

DIN EN 61375-3-3

ICS 35.240.60

**Elektronische Betriebsmittel für Bahnen –
Zug-Kommunikations-Netzwerk (TCN) –
Teil 3-3: CANopen-Consist-Netzwerk (CNN)
(IEC 61375-3-3:2012);
Englische Fassung EN 61375-3-3:2012**

Electronic railway equipment –
Train communication network (TCN) –
Part 3-3: CANopen Consist Network (CCN)
(IEC 61375-3-3:2012);
English version EN 61375-3-3:2012

Matériel électronique ferroviaire –
Réseau embarqué de train (TCN) –
Partie 3-3: Réseau de rame CANopen (CCN)
(CEI 61375-3-3:2012);
Version anglaise EN 61375-3-3:2012

Gesamtumfang 115 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2012-07-26 angenommene Europäische Norm als DIN-Norm ist 2015-10-01.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 61375-3-3:2009-09.

Für dieses Dokument ist das nationale Arbeitsgremium UK 351.1 „Fahrzeuge“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom TC 9 „Electrical equipment and systems for railways“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils aktuellste Ausgabe des in Bezug genommenen Dokuments.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe des Dokuments.

Der Zusammenhang der zitierten Dokumente mit den entsprechenden Deutschen Dokumenten ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die englische Fassung der EN 61375-3-3 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt, und Abschnitt 3, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für viele der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Nationaler Anhang NA (informativ)

1 Anwendungsbereich^{N1)}

Dieser Teil von IEC 61375 spezifiziert den Datenkommunikationsbus innerhalb von CANopen-basierten Consists. CANopen wurde entwickelt für, ist jedoch nicht beschränkt auf, die Verwendung in industriellen Automatisierungsanwendungen. Diese Anwendungen dürfen Geräte wie Eingang/Ausgang-Module, Streckensteuerungen, Mensch-Maschine-Schnittstellen, Sensoren, geschlossene Regelkreise, Codierer, hydraulische Ventile oder programmierbare Steuerungen einschließen.

Auf dem Anwendungsfeld der Bahnfahrzeuge werden CANopen-Netzwerke in Netzwerksystemen in Consists verwendet, wie z. B. Bremssteuerungssystem, Dieselmotorensteuerung und die Steuerung der Innen- und Außenbeleuchtung. Zusätzlich kann CANopen als Consist-Netzwerk verwendet werden, um den Datenaustausch zwischen unterschiedlichen Subsystemen innerhalb eines Bahnfahrzeuges oder einer Gruppe von Bahnfahrzeugen, die dasselbe Consist-Netzwerk nutzen, zu ermöglichen.

Dieser Teil von IEC 61375 gilt für Betriebsmittel und Geräte, welche in Consist-Netzwerken auf der Grundlage von CANopen innerhalb von TCN-Architekturen, wie in IEC 61375-1 beschrieben, betrieben werden.

Die Anwendbarkeit dieser Norm für eine TCN-Ausführung erlaubt die individuelle Konformitätsprüfung der Ausführung selbst, und ist eine Voraussetzung für weitere Überprüfungen der Interoperabilität unterschiedlicher TCN-Ausführungen. In jedem Fall ist vom Lieferanten der Nachweis der Kompatibilität zwischen Zug-Backbone und Consist-Netzwerk zu erbringen.

Dieser Teil von IEC 61375 gilt für den Aufbau von Kommunikationssystemen in freigeformten Zügen. Zusätzlich kann sie für Triebzüge und Mehrfachtraktionen anwendbar sein, wenn dieses zwischen Käufer und Lieferant vereinbart wurde.

3 Begriffe, Abkürzungen und Konventionen

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach ISO 11898-1, ISO 11898-2, IEC 61375-1, IEC 61375-2-1 und EN 50325-4, und die folgenden:

3.1.1

ASCII

7-Bit-Zeichenkodierung nach ISO/IEC 646

3.1.2

CANopen-Gerät

mit einem CANopen-basierten Consist-Netzwerk verbundenes Endgerät

3.1.3

Feldgerät

vernetzte eigenständige physikalische Einheit eines Automatisierungssystems, welche in der Lage ist spezifische Funktionen in einem bestimmten Zusammenhang, und begrenzt durch seine Schnittstellen auszuführen

3.1.4

logisches Consist-Netzwerk

Art der Datenübertragung im Consist-Netzwerk, ohne Berücksichtigung der physikalischen Verbindung der Endgeräte

^{N1)} Die Nummerierung aus IEC 61375-3-3 wurde beibehalten.

3.1.5

logisches Gerät

Abbildung eines Feldgerätes in Bezug auf seinen Zweck und sein Verhalten entsprechend eines Feldgerätemodells, welches die Daten und das Verhalten des Gerätes aus Netzwerkperspektive beschreibt

3.1.6

Layer Setting Services

Dienste zur Anpassung von Bitrate und Knoten-ID über die Kommunikationsschnittstelle des CANopen-Geräts

3.1.7

Knoten-ID

- a) netzwerkweiter eindeutiger Bezeichner für jedes CANopen-Gerät
- b) Geräteadresse mit 7-Bit-Kodierung in CANopen-basierten Consist-Netzwerken

3.1.8

Objekt

Einheit mit einer klar definierten Abgrenzung und Identität, welche einen Zustand und ein Verhalten aufweist

3.1.9

virtuelles Gerät

Softwareeinheit, die in der Lage ist, funktionelle Elemente eines Feldgerätes auszuführen

3.2 Abkürzungen

Für die Anwendung dieses Dokuments gelten die Abkürzungen nach ISO 11898-1, ISO 11898-2, IEC 61375-1, IEC 61375-2-1 und EN 50325-4, und die Folgenden:

ASCII	standardisierter Code für die Zeichendarstellung der USA (en: American Standard Code for Information Interchange)
BNF	Backus-Normalform (en: Backus Naur Form)
CAN	Controller Area Network
CPU	Hauptprozessor (en: Central Processing Unit)
CR	Wagenrücklauf (en: Carriage Return)
CRLF	Wagenrücklauf und Zeilenvorschub (en: Carriage Return and Line Feed)
DCF	Datei der Konfiguration des Gerätes (en: Device Configuration File)
FSA	endlicher Zustandsautomat (en: Finite State Automaton)
ID	Bezeichner (en: Identifier)
LF	Zeilenvorschub (en: Line Feed)
LSS	Layer Setting Services
NMT	Netzverwaltung (en: Network Management)
OSI	Kommunikation offener Systeme (en: Open System Interconnect)
PAS	Zugriffsdienst für Prozessdatenobjekte (en: PDO Access Service)
PDO	Prozessdatenobjekt (en: Process Data Object)
SAS	Zugriffsdienst für Servicedatenobjekte (en: Service Data Object Access Service)
SRD	Servicedaten abfragendes Gerät (en: SDO requesting device)

3.3 Konventionen

Es müssen die in IEC 61375-1 angegebenen Konventionen angewendet werden.

