doit être observé sur l'oscilloscope. Les valeurs suivantes doivent être enregistrées (voir figure 46):

- tension CAN_L récessive relativement à U_- ;
- tension CAN_H récessive relativement à U_- ;
- tension CAN_L dominante relativement à U_- ;
- tension CAN_H dominante relativement à U_- ;
- niveau de sortie différentiel.

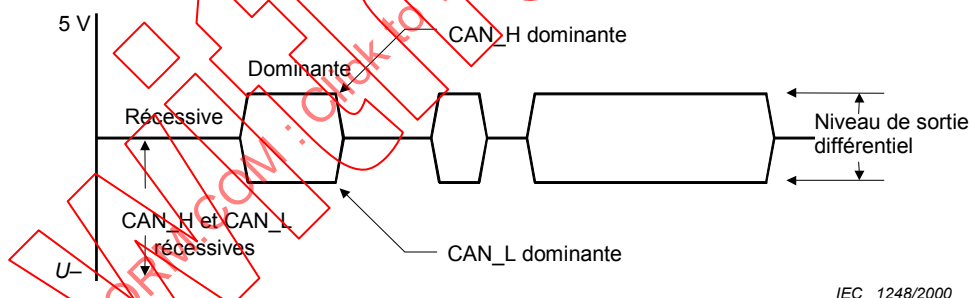


Figure 46 – Niveaux de transmission

5.2.7.4 Critères de conformité

Les valeurs doivent correspondre à celles spécifiées dans le tableau 42.

5.2.8 Délai d'acquiescement

5.2.8.1 Objet de l'essai

Cet essai a pour objet de vérifier que le délai d'acquiescement reste à l'intérieur des limites spécifiées au sein des exigences de la couche physique.

5.2.8.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 47.

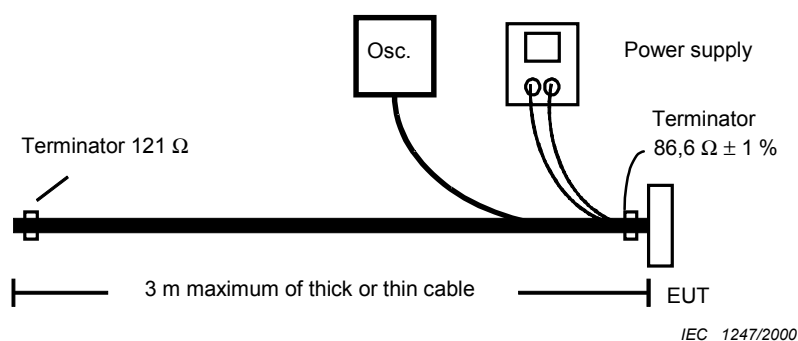


Figure 45 – Transmit level test set-up

9.2.7.3 Test procedure

For this test, the EUT shall be transmitting. With no other device on the link, the EUT will automatically start sending duplicate MAC ID request messages after completing its power-up sequence. The actual message content from the EUT shall be observed on the oscilloscope. The following values shall be recorded (see figure 46):

- recessive CAN_L voltage with respect to U_- ;
- recessive CAN_H voltage with respect to U_- ;
- dominant CAN_L voltage with respect to U_- ;
- dominant CAN_H voltage with respect to U_- ;
- differential output level.

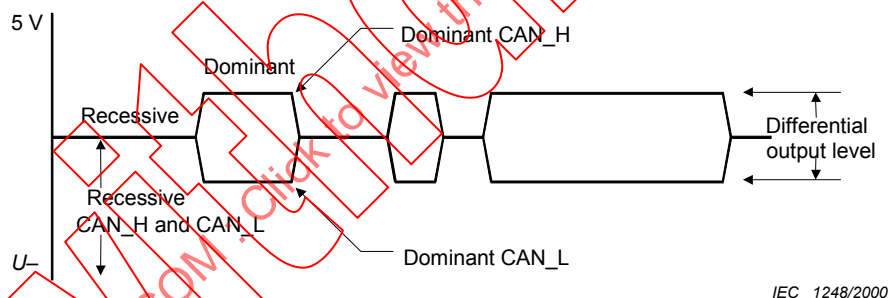


Figure 46 – Transmit levels

9.2.7.4 Criteria for compliance

The values shall be in accordance with those specified in table 42.

9.2.8 Acknowledgement delay

9.2.8.1 Test purpose

The purpose of this test is to check that the acknowledgement delay is within the limits specified as part of the physical layer requirements.

9.2.8.2 Test circuit

The test circuit shall be as shown in figure 47.

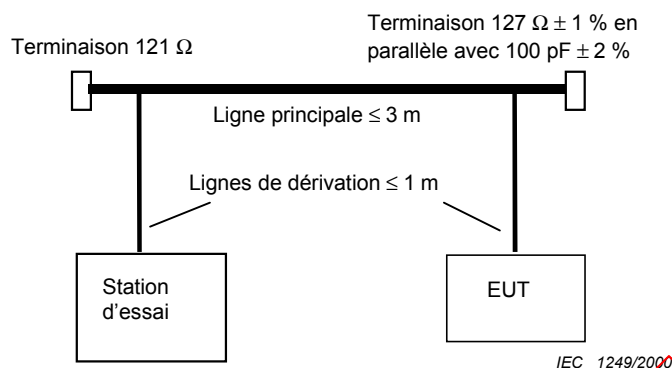


Figure 47 – Configuration de l'essai de temporisation

5.2.8.3 Procédure d'essai

L'appareil en essai (EUT) et la station d'essai doivent être réglés à la vitesse de transmission la plus haute supportée par l'appareil en essai. La station d'essai doit envoyer de façon répétée une demande `get_attribute_single` à l'appareil en essai, sans ouvrir de connexion de messagerie. Cela garantit que l'appareil en essai n'envoie pas d'autres messages que le bit d'acquittement. Un oscilloscope doit être utilisé pour mesurer le temps de réponse de l'acquittement au niveau de l'appareil en essai. Il s'agit du temps compris entre le front descendant du dernier bit envoyé par la station d'essai et le front montant du bit d'acquittement envoyé par l'appareil en essai (voir figure 48).

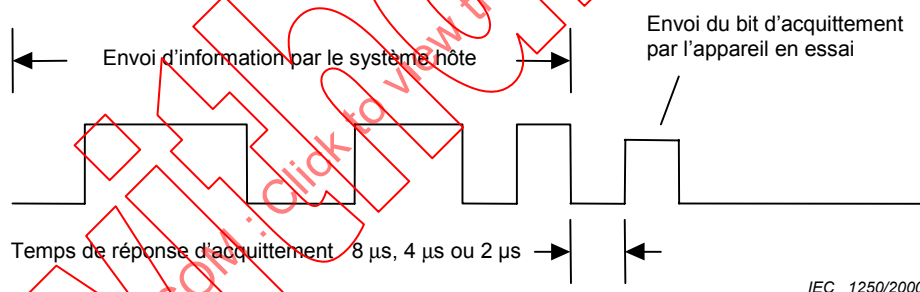


Figure 48 – Temporisation

5.2.8.4 Critères de conformité

Les valeurs suivantes doivent être soustraites des temps de réponse mesurés, en fonction de la vitesse de transmission utilisée:

125 kbit/s	8 µs
250 kbit/s	4 µs
500 kbit/s	2 µs

Le délai d'acquittement de l'appareil doit être inférieur ou égal à la valeur spécifiée en 1.6.2.