**Electronic railway equipment -
Train communication network (TCN) -
Part 3-3: CANopen Consist Network (CCN)
(IEC 61375-3-3:2012)**

Matériel électronique ferroviaire -
Réseau embarqué de train (TCN) -
Partie 3-3: Réseau de rame CANopen
(CCN)
(CEI 61375-3-3:2012)

Elektronische Betriebsmittel für Bahnen -
Zug-Kommunikations-Netzwerk (TCN) -
Teil 3-3: CANopen-Consist-Network
(CNN)
(IEC 61375-3-3:2012)

This European Standard was approved by CENELEC on 2012-07-26. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

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

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

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

CENELEC

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

Management Centre: Avenue Marnix 17, B - 1000 Brussels

Foreword

The text of document 9/1646/FDIS, future edition 1 of IEC 61375-3-3, prepared by IEC/TC 9 "Electrical equipment and systems for railways" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 61375-3-3:2012.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2013-04-26
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2015-07-26

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

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

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

Endorsement notice

The text of the International Standard IEC 61375-3-3:2012 was approved by CENELEC as a European Standard without any modification.

CONTENTS

INTRODUCTION.....	11
1 Scope	12
2 Normative references	12
3 Terms, definitions and abbreviations	13
3.1 Terms and definitions	13
3.2 Abbreviations	14
3.3 Conventions	14
4 Architecture.....	14
4.1 Content	14
4.2 Logical CANopen-based consist network.....	14
4.3 Network topology.....	15
4.4 Addressing	15
4.5 Data classes.....	16
5 Physical layer.....	16
5.1 Content	16
5.2 Cabling.....	16
5.3 Connector	16
5.4 Physical medium attachment.....	17
5.5 Physical signaling	18
6 Data Link layer	18
6.1 Content	18
6.2 CANopen data link layer	18
7 CANopen application layer.....	18
7.1 Content	18
7.2 Reference model	18
7.3 Field device model.....	19
7.4 CANopen communication objects	21
7.5 CANopen object dictionary.....	21
7.6 Predefined CANopen communication objects	22
7.6.1 Content.....	22
7.6.2 Object 1000 _h : Device type.....	22
7.6.3 Object 1001 _h : Error register	23
7.6.4 Object 1014 _h : COB-ID emergency object	23
7.6.5 Object 1017 _h : Heartbeat producer	23
7.6.6 Object 1018 _h : Identity object	23
7.6.7 Object 1029 _h : Error behavior.....	23
7.6.8 Object 67FF _h : Device type	23
7.6.9 Service data objects (SDOs)	23
7.6.10 Process data objects (PDOs).....	23
8 Application data.....	24
8.1 Content	24
8.2 CANopen application data representation.....	24
8.3 Recommended representation principle of application data.....	24
8.3.1 Content.....	24

8.3.2	Application data for door control.....	24
8.3.3	Consumed door control application objects.....	24
8.3.4	Produced door control application objects	25
9	CANopen network management.....	28
9.1	Content.....	28
9.2	CANopen NMT slave functionality.....	29
9.3	CANopen manager functionality.....	29
9.3.1	General	29
9.3.2	Object dictionary usage.....	30
9.3.3	Redundant networks	30
9.4	CANopen NMT start-up.....	31
9.4.1	NMT startup.....	31
9.4.2	NMT startup simple.....	34
9.4.3	Start process boot NMT slave	35
9.5	Boot NMT slave.....	36
9.5.1	Check configuration	42
9.5.2	Check NMT state	43
9.5.3	NMT flying master start up	43
9.5.4	Error status.....	43
9.6	Error control	45
9.6.1	Start error control	45
9.6.2	Error handler	46
9.6.3	Bootup handler	47
9.7	Additional NMT master services and protocols	47
9.8	Object dictionary entries	47
9.8.1	Object 1020 _h : Verify configuration	47
9.8.2	Object 102A _h : NMT inhibit time	48
9.8.3	Object 1F20 _h : Store DCF	49
9.8.4	Object 1F22 _h : Concise DCF	50
9.8.5	Object 1F26 _h : Expected configuration date	52
9.8.6	Object 1F27 _h : Expected configuration time	53
9.8.7	Object 1F80 _h : NMT startup.....	54
9.8.8	Object 1F81 _h : NMT slave assignment.....	56
9.8.9	Object 1F82 _h : Request NMT	59
9.8.10	Object 1F83 _h : Request node guarding	61
9.8.11	Object 1F84 _h : Device type identification	63
9.8.12	Object 1F85 _h : Vendor identification	64
9.8.13	Object 1F86 _h : Product code	65
9.8.14	Object 1F87 _h : Revision number.....	66
9.8.15	Object 1F88 _h : Serial number	67
9.8.16	Object 1F89 _h : Boot time.....	69
9.8.17	Object 1F8A _h : Restore configuration	69
9.8.18	Object 1F91 _h : Self starting nodes timing parameters	70
10	Gateway functions.....	71
10.1	Content.....	71
10.2	Gateway architecture.....	72
10.3	General principles and services	73
10.3.1	Content	73

10.3.2	Gateway class definitions	73
10.3.3	Service primitives definitions	73
10.4	Network access service specification	73
10.4.1	SDO access services	73
10.4.2	PDO access services	75
10.4.3	CANopen NMT services	78
10.4.4	Device failure management services	81
10.4.5	CANopen interface configuration services	82
10.4.6	Gateway management services	84
10.4.7	Manufacturer-specific services	85
10.5	ASCII mapping of network access services	85
10.5.1	Content	85
10.5.2	Definitions	86
10.5.3	Network access command specification	89
11	Train network management	97
11.1	Content	97
11.2	Manager, Agents and interfaces (informative)	98
11.3	Management message protocol (informative)	98
11.4	Object interfaces (informative)	98
11.5	CANopen-specific management services	99
11.5.1	General	99
11.5.2	Agent interfaces on a Station connected to CANopen consist network	99
11.5.3	Management message structure for CANopen consist networks	100
11.5.4	Notation for the CANopen specific SIF_codes	100
11.5.5	Notation for a call CANopen management message	100
11.5.6	Notation for a reply CANopen management message	100
11.5.7	Notation for the TNM CANopen services command codes	101
11.6	TNM CANopen services	102
11.6.1	Content	102
11.6.2	Call_Write_CANopen_Command (with reservation)	102
11.6.3	Reply_Write_CANopen_Command (with reservation)	102
11.6.4	Call_Read_CANopen_Command (without reservation)	103
11.6.5	Reply_Read_CANopen_Command (without reservation)	103
12	CANopen management message data handling	104
12.1	General	104
12.2	Message data format	105
12.3	Requirements for message data communication within CANopen networks	106
12.4	Object 1F78 _n : CANopen message data reception	107
13	Conformance testing	108
	Bibliography	109
	Annex ZA (normative) Normative references to international publications with their corresponding European publication	110
	Annex ZZ (informative) Coverage of Essential Requirements of EU Directives	111
	Figure 1 – Logical network architecture of the consist network	15
	Figure 2 – Network topology of CANopen-based consist network	15
	Figure 3 – 9-pin D-sub connector	16

Figure 4 – 5-pin micro style connector.....	17
Figure 5 – Field device model	19
Figure 6 – Minimum field device.....	20
Figure 7 – CANopen device structure	20
Figure 8 – Structure of the device type object.....	23
Figure 9 – Object structure	25
Figure 10 – Object structure.....	26
Figure 11 – Object structure.....	27
Figure 12 – NMT startup, part 1	31
Figure 13 – NMT startup, part 2	33
Figure 14 – NMT startup simple	34
Figure 15 – Start process boot NMT slave.....	35
Figure 16 – Boot NMT slave, part 1.....	36
Figure 17 – Boot NMT slave, part 2.....	38
Figure 18 – Boot NMT slave, part 3.....	40
Figure 19 – Check configuration.....	42
Figure 20 – Check NMT state.....	43
Figure 21 – Start error control	45
Figure 22 – Error handler.....	46
Figure 23 – Bootup handler.....	47
Figure 24 – Data stream definition of concise DCF	51
Figure 25 – Object structure.....	54
Figure 26 – Bit structure of the configuration value	55
Figure 27 – Object structure of the value.....	57
Figure 28 – Bit structure of the configuration value	57
Figure 29 – Gateway between Train backbone and CANopen consist network.....	72
Figure 30 – Management messages (informative).....	98
Figure 31 – Agent interface on a CANopen (gateway) station for message data	99
Figure 32 – Call_Write_CANopen_Command	102
Figure 33 – Reply_Write_CANopen_Command.....	103
Figure 34 – Call_Read_CANopen_Command (without reservation)	103
Figure 35 – Reply_Read_CANopen_command (without reservation)	104
Figure 36 – CANopen device capable to handle TNM management messages	104
Figure 37 – Message data format comparison	106
Table 1 – Pinning for 9-pin D-sub connector.....	17
Table 2 – Pinning for 5-pin micro style connector	17
Table 3 – Bit timing.....	18
Table 4 – CANopen object dictionary structure	22
Table 5 – Value definition	25
Table 6 – Object description	25
Table 7 – Entry description	25

Table 8 – Value definition	26
Table 9 – Object description.....	26
Table 10 – Entry description.....	27
Table 11 – Value definition.....	28
Table 12 – Object description.....	28
Table 13 – Entry description.....	28
Table 14 – Error status	44
Table 15 – Object description.....	48
Table 16 – Entry description.....	48
Table 17 – Object description.....	49
Table 18 – Entry description.....	49
Table 19 – Object description.....	49
Table 20 – Entry description.....	50
Table 21 – Object description.....	51
Table 22 – Entry description.....	52
Table 23 – Object description.....	52
Table 24 – Entry description.....	53
Table 25 – Object description.....	54
Table 26 – Entry description.....	54
Table 27 – Value NMT master (bit: 0)	55
Table 28 – Value Start all nodes (bit: 1).....	55
Table 29 – Value NMT master start (bit: 2)	55
Table 30 – Value Start node (bit: 3).....	55
Table 31 – Reset all nodes (bit: 4).....	55
Table 32 – Flying master (bit: 5).....	55
Table 33 – Stop all nodes (bit: 6)	56
Table 34 – Exceptions for NMT start-up capable devices	56
Table 35 – Object description.....	56
Table 36 – Entry description.....	56
Table 37 – NMT slave (bit: 0)	57
Table 38 – NMT boot slave (bit: 2)	57
Table 39 – Mandatory (bit: 3)	57
Table 40 – Reset communication (bit: 4).....	57
Table 41 – Software version (bit: 5).....	57
Table 42 – Software update (bit: 6)	58
Table 43 – Restore (bit: 7)	58
Table 44 – Object description.....	58
Table 45 – Entry description.....	59
Table 46 – Value definition.....	60
Table 47 – Object description.....	60
Table 48 – Entry description.....	61
Table 49 – Value definition.....	62

Table 50 – Object description	62
Table 51 – Entry description	63
Table 52 – Object description	64
Table 53 – Entry description	64
Table 54 – Object description	65
Table 55 – Entry description	65
Table 56 – Object description	66
Table 57 – Entry description	66
Table 58 – Object description	67
Table 59 – Entry description	67
Table 60 – Object description	68
Table 61 – Entry description	68
Table 62 – Object description	69
Table 63 – Entry description	69
Table 64 – Object description	69
Table 65 – Entry description	70
Table 66 – Object description	70
Table 67 – Entry description	71
Table 68 – Upload SDO service	74
Table 69 – Download SDO parameters.....	75
Table 70 – Configure SDO timeout parameters.....	75
Table 71 – Configure RPDO service parameters.....	76
Table 72 – Configure TPDO service parameters	77
Table 73 – Read PDO data service parameters	77
Table 74 – Write PDO data service parameters	78
Table 75 – RPDO received service parameters	78
Table 76 – Start node service parameters	78
Table 77 – Stop node service parameters	79
Table 78 – Set node to pre-operational service parameters	79
Table 79 – Reset node service parameters.....	79
Table 80 – Reset communication service parameters	80
Table 81 – Enable node guarding service parameters.....	80
Table 82 – Disable node guarding service parameters.....	80
Table 83 – Start heartbeat consumer service parameters	81
Table 84 – Disable heartbeat consumer service parameters	81
Table 85 – Error control event received parameters.....	81
Table 86 – Read device error service parameters.....	82
Table 87 – Emergency event received service parameters.....	82
Table 88 – Initialize gateway service parameters.....	82
Table 89 – Store configuration service parameters	83
Table 90 – Restore configuration service parameters	83
Table 91 – Set heartbeat producer service parameters	83