5.2.9 Essai CDI

5.2.9.1 Objet de l'essai

Cet essai a pour objet de déterminer si, dans les conditions les plus défavorables, les bits d'acquittement manquants et les conditions bus-off restent dans les limites spécifiées dans la documentation du fabricant.

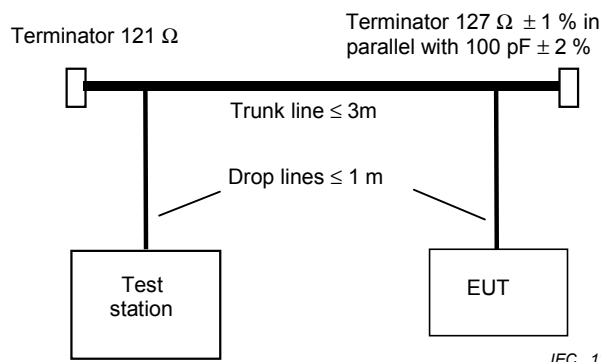


Figure 47 – Timing test set-up

9.2.8.3 Test procedure

The EUT and the test station shall be set to the highest bit rate supported by the EUT. The test station shall repetitively send a `get_attribute_single` request to the EUT without opening a messaging connection. This ensures that the EUT sends no messages other than the acknowledgement bit. An oscilloscope shall be used to measure the acknowledgement response time at the EUT. This is the time from the falling edge of the last bit sent by the test station to the rising edge of the acknowledgement bit sent by the EUT (see figure 48).

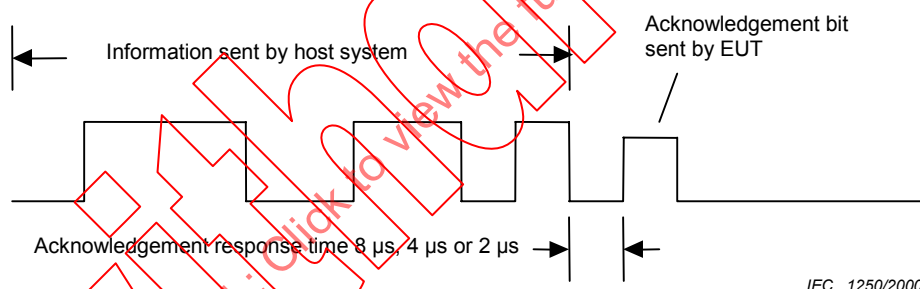


Figure 48 – Timing

9.2.8.4 Criteria for compliance

Depending on the bit rate used, the following values shall be subtracted from the response times measured:

125 kbit/s	8 μs
250 kbit/s	4 μs
500 kbit/s	2 μs

The resultant device acknowledgement delay shall be less than or equal to the value specified in 5.6.2.

9.2.9 CDI test

9.2.9.1 Test purpose

The purpose of this test is to determine whether the missing acknowledgement bits and bus-off conditions under worst case conditions are within the limits specified within the manufacturer's documentation.

5.2.9.2 Circuit d'essai

Le circuit d'essai doit correspondre aux indications de la figure 49.

- le système doit fonctionner à la vitesse de transmission maximale supportée par l'appareil en essai (EUT);
- le système de câblage doit être construit avec une longueur de câble épais correspondant à la longueur maximale de la ligne principale pour cette vitesse de transmission;
- le cumul maximal de lignes de dérivation doit être construit avec une longueur de câble fin correspondant à la longueur maximale des lignes de dérivation pour cette vitesse de transmission;

- 62 appareils ou 62 charges équivalentes doivent être utilisés pour charger le CDI près de l'appareil en essai (EUT);

NOTE Une charge équivalente correspond à l'impédance d'entrée différentielle minimale définie dans le tableau 41.

- le bloc d'alimentation utilisé doit répondre aux spécifications décrites dans le tableau 65 (capacité minimale de courant de 4 A);
- la charge d'alimentation liaison la plus défavorable doit être connectée près de l'appareil en essai (EUT). Cette charge doit provoquer, pour U_{-} , une chute de $5_{-0,1}^0$ V entre le bloc d'alimentation et l'appareil en essai;
- la station d'essai utilisée doit pouvoir à la fois produire un trafic de communications valides pour l'appareil en essai (EUT) et surveiller tout le trafic de communications.

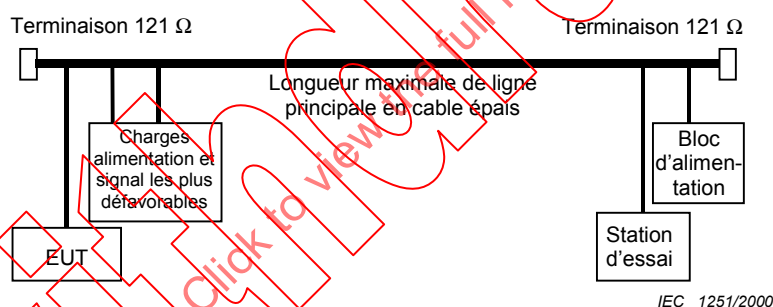


Figure 49 – Configuration de l'essai CDI

5.2.9.3 Procédure d'essai

Des communications doivent être établies entre la station d'essai et l'appareil en essai (EUT). Tout bit d'acquiescement manquant ou toute condition bus-off doit être enregistré.

5.2.9.4 Critères de conformité

Les bits d'acquiescements manquants et conditions bus-off enregistrés doivent correspondre aux spécifications données par le fabricant de l'appareil en essai (EUT).

5.2.10 Essais de compatibilité électromagnétique

5.2.10.1 Généralités

Le circuit d'essai doit être tel qu'indiqué à la figure 49:

- le système doit fonctionner à la vitesse de transmission maximale supportée par l'appareil en essai (EUT);
- la ligne principale doit être réalisée avec du câble épais;

9.2.9.2 Test circuit

The test circuit shall be as shown in figure 49:

- the system shall be run at the maximum bit rate supported by the EUT;
- the cable system shall be constructed of the maximum trunk line length of thick cable suitable for this bit rate;
- the maximum cumulative drop line shall be constructed of the maximum thin cable drop line length suitable for this bit rate;
- 62 devices or 62 device equivalents shall be used to load the CDI near the EUT;
NOTE A device equivalent is the minimum differential input impedance as defined in table 41.
- a power supply shall be used that meets the specifications described in table 65 (minimum current capacity 4 A);
- a worst case link power load shall be connected adjacent to the EUT. This load shall cause the U -line to develop a $5_{-0,1}^0$ V drop from the power supply to the EUT;
- a test station shall be used that is capable of both producing valid communication traffic to the EUT and monitoring all communication traffic.

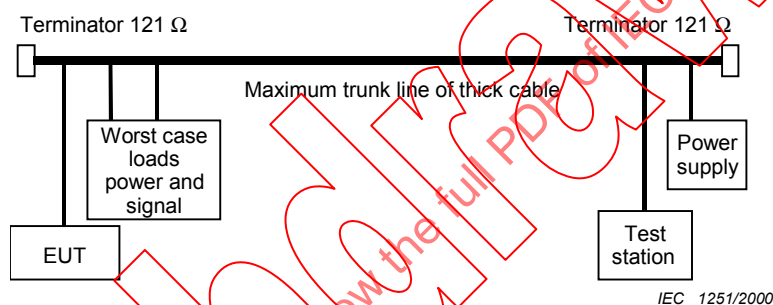


Figure 49 – CDI test configuration

9.2.9.3 Test procedure

Communications shall be established between the test station and the EUT. Any missing acknowledgement bits or bus-off conditions shall be recorded.