Table 92 – Set node-ID service parameters	84
Table 93 – Start emergency consumer service parameters	84
Table 94 – Stop emergency consumer service parameters	84
Table 95 – Set default network service parameters	85
Table 96 – Start default node-ID service parameters	85
Table 97 – Get version service parameters	85
Table 98 – Syntax and CANopen data types	86
Table 99 – Command notation in BNF	87
Table 100 – Response notation	88
Table 101 – Internal error code (InEC)	88
Table 102 – Notation for event triggered messages	88
Table 103 – Syntax for upload SDO command	89
Table 104 – Examples for upload SDO command	89
Table 105 – Syntax for Download SDO command	89
Table 106 – Examples for download SDO command	89
Table 107 – Syntax for configure SDO timeout command	89
Table 108 – Syntax for configure RPDO command	90
Table 109 – Examples for configure RPDO command	90
Table 110 – Syntax for configure TPDO command	90
Table 111 – Examples for configure TPDO command	90
Table 112 – Syntax for read PDO data command	91
Table 113 – Response syntax for read PDO data command	91
Table 114 – Syntax for write PDO data command	91
Table 115 – Syntax for RPDO receive command	91
Table 116 – Examples RPDO received command	91
Table 117 – Syntax for start node command	92
Table 118 – Syntax for stop node command	92
Table 119 – Syntax set node to pre-operational command	92
Table 120 – Syntax reset node command	92
Table 121 – Syntax reset communication command	92
Table 122 – Syntax enable node guarding command	93
Table 123 – Syntax disable node guarding command	93
Table 124 – Syntax start heartbeat consumer command	93
Table 125 – Syntax disable heartbeat consumer command	93
Table 126 – Syntax for error control event received command	93
Table 127 – Syntax for read device error command	94
Table 128 – Syntax for emergency event received command	94
Table 129 – Syntax for initialize gateway command	94
Table 130 – Bit rate indices	94
Table 131 – Syntax for store configuration command	95
Table 132 – Storage specifier	95
Table 133 – Syntax restore configuration command	95

Table 134 – Syntax set heartbeat producer command.....	95
Table 135 – Syntax set node-ID command	95
Table 136 – Syntax set default network command	96
Table 137 – Syntax set default node-ID command	96
Table 138 – Syntax for get version command	96
Table 139 – Response syntax for get version command.....	97
Table 140 – Example for get version response	97
Table 141 – Management message structure	100
Table 142 – CANopen specific SIF_codes	100
Table 143 – Notation for a call CANopen management message	100
Table 144 – Notation for a reply CANopen management message	101
Table 145 – TNM CANopen services command codes (reservation required)	101
Table 146 – TNM CANopen services command codes (reservation not required)	102
Table 147 – Value definition for Call_Write_CANopen_Command	102
Table 148 – Value definition Reply_Write_CANopen_Command	103
Table 149 – Value definition for Call_Read_CANopen_Command (without reservation)	103
Table 150 – Value definition for Reply_Read_CANopen_Command (without reservation) ...	104
Table 151 – Object description.....	107
Table 152 – Entry description	107

INTRODUCTION

TCN is an International Standard with the aim of defining interfaces so as to achieve plug-in compatibility:

- a) between equipment located in different vehicles or consists, and
- b) between equipment and devices located within the same vehicle or consist.

One of the key success factors for the deployment of any technology is standardization and ensuring interoperability among various implementations. To facilitate interoperability a conformance test should be implemented.

In this part of IEC 61375, the TCN deals with:

the consist network based on CANopen.

In addition gateway devices between the Train Backbone and the CANopen-based consist network are considered.

This standard is structured into 13 clauses.

ELECTRONIC RAILWAY EQUIPMENT – TRAIN COMMUNICATION NETWORK (TCN) –

Part 3-3: CANopen Consist Network (CCN)

1 Scope

This part of IEC 61375 specifies the data communication bus inside consists that are based on CANopen. CANopen was developed for use in, but is not limited to, industrial automation applications. These applications may include devices such as input/output modules, motion controllers, human machine interfaces, sensors, closed-loop controllers, encoders, hydraulic valves or programmable controllers.

In the application field of rail vehicles CANopen networks are utilized to network subsystems in consists such as e.g. brake control system, diesel engine control system and interior or exterior lighting control system. In addition CANopen is utilized as consist network to enable the data exchange between the different subsystems within one single rail vehicle or a group of rail vehicles sharing the same Consist Network.

This part of IEC 61375 applies to all equipment and devices operated on a CANopen-based consist network within TCN architecture as described in IEC 61375-1.

The applicability of this standard to a TCN implementation allows for individual conformance checking of the implementation itself and is a pre-requisite for further interoperability checking between different TCN implementations. In any case, proof of compatibility between Train Backbone and the Consist Network will have to be brought by the supplier.

This part of IEC 61375 applies to the architecture of communication systems in Open trains. In addition it may be applicable to closed trains and multiple unit trains when so agreed between purchaser and supplier.

2 Normative references

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

IEC 61131 (all parts): *Programmable controllers*

IEC 61375-1, ed3, *Electronic railway equipment – Train Communication Network (TCN) – Part 1: General Architecture*

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

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

ISO/IEC 646:1991 *Information technology – ISO 7-bit coded character set for information interchange*

ISO/IEC 9899:1999, *Programming languages – C*

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

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

EN 50325-4:2002, *Industrial communication subsystems based on ISO 11898 (CAN) for controller-device interfaces – Part 4: CANopen*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 11898-1, ISO 11898-2, IEC 61375-1, IEC 61375-2-1 and EN 50325-4, as well as the following apply.

3.1.1

ASCII

7-bit coded character set according to ISO/IEC 646

3.1.2

CANopen device

End Device connected to a CANopen-based consist network

3.1.3

Field device

networked independent physical entity of an automation system capable of performing specified functions in a particular context and delimited by its interfaces

3.1.4

logical consist network

the way the data passes through the consist network without considering the physical interconnection of the End Devices

3.1.5

logical device

representation of a field device in terms of its objects and behavior according to a field device model that describes the device's data and behavior as viewed through a network

3.1.6

Layer Setting Services

services for adjusting bit rate and Node-ID via the communication interface of the CANopen device

3.1.7

Node-ID

- a) network-wide unique identifier for each CANopen device
- b) 7-bit coded Device Address in CANopen-based consist networks

3.1.8

Object

entity with a well-defined boundary and identity that encapsulates state and behavior

3.1.9

virtual device

entity of software capable of accomplishing a functional element of a field device

3.2 Abbreviations

For the purposes of this document, abbreviations given in ISO 11898-1, ISO 11898-2, IEC 61375-1, IEC 61375-2-1 and EN 50325-4, as well as the following apply.

ASCII American Standard Code for Information Interchange

BNF Backus Naur Form

CAN Controller Area Network

CPU Central Processing Unit

CR Carriage Return

CRLF Carriage Return and Line Feed

DCF Device configuration file

FSA Finite state automaton

ID Identifier

LF Line Feed

LSS Layer Setting Services

NMT Network management

OSI Open system interconnect

PAS PDO access service

PDO Process data object

SAS Service data object access service

SRD SDO requesting device

3.3 Conventions

The conventions given in IEC 61375-1 shall apply.

4 Architecture

4.1 Content

This clause provides the definition of the architecture for Consist Networks based on CANopen.

4.2 Logical CANopen-based consist network

The overall Logical CANopen consist network, connecting several Virtual Devices, is illustrated in Figure 1. The CANopen-based Consist Network interconnects several Virtual Devices and sub-systems such as e.g. Train operating system (TOS), Monitoring and safety system (MSS), Auxiliary operating system (AUX), Power (drive) system (PDS), Running gear system (RGS), Brake control system (BCS), Ancillary operating system (ANC), Vehicle linkage system (VLS), Exterior lighting system (ELS), Interior lighting system (ILS), Door control system (DCS), HVAC system (HS), Passenger information system (PIS), Diagnostic system (DS), or Train-to-ground communication system (TCS).

The Logical CANopen-based consist network is connected to the Train Backbone via a Gateway. The Gateway manages the information exchange as well as the process data marshalling between the Train Backbone and the Consist Network.

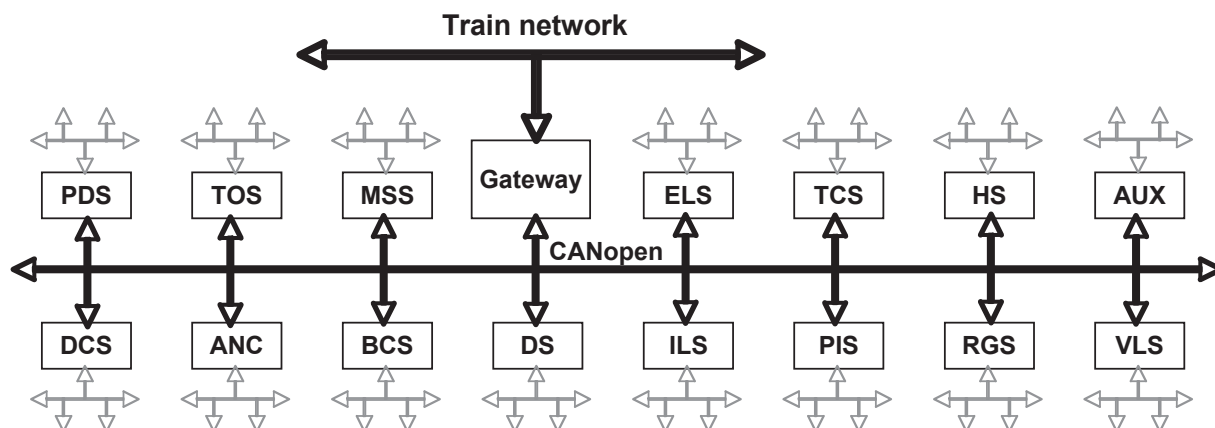


Figure 1 – Logical network architecture of the consist network

4.3 Network topology

The network topology shall be as close as possible to a single line structure with termination resistors on both sides of the network as suggested by ISO11898-2. The total network length shall not exceed 450 m, if a transmission rate of 125 kbit/s is used.

The accumulated stub length shall not exceed 110 m, and the single stub length shall not exceed 22 m, in case a transmission rate of 125 kbit/s is used. It is recommended to keep the stub lengths as short as possible. The CAN transceiver chips shall be galvanically isolated. Optocouplers are placed between CAN controller and CAN transceiver. This affects the maximum bus length depending upon the propagation delay of the optocouplers. The termination resistor on both ends of the bus lines shall be 120 Ω or higher.

The network topology is illustrated in Figure 2.

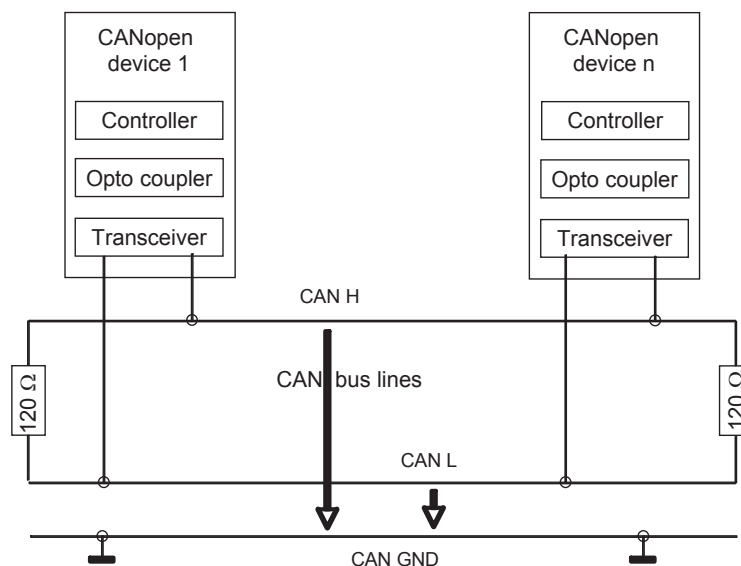


Figure 2 – Network topology of CANopen-based consist network

4.4 Addressing

Any End Device that is connected to a CANopen-based Consist Network requires a unique CANopen Node-ID in the range from 01_h to 7F_h. The CANopen Node-ID is either adjusted via hardware switches or via software. In case the CANopen Node-ID is adjusted via the CANopen communication

interface of the device, the CANopen Layer Setting Services shall be used. The CANopen Node-ID shall not be adjusted via the CANopen object dictionary of the related device.