9.2.9.4 Criteria for compliance

The recorded missing acknowledgement bits and bus-off conditions shall be as specified by the manufacturer of the EUT.

9.2.10 Electromagnetic compatibility testing

9.2.10.1 General

The test circuit shall be as shown in figure 49:

- the system shall be run at the maximum bit rate supported by the EUT;
- the trunk line shall be constructed of thick cable;

- le bloc d'alimentation utilisé doit répondre aux spécifications décrites dans le tableau 65 (capacité minimale de courant de 4 A);
- la station d'essai utilisée doit pouvoir à la fois produire un trafic de communications valides pour l'appareil en essai (EUT) et surveiller tout le trafic de communications.

Sauf indication contraire, les essais doivent être effectués à la température ambiante de $+23\text{ °C} \pm 5\text{ °C}$.

L'appareil en essai (EUT) doit être monté à l'air libre, connecté au CDI DeviceNet selon les instructions du fabricant et alimenté en tension nominale.

5.2.10.2 Immunité

5.2.10.2.1 Immunité aux décharges électrostatiques (ESD)

Cet essai doit être effectué conformément à la CEI 61000-4-2 et à 4.9.2.2.

La longueur de la ligne principale doit être aussi courte que possible. L'appareil en essai (EUT) doit être connecté directement à la ligne principale. Si cela n'est pas possible, la ligne de dérivation ne doit pas dépasser 1 m.

5.2.10.2.2 Immunité aux champs électromagnétiques rayonnés aux fréquences radio-électriques

Cet essai doit être effectué conformément à la CEI 61000-4-3 et à 4.9.2.3.

La ligne principale doit avoir la longueur maximale admise (voir 5.6.6.2). L'appareil en essai (EUT) doit être connecté à la ligne principale en utilisant une ligne de dérivation ayant la longueur maximale admise (voir 5.6.6.3).

5.2.10.2.3 Immunité aux transitoires électriques rapides en sèves

Cet essai doit être effectué conformément à la CEI 61000-4-4 et à 4.9.2.4.

La ligne principale doit avoir la longueur maximale admise (voir 5.6.6.2). L'appareil en essai (EUT) doit être connecté directement à la ligne principale. Si cela n'est pas possible, la ligne de dérivation ne doit pas dépasser 1 m.

5.2.10.2.4 Immunité aux ondes de choc

Cet essai doit être effectué conformément à la CEI 61000-4-5 et à 4.9.2.5.

La ligne principale doit avoir la longueur maximale admise (voir 5.6.6.2). L'appareil en essai (EUT) doit être connecté directement à la ligne principale. Si cela n'est pas possible, la ligne de dérivation ne doit pas dépasser 1 m.

5.2.10.2.5 Immunité aux perturbations conduites aux fréquences radioélectriques

Cet essai doit être effectué conformément à la CEI 61000-4-6 et à 4.9.2.6.

La ligne principale doit avoir la longueur maximale admise (voir 5.6.6.2). L'appareil en essai (EUT) doit être connecté directement à la ligne principale. Si cela n'est pas possible, la ligne de dérivation ne doit pas dépasser 1 m.

- a power supply shall be used that meets the specifications described in table 65 (minimum current capacity 4 A);
- a test station shall be used that is capable of both producing valid communication traffic to the EUT and monitoring all communication traffic.

Unless otherwise noted, the tests shall be carried out at an ambient temperature of $+23\text{ }^{\circ}\text{C} \pm 5\text{ }^{\circ}\text{C}$.

The EUT shall be mounted in free air and shall be connected to the DeviceNet CDI according to the manufacturer's instructions and supplied with its rated voltage.

9.2.10.2 Immunity

9.2.10.2.1 Electrostatic discharge (ESD) immunity

This test shall be performed in accordance with IEC 61000-4-2 and 8.9.2.2.

The trunk line length shall be as short as possible. The EUT shall be directly connected to the trunk line. If this is not possible, then the drop line shall not exceed 1 m.

9.2.10.2.2 Radiated radio-frequency electromagnetic field immunity

This test shall be performed in accordance with IEC 61000-4-3 and 8.9.2.3.

The trunk line length shall be the maximum allowable (see 5.6.6.2). The EUT shall be connected to the trunk line using the maximum length drop line allowable (see 5.6.6.3).

9.2.10.2.3 Electrical fast transient/burst immunity

This test shall be performed in accordance with IEC 61000-4-4 and 8.9.2.4.

The trunk line length shall be the maximum allowable (see 5.6.6.2). The EUT shall be directly connected to the trunk line. If this is not possible, then the drop line shall not exceed 1 m.

9.2.10.2.4 Surge immunity

This test shall be performed in accordance with IEC 61000-4-5 and 8.9.2.5.

The trunk line length shall be the maximum allowable (see 5.6.6.2). The EUT shall be directly connected to the trunk line. If this is not possible, then the drop line shall not exceed 1 m.

9.2.10.2.5 Conducted radio-frequency disturbance immunity

This test shall be performed in accordance with IEC 61000-4-6 and 8.9.2.6.

The trunk line length shall be the maximum allowable (see 5.6.6.2). The EUT shall be directly connected to the trunk line. If this is not possible, then the drop line shall not exceed 1 m.

5.2.10.3 Emissions

5.2.10.3.1 Emissions rayonnées

Cet essai doit être effectué conformément à la CISPR 11, groupe 1, classe A et à 4.9.3.1.

La ligne principale doit avoir la longueur maximale admise (voir 5.6.6.2). L'appareil en essai (EUT) doit être connecté à la ligne principale en utilisant une ligne de dérivation ayant la longueur maximale admise (voir 5.6.6.3).

5.2.10.3.2 Emissions conduites

Cet essai doit être effectué conformément à la CISPR 11, groupe 1, classe A et à 4.9.3.2.

La ligne principale doit avoir la longueur minimale nécessaire à l'installation d'essai et ne doit pas dépasser la longueur maximale spécifiée en 5.6.6.2. L'appareil en essai (EUT) doit être connecté à la ligne principale en utilisant une ligne de dérivation ayant la longueur maximale admise (voir 5.6.6.3).

5.3 Essais logiques

5.3.1 Généralités

Les essais doivent être effectués uniquement sur des stations qui acceptent la fonctionnalité qui fait l'objet des essais.

Les essais logiques nécessitent la présence d'une station d'essai sur l'interface appareil de commande-appareil, qui peut:

- envoyer des messages DeviceNet;
- recevoir et évaluer des messages DeviceNet;
- enregistrer et horodater le trafic sur le CDI.

5.3.2 Essai de vérification de duplication d'adresses MAC ID

5.3.2.1 Objet de l'essai

Cet essai a pour objet de vérifier que l'appareil en essai (EUT) satisfait aux exigences concernant la gestion des mécanismes de duplication d'adresses MAC ID.

5.3.2.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 50.

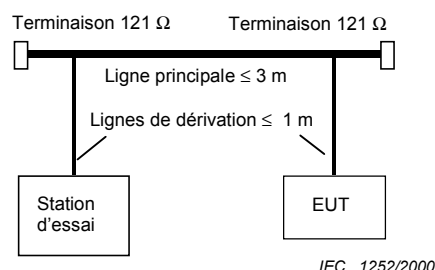


Figure 50 – Circuit d'essai pour les essais logiques

9.2.10.3 Emissions

9.2.10.3.1 Radiated emissions

This test shall be performed in accordance with CISPR 11, group 1, class A, and 8.9.3.1.