The definition of the Layer Setting Services (LSS) is not in the scope of this standard

NOTE LSS are specified in the document CiA 305.

4.5 Data classes

To be able to exchange meaningful data across the CANopen-based Consist Network, it is necessary that the format of this data and its meaning is known by the producer and the consumer(s). This specification models this by the concept of data types.

The encoding rules define the representation of values of data types and the transfer syntax for the representations. Values are represented as bit sequences. Bit sequences are transferred in sequences of octets (bytes). For numerical data types the encoding is little endian style.

Applications often require data types beyond the basic data types. Using the compound data type mechanism, it is possible to extend the list of available data types. Some general extended data types are defined as “Visible String” or “Time of Day”. Compound data types are a means to implement user defined “DEFTYPES” in the terminology of the EN 50325-4 and not “DEFSTRUCTS”.

The data types and encoding rules provided in EN 50325-4 shall apply.

5 Physical layer

5.1 Content

This clause provides definitions for the physical layer of a CANopen-based consist network.

5.2 Cabling

The main trunk circuit shall be at least a single twisted pair of nominal characteristic impedance as defined in ISO 11898-2. In addition a CAN_Ground line as clear reference level for CAN_H and CAN_L voltage level shall be applied.

5.3 Connector

It is recommended that devices connected to a CANopen-based consist network support either the 9-pin D-sub connector or the 5-pin micro style (M12) connector. The connectors are illustrated in Figure 3 and Figure 4. The 9-pin D-sub connector shall support a pinning as defined in Table 1. The 5-pin micro style connector (M12) shall support a pinning as defined in Table 2. As the CANopen devices shall be galvanically isolated, the optional V+ line shall only be used for supply of CAN transceiver and optocoupler, in case no additional power supply is used.

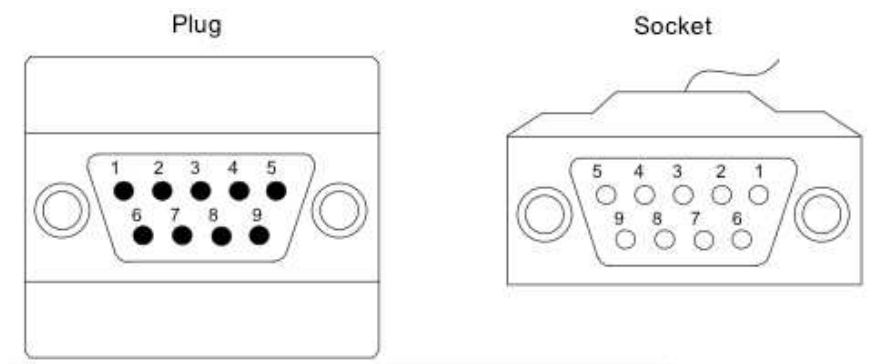


Figure 3 – 9-pin D-sub connector

Table 1 – Pinning for 9-pin D-sub connector

Pin	Signal	Description
1	-	Reserved
2	CAN_L	CAN_L bus line (dominant low)
3	CAN_GND	CAN ground
4	-	Reserved
5	(CAN_SHLD)	Optional CAN shield
6	(GND)	Optional ground
7	CAN_H	CAN_H bus line (dominant high)
8	-	Reserved
9	(CAN_V+)	Optional CAN external positive supply (dedicated for supply of transceiver and optocouplers for galvanically isolated bus nodes)

NOTE It is highly recommended not to connect CAN_GND and GND.

In case shielding is required, it is recommended to connect a shield via the metal housing of the used connector. Optionally the CAN_SHLD pin can be used.

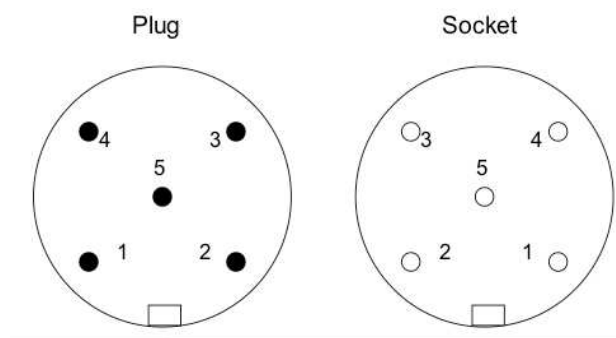


Figure 4 – 5-pin micro style connector

Table 2 – Pinning for 5-pin micro style connector

Pin	Signal	Description
1	(CAN_SHLD)	Optional CAN shield
2	(CAN_V+)	Optional CAN external positive supply (dedicated for supply of transceiver and optocouplers, if galvanic isolation of the bus node applies)
3	CAN_GND	Ground / 0V / V-
4	CAN_H	CAN_H bus line (dominant high)
5	CAN_L	CAN_L bus line (dominant low)

5.4 Physical medium attachment

The physical medium for a device connected to CANOpen-based consist networks shall be a differentially driven two-wire bus line with common return according to high-speed transmission specification in ISO 11898-2.

Using the high-speed transceiver according to ISO 11898-2 the maximum rating for V_{CAN_H} and V_{CAN_L} shall be +16 V. Galvanic isolation between CANOpen devices is optional. It is recommended to use a CAN transceiver that is capable of sustaining misconnection of any of the wires of the connector including the optional V+ voltages of up to 30 V.

5.5 Physical signaling

The bit encoding/decoding and synchronization shall meet the requirements defined in ISO 11898-1.

The bit timing shall meet the requirements defined in ISO 11898-1 and the definitions as given in Table 3. The recommended location of the sample point is as close as possible to 87,5 % of the related bit time.