The trunk line length shall be the maximum allowable (see 5.6.6.2). The EUT shall be connected to the trunk line using the maximum length drop line allowable (see 5.6.6.3).

9.2.10.3.2 Conducted emissions

This test shall be performed in accordance with CISPR 11, group 1, class A, and 8.9.3.2.

The trunk line length shall be the minimum required for the test fixture and shall not exceed the maximum length specified in 5.6.6.2. The EUT shall be connected to the trunk line using the maximum length drop line allowable (see 5.6.6.3).

9.3 Logical testing

9.3.1 General

Tests shall only be performed on nodes that support the particular functionality being tested.

Logical tests require the presence of a test station on the controller-device interface, which can:

- send DeviceNet messages;
- receive and evaluate DeviceNet messages;
- record and time stamp traffic on the CDI.

9.3.2 Duplicate MAC ID check test

9.3.2.1 Test purpose

The purpose of this test is to verify that the EUT meets the requirements regarding the handling of duplicate MAC ID mechanisms.

9.3.2.2 Test circuit

The test circuit shall be as shown in figure 50.

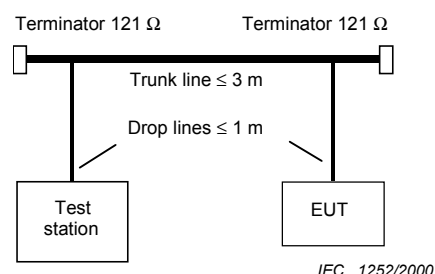


Figure 50 – Test circuit for logical tests

5.3.2.3 Procédure d'essai

5.3.2.3.1 Vérification réussie de duplication d'adresses MAC ID

Le MAC ID de l'appareil en essai (EUT) doit être fixé à 0. L'alimentation externe nécessaire doit être fournie à l'appareil en essai. L'appareil en essai doit être raccordé physiquement à la liaison. La station d'essai doit ensuite être raccordée à la liaison et doit enregistrer le trafic de communications.

5.3.2.3.2 Echec de la vérification de duplication d'adresses MAC ID

Le MAC ID de l'appareil en essai (EUT) doit être fixé à 0. L'alimentation externe nécessaire doit être fournie à l'appareil en essai. L'appareil en essai doit ensuite être raccordé physiquement à la liaison. La station d'essai doit répondre au premier message de demande de vérification de duplication d'adresses MAC ID par un message réponse de vérification de duplication d'adresses MAC ID.

5.3.2.4 Critères de conformité

5.3.2.4.1 Vérification réussie de duplication d'adresses MAC ID

Il faut vérifier que:

- l'appareil en essai (EUT) transmet un message de demande de vérification de duplication d'adresses MAC ID;
- l'appareil en essai transmet une deuxième demande de vérification de duplication d'adresses MAC ID entre 0,9 s et 1,5 s plus tard;
- l'appareil en essai adopte l'état en ligne au plus tôt 0,9 s après la transmission du deuxième message de demande de vérification de duplication d'adresses MAC ID;
- l'appareil en essai ne transmet aucun autre message avant d'adopter l'état en ligne.

5.3.2.4.2 Echec de la vérification de duplication d'adresses MAC ID

L'appareil en essai (EUT) doit adopter l'état de défaut de communication (voir tableau 35) après réception du message réponse de vérification de duplication d'adresses MAC ID.

5.3.3 UCMM

5.3.3.1 Objet de l'essai

Cet essai a pour objet de déterminer si une station qui gère l'UCMM satisfait aux exigences associées à la création de connexions de messagerie explicite.

5.3.3.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 50.

5.3.3.3 Procédure d'essai

La station d'essai doit transmettre un message UCMM de demande d'ouverture en utilisant le groupe de messages 1. Après l'ouverture de cette connexion de messagerie, un message explicite doit être transmis via cette connexion. La station d'essai doit transmettre un message UCMM de demande de fermeture. Elle doit transmettre un deuxième message explicite puis attendre 10 s une réponse.

La procédure ci-dessus doit être répétée pour les messages des groupes 2 et 3.

9.3.2.3 Test procedure

9.3.2.3.1 Successful duplicate MAC ID check

The EUT MAC ID shall be set to 0. Any necessary external power shall be applied to the EUT. The EUT shall be physically connected to the link. The test station shall then be connected to the link and shall record the communication traffic.

9.3.2.3.2 Unsuccessful duplicate MAC ID check

The EUT MAC ID shall be set to 0. Any necessary external power shall be applied to the EUT. The EUT shall then be physically connected to the link. The test station shall respond to the first duplicate MAC ID check request message with a duplicate MAC ID check response message.

9.3.2.4 Criteria for compliance

9.3.2.4.1 Successful duplicate MAC ID check

It shall be verified that:

- the EUT transmits a duplicate MAC ID check request message;
- the EUT transmits a second duplicate MAC ID check request between 0,9 s and 1,5 s later;
- the EUT assumes the on-line state no earlier than 0,9 s after transmitting the second duplicate MAC ID check request message;
- the EUT transmits no other messages before assuming the on-line state.

9.3.2.4.2 Unsuccessful duplicate MAC ID check

The EUT shall assume the communication fault state (see table 35) after receiving the duplicate MAC ID check response message.

9.3.3 UCMM

9.3.3.1 Test purpose

The purpose of this test is to determine whether a node that supports the UCMM meets the requirements associated with the creation of explicit messaging connections.

9.3.3.2 Test circuit

The test circuit shall be as shown in figure 50.

9.3.3.3 Test procedure

The test station shall transmit a UCMM open request message using message group 1. After opening this messaging connection, an explicit message shall be transmitted across this connection. The test station shall transmit a UCMM close request message. The test station shall transmit a second explicit message and wait 10 s for a response.

The above procedure shall be repeated for message groups 2 and 3.

5.3.3.4 Critères de conformité

Il faut vérifier que:

- le premier message explicite est transmis et reçu avec succès pour au moins un groupe de messages;
- l'appareil en essai (EUT) ne transmet pas de réponse au deuxième message explicite.

5.3.4 Attribution d'un jeu de connexions maître/esclave prédéfini pour une connexion de messagerie explicite

5.3.4.1 Objet de l'essai

Cet essai a pour objet de vérifier que l'appareil en essai (EUT) satisfait aux exigences associées à l'attribution d'un jeu de connexions maître/esclave prédéfini pour une connexion de messagerie explicite.

5.3.4.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 50.

5.3.4.3 Procédure d'essai

La station d'essai doit envoyer une demande d'attribution du jeu de connexions maître/esclave prédéfini (choix d'attribution = 1). Après attribution de cette connexion de messagerie, un message explicite doit être transmis via cette connexion. La station d'essai doit libérer le jeu de connexions maître/esclave prédéfini (choix de libération = 1). Elle doit transmettre un deuxième message explicite puis attendre 10 s une réponse.

5.3.4.4 Critères de conformité

Il faut vérifier que:

- le premier message explicite est transmis et reçu comme indiqué dans le jeu de connexions maître/esclave prédéfini;
- l'appareil en essai (EUT) ne transmet pas de réponse au deuxième message explicite.

5.3.5 Attribution d'un jeu de connexions maître/esclave prédéfini pour une connexion de messagerie d'E/S

5.3.5.1 Objet de l'essai

Cet essai a pour objet de vérifier que l'appareil en essai (EUT) satisfait aux exigences associées à l'attribution d'un jeu de connexions maître/esclave prédéfini pour une connexion de messagerie E/S.

5.3.5.2 Circuit d'essai

Le circuit d'essai doit être tel qu'indiqué à la figure 50.

9.3.3.4 Criteria for compliance

It shall be verified that:

- the first explicit message is transmitted and received successfully in at least one message group;
- the EUT does not transmit a response to the second explicit message.

9.3.4 Allocation of a predefined master/slave connection set for an explicit messaging connection

9.3.4.1 Test purpose

The purpose of this test is to determine whether the EUT meets the requirements associated with the allocation of a predefined master/slave connection set for explicit messaging connections.

9.3.4.2 Test circuit

The test circuit shall be as shown in figure 50.

9.3.4.3 Test procedure

The test station shall send a request to allocate the predefined master/slave connection set (allocation choice = 1). After allocating this messaging connection, an explicit message shall be transmitted across this connection. The test station shall de-allocate the predefined master/slave connection set (release choice = 1). The test station shall transmit a second explicit message and wait 10 s for a response.

9.3.4.4 Criteria for compliance

It shall be verified that:

- the first explicit message is transmitted and received as defined in the predefined master/slave connection set;
- the EUT does not transmit a response to the second explicit message.

9.3.5 Allocation of a predefined master/slave connection set for an I/O messaging connection

9.3.5.1 Test purpose

The purpose of this test is to determine whether the EUT meets the requirements associated with the allocation of a predefined master/slave connection set for I/O messaging connections.

9.3.5.2 Test circuit

The test circuit shall be as shown in figure 50.

5.3.5.3 Procédure d'essai

La station d'essai doit envoyer une demande d'attribution du jeu de connexions maître/esclave prédéfini. Après attribution de cette connexion de messagerie, un message d'E/S doit être transmis via cette connexion. La station d'essai doit libérer le jeu de connexions maître/esclave prédéfini. Elle doit transmettre un deuxième message d'E/S et attendre 10 s une réponse.

Cette procédure doit être effectuée pour tous les choix d'attribution acceptés, à l'exception de tous les choix d'attribution de messages explicites.

5.3.5.4 Critères de conformité

Il faut vérifier que:

- le premier message d'E/S est transmis et reçu comme indiqué dans le jeu de connexions maître/esclave prédéfini;
- l'appareil en essai (EUT) ne transmet pas de réponse au deuxième message d'E/S.

9.3.5.3 Test procedure

The test station shall send a request to allocate the predefined master/slave connection set. After allocating this messaging connection, an I/O message shall be transmitted across this connection. The test station shall de-allocate the predefined master/slave connection set. The test station shall transmit a second I/O message and wait 10 s for a response.

This procedure shall be performed for all allocation choices supported, with the exception of all explicit message allocation choices.

9.3.5.4 Criteria for compliance

It shall be verified that:

- the first I/O message is transmitted and received as defined in the predefined master/slave connection set;
- the EUT does not transmit a response to the second I/O message.

Annexe A (normative)

Services communs

A.1 Codes et noms des services DeviceNet

Les codes et noms des services sont donnés dans le tableau A.1.

Tableau A.1 – Codes et noms des services DeviceNet

Code de service (hex)	Nom du service
00	Réservé
01	Get_attributes_all
02	Set_attributes_all
03 – 04	Réservé
05	Reset
06	Start
07	Stop
08	Create
09	Delete
0A-0C	Réservé
0D	Apply_attributes
0E	Get_attribute_single
0F	Réservé
10	Set_attribute_single
11 – 13	Réservé
14	Réponse d'erreur
15	Restore
16	Save
17	No_operation (NOP)
18 – 31	Réservé

A.2 Définition des services communs

A.2.1 Généralités

Les classes et instances d'objets qui utilisent ces services doivent fournir une description détaillée de leur usage spécifique et du format des paramètres, et doivent indiquer si ces services sont nécessaires ou non. Tous les paramètres du champ des données de service, pour une demande ou une réponse positive, doivent se conformer aux indications du tableau A.2, sauf indication contraire.

Annex A (normative)

Common services

A.1 DeviceNet service codes and names

Service codes and names are given in table A.1.

Table A.1 – DeviceNet service codes and names

Service code (hex)	Service name
00	Reserved
01	Get_attributes_all
02	Set_attributes_all
03 – 04	Reserved
05	Reset
06	Start
07	Stop
08	Create
09	Delete
0A – 0C	Reserved
0D	Apply_attributes
0E	Get_attribute_single
0F	Reserved
10	Set_attribute_single
11 – 13	Reserved
14	Error response
15	Restore
16	Save
17	No_operation (NOP)
18 – 31	Reserved

A.2 Common service definitions

A.2.1 General

Object classes/instances that utilise these services shall provide a detailed description of their specific use and parameter formats, and indicate whether or not any of these services are required. All request service data field parameters and all success response data field parameters shall be as shown in table A.2, unless otherwise specified.

Tableau A.2 – Paramètres des champs des données de service et de réponse positive

Nom	Type de données	Description du paramètre
Données spécifiques à l'objet	Spécifique au service pour l'objet/la classe	Contient les paramètres spécifiques à la classe/l'instance

A.2.2 Get_attributes_all (code du service: 01_{hex})**A.2.2.1 Généralités**

Get_attributes_all renvoie le contenu de tous les attributs d'un objet ou d'une classe.

A.2.2.2 Exigences de service

La liste suivante indique les exigences associées au service get_attributes_all:

- la structure des informations du champ des données de service pour la réponse doit être conforme à la structure de la réponse get_attributes_all définie par l'objet ou la classe;
- s'il est répondu à la demande avec succès, le champ des données de service pour la réponse contient les données de l'attribut. Si une erreur est détectée, une réponse d'erreur est alors renvoyée.

A.2.2.3 Paramètres du champ des données de service d'une demande

Aucun.

A.2.2.4 Paramètres du champ des données de service d'une réponse positive

Les données de service pour une réponse positive à get_attributes_all sont indiquées dans le tableau A.3.

Tableau A.3 – Données de service pour la réponse positive à get_attributes_all

Nom	Type de données	Description du paramètre
Données de l'attribut	Spécifique à l'attribut pour l'objet/la classe	Flux d'informations contenant tous les attributs

A.2.3 Set_attributes_all (code du service: 02_{hex})**A.2.3.1 Généralités**

Modifie le contenu des attributs de la classe ou de l'objet.

A.2.3.2 Exigences de service

La liste suivante indique les exigences associées au service set_attributes_all:

- la structure des informations du champ des données de service pour la demande doit être conforme à la définition de la demande set_attributes_all pour l'objet ou la classe;
- si la possibilité de modifier un attribut change en fonction de l'état de l'objet, la définition de l'objet doit fournir une description détaillée de la façon dont ce service est affecté;
- les attributs ne doivent être modifiés que si toutes les vérifications de valeurs spécifiques aux attributs sont satisfaisantes. Le premier attribut échouant à cette vérification doit être spécifié dans le paramètre de code d'erreur supplémentaire du message réponse d'erreur;

Table A.2 – Service data and success response data field parameters

Name	Data type	Description of parameter
Object-specific data	Object/class service-specific	Contains class/instance-specific parameters

A.2.2 Get_attributes_all (service code: 01_{hex})**A.2.2.1 General**

Get_attributes_all returns the contents of all attributes of an object or class.

A.2.2.2 Service requirements

The following list details the requirements associated with the get_attributes_all service:

- the structure of the information in the service data field of the response shall conform to the get_attributes_all response structure defined by the object or class;
- if the request is successfully serviced, the service data field of the response contains the attribute data. If an error is detected, then an error response is returned.

A.2.2.3 Request service data field parameters

None.

A.2.2.4 Success response service data field parameters

Service data for a get_attributes_all success response is given in table A.3.

Table A.3 – Service data for get_attributes_all success response

Name	Data type	Description of parameter
Attribute data	Object/class attribute-specific	A stream of information containing all of the attributes

A.2.3 Set_attributes_all (service code: 02_{hex})**A.2.3.1 General**

Modifies the contents of the attributes of the class or object.

A.2.3.2 Service requirements

The following list details the requirements associated with the set_attributes_all service:

- the structure of the information in the service data field of the request conforms to the definition of the set_attributes_all request for the object or class;
- if the ability to modify an attribute changes based on the state of the object, the object definition shall provide a detailed description of how this service is affected;
- attributes shall be modified only if all attribute-specific value verifications are successful. The first attribute failing verification shall be specified in the additional error code parameter of the error response message;

- si une autre erreur est détectée, une réponse d'erreur est alors renvoyée;
- si toutes les vérifications sont satisfaisantes, les attributs sont modifiés et une réponse positive à set_attributes_all est renvoyée.

A.2.3.3 Paramètres du champ des données de service d'une demande

Les données de service pour une demande set_attributes_all sont indiquées dans le tableau A.4.

Tableau A.4 – Données de service pour la demande set_attributes_all

Nom	Type de données	Description du paramètre
Données de l'attribut	Spécifique à l'attribut pour l'objet/la classe	Flux d'informations contenant tous les attributs

A.2.3.4 Paramètres du champ des données de service d'une réponse positive

Aucun.

A.2.4 Reset (code du service: 05_{hex})

A.2.4.1 Généralités

Invoke le service de remise à zéro de la classe/de l'objet spécifié.

A.2.4.2 Exigences du service

La liste suivante indique les exigences associées au service reset:

- si l'exécution de la demande du service reset place l'objet/l'appareil dans un état qui lui permet de répondre au demandeur, la réponse ne doit pas être renvoyée avant que l'exécution du service soit achevée. Sinon, l'objet ou l'appareil doit répondre avant d'exécuter le service reset;
- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse reset positive doit être renvoyée.

A.2.5 Start (code du service: 06_{hex})

A.2.5.1 Généralités

Invoke le service de démarrage de la classe/de l'objet spécifié.

A.2.5.2 Exigences de service

La liste suivante indique les exigences associées au service start:

- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse start positive doit être renvoyée.

A.2.6 Stop (code du service: 07_{hex})

A.2.6.1 Généralités

Invoke le service d'arrêt de la classe/de l'objet spécifié.

- if any other error is detected, then an error response is returned;
- if all verification checks are successful, then the attributes are modified and a set_attributes_all success response is returned.

A.2.3.3 Request service data field parameters

Service data for a set_attributes_all request is given in table A.4.

Table A.4 – Service data for set_attributes_all request

Name	Data type	Description of parameter
Attribute data	Object/class attribute-specific	A stream of information containing all of the attributes

A.2.3.4 Success response service data field parameters

None.

A.2.4 Reset (service code: 05_{hex})

A.2.4.1 General

Invokes the reset service of the specified class/object.

A.2.4.2 Service requirements

The following list details the requirements associated with the reset service:

- if the execution of the reset service request would place the object/device in a state that enables it to respond to the requester, then the response shall not be returned until the service has completed. Otherwise, the object/device shall respond prior to executing the reset service;
- if an error is detected, then an error response shall be returned. Otherwise, a successful reset response shall be returned.

A.2.5 Start (service code: 06_{hex})

A.2.5.1 General

Invokes the start service of the specified class/object.

A.2.5.2 Service requirements

The following list details the requirements associated with the start service:

- if an error is detected, then an error response shall be returned. Otherwise, a successful start response shall be returned.

A.2.6 Stop (service code: 07_{hex})

A.2.6.1 General

Invokes the stop service of the specified class/object.

A.2.6.2 Exigences du service

La liste suivante indique les exigences associées au service stop:

- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse stop positive doit être renvoyée.

A.2.7 Create (code du service: 08_{hex})

A.2.7.1 Généralités

Crée un nouvel objet dans la classe spécifiée.

A.2.7.2 Exigences de service

La liste suivante indique les exigences associées au service create:

- l'instance de l'objet est créée et initialisée conformément à la définition de l'objet;
- si une erreur est détectée, l'instance de l'objet ne doit pas être créée et une réponse d'erreur doit être renvoyée. Sinon, une réponse create positive doit être renvoyée.

A.2.7.3 Paramètres du champ des données de service d'une réponse positive

Les données de service pour une réponse positive à create sont indiquées dans le tableau A.5.

Tableau A.5 – Données de service pour la réponse positive à create

Nom	Type de données	Description du paramètre
ID d'instance	UINT	Valeur entière attribuée pour identifier l'objet nouvellement créé
Données spécifiques à l'objet	Spécifique au service pour l'objet/la classe	Contient les paramètres spécifiques à la classe/l'instance

A.2.8 Delete (code du service: 09_{hex})

A.2.8.1 Généralités

Supprime une instance d'objet de la classe spécifiée.

A.2.8.2 Exigences de service

La liste suivante indique les exigences associées au service delete:

- toutes les ressources sont libérées;
- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse delete positive doit être renvoyée.

A.2.9 Apply_attributes (code du service: 0D_{hex})

A.2.9.1 Généralités

Rend actives les valeurs d'attribut en attente d'utilisation.

A.2.6.2 Service requirements

The following list details the requirements associated with the stop service:

- if an error is detected, then an error response shall be returned. Otherwise, a successful stop response shall be returned.

A.2.7 Create (service code: 08_{hex})

A.2.7.1 General

Creates a new object within the specified class.

A.2.7.2 Service requirements

The following list details the requirements associated with the create service:

- the object instance is created and initialized in accordance with the object definition;
- if an error is detected, the object instance shall not be created and an error response shall be returned. Otherwise, a successful create response shall be returned.

A.2.7.3 Success response service data field parameters

Service data for a create success response is given in table A.5.

Table A.5 – Service data for create success response

Name	Data type	Description of parameter
Instance ID	UINT	Integer value assigned to identify the newly created object
Object-specific data	Object/class service-specific	Contains class/instance specific parameters

A.2.8 Delete (service code: 09_{hex})

A.2.8.1 General

Deletes an object instance of the specified class.

A.2.8.2 Service requirements

The following list details the requirements associated with the delete service:

- all resources are released;
- if an error is detected, then an error response shall be returned. Otherwise, a successful delete response shall be returned.

A.2.9 Apply_attributes (service code: 0D_{hex})

A.2.9.1 General

Causes pending attribute values to become active.

A.2.9.2 Exigences de service

La liste suivante indique les exigences associées au service `apply_attributes`:

- les données des attributs en attente doivent être vérifiées avant d'être utilisées;
- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse positive à `apply_attributes` doit être renvoyée.

A.2.10 `Get_attribute_single` (code du service: 0E_{hex})

A.2.10.1 Généralités

Renvoie le contenu de l'attribut spécifié.

A.2.10.2 Exigences de service

La liste suivante indique les exigences associées au service `get_attribute_single`:

- le service demande à la classe/l'objet de renvoyer le contenu de l'attribut spécifié;
- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse positive à `get_attribute_single` doit être renvoyée avec les données de l'attribut demandées.

A.2.10.3 Paramètres du champ des données de service d'une demande

Les données de service pour une demande `get_attribute_single` sont indiquées dans le tableau A.6.

Tableau A.6 – Données de service pour une demande `get_attribute_single`

Nom	Type de données	Description du paramètre
ID d'attribut	USINT	Identifie l'attribut à lire/envoyer

A.2.10.4 Paramètres du champ des données de service d'une réponse positive

Les données de service pour une réponse positive à `get_attribute_single` sont indiquées dans le tableau A.7.

Tableau A.7 – Données de service pour la réponse positive à `get_attribute_single`

Nom	Type de données	Description du paramètre
Données de l'attribut	Spécifique à l'attribut	Contient les données de l'attribut demandées

A.2.11 `Set_attribute_single` (code du service: 10_{hex})

A.2.11.1 Généralités

Modifie la valeur d'un attribut.

A.2.11.2 Exigences de service

La liste suivante indique les exigences associées au service `set_attribute_single`:

- les données de l'attribut sont vérifiées avant que la modification soit effectuée;
- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse positive à `set_attribute_single` doit être renvoyée.

A.2.9.2 Service requirements

The following list details the requirements associated with the `apply_attributes` service:

- data for pending attributes shall be verified before it is used;
- if an error is detected, then an error response shall be returned. Otherwise, a successful `apply_attributes` response shall be returned.

A.2.10 `Get_attribute_single` (service code: `0Ehex`)

A.2.10.1 General

Returns the contents of the specified attribute.

A.2.10.2 Service requirements

The following list details the requirements associated with the `get_attribute_single` service:

- the service causes the class/object to return the contents of the specified attribute;
- if an error is detected, then an error response shall be returned. Otherwise, a successful `get_attribute_single` response shall be returned along with the requested attribute data.

A.2.10.3 Request service data field parameters

Service data for a `get_attribute_single` request is given in table A.6.

Table A.6 – Service data for `get_attribute_single` request

Name	Data type	Description of parameter
Attribute ID	USINT	Identifies the attribute to be read/returned

A.2.10.4 Success response service data field parameters

Service data for a `get_attribute_single` success response is given in table A.7.

Table A.7 – Service data for `get_attribute_single` success response

Name	Data type	Description of parameter
Attribute data	Attribute-specific	Contains the requested attribute data

A.2.11 `Set_attribute_single` (service code: `10hex`)

A.2.11.1 General

Modifies an attribute value.

A.2.11.2 Service requirements

The following list details the requirements associated with the `set_attribute_single` service:

- the attribute data is verified prior to the modification taking place;
- if an error is detected, then an error response shall be returned. Otherwise, a successful `set_attribute_single` response shall be returned.

A.2.11.3 Paramètres du champ des données de service d'une demande

Les données de service pour une demande `set_attribute_single` sont indiquées dans le tableau A.8.

Tableau A.8 – Données de service pour une demande `set_attribute_single`

Nom	Type de données	Description du paramètre
ID d'attribut	USINT	Identifie l'attribut à lire/renvoyer
Données de l'attribut	Spécifique à l'attribut	Contient la valeur de modification de l'attribut spécifié

A.2.12 Réponse d'erreur (code du service: 14_{hex})

A.2.12.1 Généralités

Ce service est utilisé pour répondre négativement à un message de demande explicite reçu antérieurement. Ce code ne doit être présent que dans un message réponse explicite.

A.2.12.2 Comportement requis

Sans objet.

A.2.12.3 Paramètres du champ des données de service d'une demande

Sans objet.

A.2.12.4 Paramètres du champ des données de service

Les données de service pour une réponse d'erreur sont indiquées dans le tableau A.9.

Tableau A.9 – Données de service pour une réponse d'erreur

Nom	Type de données	Description du paramètre
Code d'erreur général	USINT	Identifie l'erreur rencontrée. Voir l'annexe B pour la liste des codes d'erreurs généraux
Code d'erreur supplémentaire	USINT	Contient une valeur spécifique à l'objet ou au service qui décrit davantage la condition d'erreur. Si l'objet qui répond n'a pas d'informations supplémentaires à spécifier, la valeur 255 doit être placée dans ce champ

A.2.13 Restore (code du service: 15_{hex})

A.2.13.1 Généralités

Restaure le contenu des attributs d'une classe ou d'un objet à partir d'un emplacement mémoire accessible par le service `save`. Les données des attributs sont copiées de la zone de stockage dans la zone de mémoire active utilisée par la classe/l'objet.

A.2.13.2 Exigences de service

La liste suivante indique les exigences associées au service `restore`:

- les données de l'attribut doivent être vérifiées avant de les copier de la «zone de stockage» dans la zone «active utilisée»;

A.2.11.3 Request service data field parameters

Service data for a set_attribute_single request is given in table A.8.

Table A.8 – Service data for set_attribute_single request

Name	Data type	Description of parameter
Attribute ID	USINT	Identifies the attribute to be read/returned
Attribute data	Attribute-specific	Contains the value to which the specified attribute is to be modified

A.2.12 Error response (service code: 14_{hex})

A.2.12.1 General

This service is used to respond negatively to a previously received explicit request message. This code shall only be present in an explicit response message.

A.2.12.2 Required behaviour

Not applicable.

A.2.12.3 Request service data field parameters

Not applicable.

A.2.12.4 Service data field parameters

Service data for an error response request is given in table A.9.

Table A.9 – Service data for error response request

Name	Data type	Description of parameter
General error code	USINT	Identifies the encountered error. See annex B for a list of general error codes
Additional code	USINT	Contains an object/service-specific value that further describes the error condition. If the responding object has no additional information to specify, then the value 255 shall be placed within this field

A.2.13 Restore (service code: 15_{hex})

A.2.13.1 General

Restores the contents of a class/object's attributes from a storage location accessible by the save service. The attribute data is copied from a storage area to the currently active memory area used by the class/object.

A.2.13.2 Service requirements

The following list details the requirements associated with the restore service:

- the attribute data shall be verified before the copy from the “storage area” to the “actively used” area is performed;

- si la possibilité de modifier un attribut change en fonction de l'état de l'objet, la définition de l'objet doit fournir une description détaillée de la façon dont ce service est affecté;
- les attributs ne doivent être modifiés que si toutes les vérifications de valeurs spécifiques aux attributs sont satisfaisantes. Si une vérification échoue, une réponse d'erreur doit être renvoyée. Le premier attribut manquant à réaliser cette vérification doit être spécifié dans le paramètre de code d'erreur supplémentaire du message réponse;
- si une autre erreur est détectée, une réponse d'erreur doit être renvoyée;
- si toutes les vérifications sont satisfaisantes, les attributs sont modifiés et une réponse positive à restore doit être renvoyée.

A.2.14 Save (code du service: 16_{hex})

A.2.14.1 Généralités

Copie le contenu des attributs d'une classe ou d'un objet dans un emplacement accessible par le service restore.

A.2.14.2 Exigences de service

La liste suivante indique les exigences associées au service save:

- le service ne doit signaler la réussite qu'une fois la copie terminée et vérifiée;
- si une erreur est détectée, une réponse d'erreur doit être renvoyée. Sinon, une réponse positive à save doit être renvoyée.

A.2.15 No_operation (NOP) (code du service: 17_{hex})

A.2.15.1 Généralités

Ce service fait renvoyer à l'objet récepteur une réponse no_operation. L'objet récepteur ne doit effectuer aucune autre action interne.

A.2.15.2 Comportement requis

Si l'objet vers lequel la demande est dirigée accepte le service, une réponse positive doit être renvoyée. Sinon, une réponse d'erreur doit être renvoyée.

A.2.15.3 Paramètres du champ des données de service d'une demande

Aucun.

A.2.15.4 Paramètres du champ des données de service d'une réponse positive

Aucun.

- if the ability to modify an attribute changes based on the state of the object, the object definition shall provide a detailed description of how this service is affected;
- the attributes shall be modified only if all attribute-specific value verifications are successful. If a verification fails, an error response shall be returned. The first attribute failing verification shall be specified in the additional error code parameter of the error response message;
- if any other error is detected, then an error response shall be returned;
- if all verification checks are successful, then the attributes are modified and a restore success response shall be returned.

A.2.14 Save (service code: 16_{hex})

A.2.14.1 General

Copies the contents of a class/object's attributes to a location accessible by the restore service.

A.2.14.2 Service requirements

The following list details the requirements associated with the save service:

- the service shall report success only after the copy has been completed and verified;
- if an error is detected, then an error response shall be returned. Otherwise, a successful save response shall be returned.

A.2.15 No_operation (NOP) (service code: 17_{hex})

A.2.15.1 General

This service causes the receiving object to return a no_operation response. The receiving object shall not perform any other internal action.

A.2.15.2 Required behaviour

If the object to which the request is delivered supports the service, then a success response shall be returned. Otherwise, an error response shall be returned.

A.2.15.3 Request service data field parameters

None.

A.2.15.4 Success response service data field parameters

None.

Annexe B (normative)

Codes d'erreurs DeviceNet

Le tableau B.1 énumère les codes d'erreurs pouvant se présenter dans le champ «code d'erreur général» d'un message réponse d'erreur.

Tableau B.1 – Codes d'erreurs DeviceNet

Code d'erreurs (hex)	Nom de l'erreur	Description de l'erreur
00 – 01		Réservé
02	Ressource non disponible	Les ressources nécessaires pour que l'objet exécute le service demandé ne sont pas disponibles
03 – 07		Réservé
08	Service non accepté	Le service demandé n'a pas été incorporé ou n'a pas été défini pour cette classe/instance d'objet
09	Valeur d'attribut incorrecte	Données d'attribut incorrectes détectées
0A		Réservé
0B	Déjà dans le mode/l'état demandé	L'objet est déjà dans le mode/l'état demandé par le service
0C	Conflit d'état de l'objet	L'objet ne peut pas exécuter le service demandé dans son mode/état actuel
0D		Réservé
0E	Attribut non modifiable	L'attribut spécifié ne peut être modifié
0F	Violation de privilège	Une vérification de permission/de privilège a échoué
10	Conflit d'état de l'appareil	Le mode/l'état actuel de la station ne permet pas d'exécuter le service demandé
11	Les données de réponse sont trop longues	Les données à transmettre dans la mémoire tampon des réponses sont plus grandes que la taille attribuée à la mémoire tampon
12		Réservé
13	Données insuffisantes	Le service n'a pas fourni suffisamment de données pour effectuer l'opération spécifiée
14	Attribut non accepté	L'attribut spécifié dans la demande n'est pas disponible
15	Trop de données	Le service a fourni plus de données que prévu
16	L'objet n'existe pas	L'objet spécifié n'existe pas dans la station
17		Réservé
18	Aucune donnée d'attribut stockée	Les données d'attribut de cet objet n'ont pas été sauvegardées avant le service demandé
19	Echec de l'opération de stockage	La tentative de sauvegarde des données de l'attribut de cet objet n'a pas réussi
1A – 1E		Réservé
1F	Erreur spécifique au fabricant	Une erreur spécifique au fabricant s'est produite. Le champ de code d'erreur supplémentaire de la réponse indique l'erreur particulière rencontrée. Ce code d'erreur général ne doit être utilisé que lorsqu'aucun code d'erreur indiqué dans ce tableau ou dans la définition d'une classe d'objet ne décrit l'erreur de manière appropriée
20	Paramètre incorrect	Un paramètre associé à la demande n'était pas valide. Ce code doit être utilisé quand un paramètre ne correspond pas aux exigences de cette spécification et/ou aux exigences de la spécification d'un objet applicatif
21 – CF		Réservé
D0 – FF		Réservé aux codes d'erreurs spécifiques à la classe de l'objet

Annex B (normative)

DeviceNet error codes

Table B.1 lists the error codes that may be present in the general error code field of an error response message.

Table B.1 – DeviceNet error codes

Error code (hex)	Error name	Description of error
00 – 01		Reserved
02	Resource unavailable	Resources needed for the object to perform the requested service are unavailable
03 – 07		Reserved
08	Service not supported	The requested service was not implemented or was not defined for this object class/instance
09	Invalid attribute value	Invalid attribute data detected
0A		Reserved
0B	Already in requested mode/state	The object is already in the mode/state being requested by the service
0C	Object state conflict	The object is unable to perform the requested service in its current mode/state
0D		Reserved
0E	Attribute not settable	The attribute specified may not be modified
0F	Privilege violation	A permission/privilege check failed
10	Device state conflict	The current mode/state of the node does not allow the execution of the requested service
11	Reply data too large	The data to be transmitted in the response buffer is larger than the allocated response buffer
12		Reserved
13	Not enough data	The service did not supply enough data to perform the specified operation
14	Attribute not supported	The attribute specified in the request is not supported
15	Too much data	The service supplied more data than was expected
16	Object does not exist	The object specified does not exist in the node
17		Reserved
18	No stored attribute data	The attribute data of this object was not saved prior to the requested service
19	Store operation failure	The attempt to save the attribute data of this object was not successful
1A – 1E		Reserved
1F	Manufacturer-specific error	A manufacturer-specific error has been encountered. The additional code field of the error response defines the particular error encountered. This general error code shall only be used when none of the error codes which are either listed in this table or defined within an object class definition accurately reflects the error
20	Invalid parameter	A parameter associated with the request was invalid. This code shall be used when a parameter does not meet the requirements of this specification and/or the requirements defined in an application object-specification
21 – CF		Reserved
D0 – FF		Reserved for object class-specific error codes

Annexe C (normative)

Définition de l'attribut chemin de connexion

C.1 Généralités

La définition des attributs chemin (path) définit le ou les objets spécifiques de l'application vers/depuis lesquels les données sont transférées. Ces attributs peuvent éventuellement contenir des informations de configuration pour le ou les objets applicatifs concernés.

Un attribut chemin de connexion (connection path) est divisé en segments comme suit.

- segment logique;
- segment symbolique;
- segment de données.

Un segment contient des informations qui définissent son type, le format des données, et les données elles-mêmes de ce segment.

C.2 Type et format d'un segment

C.2.1 Introduction

Le premier octet de chaque segment d'un attribut chemin (path) est l'octet contenant le type et le format. Cet octet détermine comment interpréter le segment (voir figure C.1).

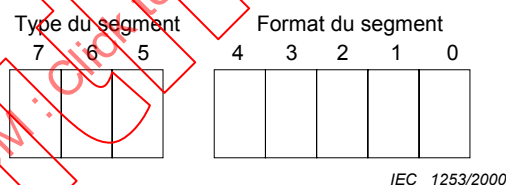


Figure C.1 – Type/format du segment

La signification des bits du format de segment est déterminée par le type de segment spécifié. Les bits du type de segment et les formats associés sont décrits dans les paragraphes suivants.

C.2.2 Segment logique

Le segment logique sélectionne une classe d'objet applicatif, une instance d'objet applicatif et/ou un attribut d'objet applicatif particuliers dans une station (voir figure C.2).