Table 3 – Bit timing

Bit rate	Nominal bit time t_b μs	Valid range for location of sample point %
1 Mbit/s	1	75 to 90
800 kbit/s	1,25	75 to 90
500 kbit/s	2	85 to 90
250 kbit/s	4	85 to 90
125 kbit/s	8	85 to 90
50 kbit/s	20	85 to 90
20 kbit/s	50	85 to 90
10 kbit/s	100	85 to 90

Devices connected to a CANopen-based consist network shall support a bit rate of 125 kbit/s. Optionally further bit rates as given in Table 3 may be supported.

6 Data Link layer

6.1 Content

This clause provides definitions for the data link layer of a CANopen-based consist network.

6.2 CANopen data link layer

The described CANopen-based consist network shall be based on a data link layer and its sub-layers according to ISO 11898-1.

This specification is based on the CAN base frame format with 11-bit CAN-Identifier. Therefore it is not required to support the CAN extended frame format with 29-bit CAN-Identifier.

NOTE As certain applications require the usage of the CAN extended frame format, the network can be operated in this mode as well, in case all connected CANopen devices support this format.

7 CANopen application layer

7.1 Content

This clause provides definitions with regard to the upper layers of the ISO OSI reference model for devices connected to a CANopen-based consist network.

7.2 Reference model

Devices connected to a CANopen-based consist network compliant to this specification shall use the reference model, as defined in EN 50325-4.

7.3 Field device model

Devices connected to a CANopen-based consist network shall comply with the field device model as given in Figure 5.

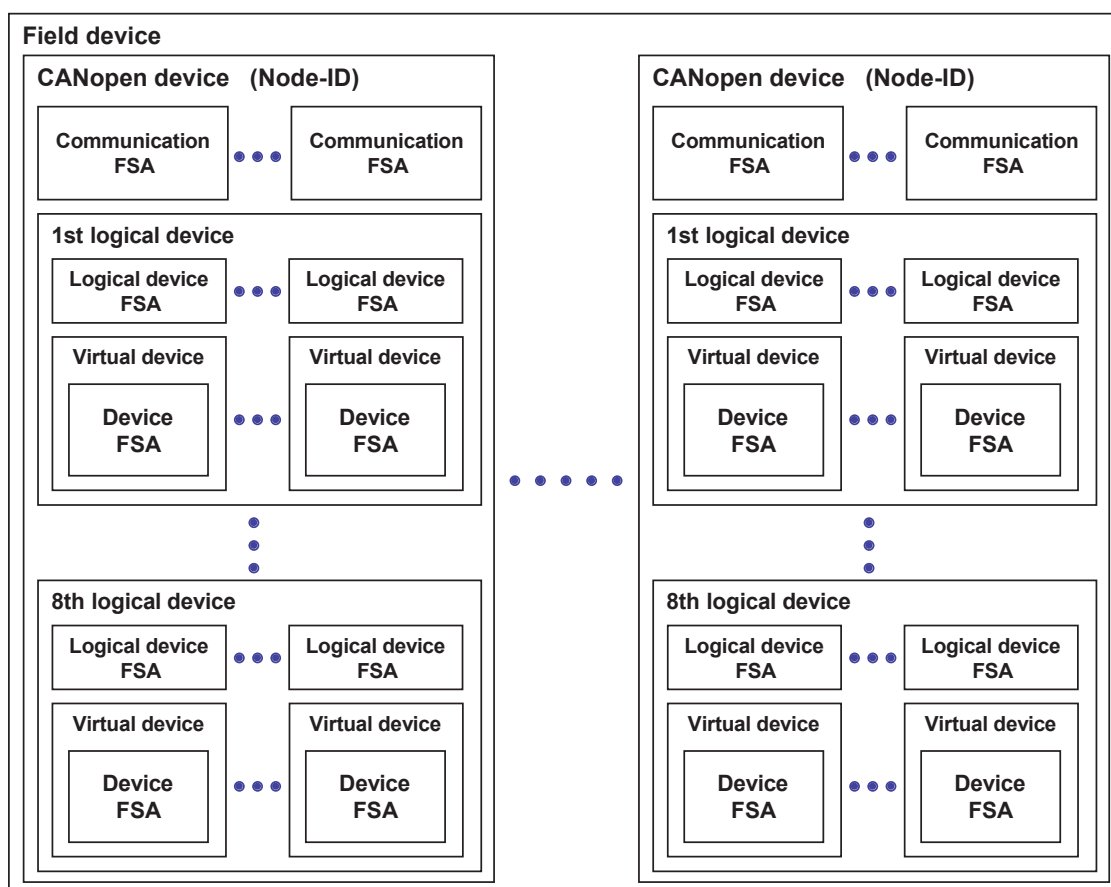


Figure 5 – Field device model

The field device as illustrated in Figure 5 shall provide at least one CANopen device. Each CANopen device within the field device shall provide at least one associated network interface comprising the data link layer protocol (see Clause 6) and the physical layer (see Clause 5), one CANopen Node-ID, and at least one communication FSA. The first communication FSA contains the NMT slave state machine as defined in EN 50325-4. Additional communication FSAs contain an emergency state machine (see EN 50325-4) and others. A CANopen device shall support at least one and may support up to eight logical devices. Each logical device may contain a number of virtual devices and optionally a logical device FSA. A virtual device contains a virtual device FSA and is not distributed to several logical devices. The minimum field device is shown in Figure 6.

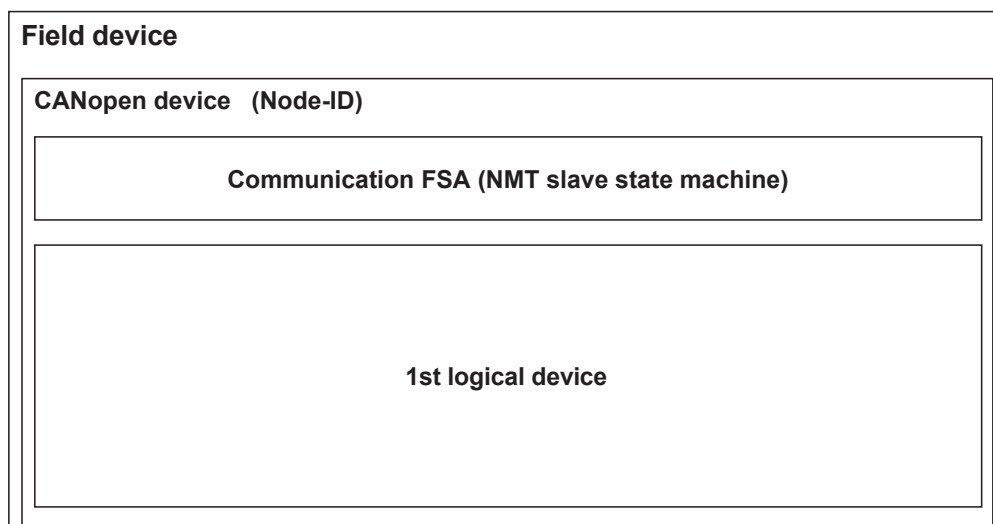


Figure 6 – Minimum field device

A CANopen device is structured as shown in Figure 7:

- Communication – This function unit provides the communication objects and the appropriate functionality to transport data items via the underlying network structure.
- Object dictionary – The object dictionary is a collection of all the data items which have an influence on the behavior of the application objects, the communication objects and the state machine used on this device.
- Application – The application comprises the functionality of the device with respect to the interaction with the process environment.

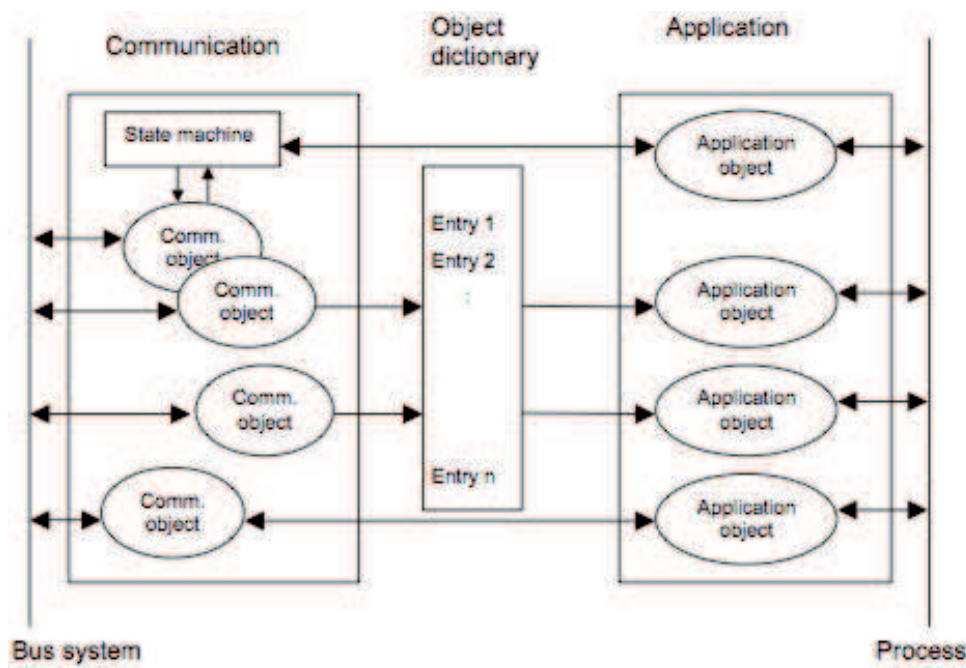


Figure 7 – CANopen device structure

7.4 CANopen communication objects

Devices connected to a CANopen-based consist network shall support all mandatory communication objects of EN 50325-4 as well as the CANopen network management state machine as defined in EN 50325-4.

In order to reduce configuration effort for simple CANopen networks a CAN-ID allocation scheme is defined. These CAN-IDs shall be available in the NMT state Pre-operational directly after the NMT state Initialization (if no modifications have been stored). The objects SYNC, TIME, EMCY write and PDO may be deleted and re-created with new CAN-IDs by means of dynamic distribution. A CANopen device shall provide the corresponding CAN-IDs only for the supported communication objects. The CAN-ID allocation scheme is defined in EN 50325-4.

7.5 CANopen object dictionary

The CANopen object dictionary, as defined in EN 50325-4, contains a maximum of 65536 objects that are addressed through a 16-bit index and up to 256 sub-indices per object, which are addressed through an 8-bit sub-index.

The static data types at indices from 0001_h to 001F_h contain type definitions for standard data types like BOOLEAN, INTEGER, UNSIGNED, floating point, string, etc.

Complex data types at indices from 0020_h to 003F_h are pre-defined structures that are composed of standard data types and are common to all CANopen devices.

Manufacturer-specific complex data types at indices from 0040_h to 005F_h are structures composed of standard data types but are specific to a particular CANopen device.

CANopen device profiles may define additional data types specific to their device type. The static data types and the complex data types defined by the CANopen device profile are listed at indices from 0060_h to 025F_h.

A CANopen device optionally provides the structure of the supported complex data types (indices from 0020_h to 005F_h and from 0060_h to 025F_h) at read access to the corresponding index. Sub-index 00_h then provides the highest sub-index supported at this index, and the following sub-indices contain the data type encoded as UNSIGNED16 according to EN 50325-4.

The communication profile area at indices from 1000_h to 1FFF_h contains the communication specific parameters. These objects are common to all CANopen devices.

The standardized profile area at indices from 6000_h to 9FFF_h contains all data objects common to a class of CANopen devices that may be read or written via the network. The CANopen device profiles use objects from 6000_h to 9FFF_h to describe parameters and functionality.

The object dictionary concept caters for optional features, which means a manufacturer may not provide certain extended functionality on his CANopen devices but if he wishes to do so he shall do it in a pre-defined fashion. Space is left in the object dictionary at indices from 2000_h to 5FFF_h for manufacturer-specific functionality.

The network variables at indices from A000_h to AFFF_h contain input variables and output variables, which are part of a programmable CANopen device.

The system variables at indices from B000_h to BFFF_h contain input variables and output variables, which are part of an underlying CANopen network in a hierarchical sense.

The general CANopen object dictionary structure is illustrated in Table 4.

Table 4 – CANopen object dictionary structure

Index	Object
0000 _h	not used
0001 _h – 001F _h	Static data types
0020 _h – 003F _h	Complex data types
0040 _h – 005F _h	Manufacturer-specific complex data types
0060 _h – 025F _h	Device profile specific data types
0260 _h – 03FF _h	reserved
0400 _h – 0FFF _h	reserved
1000 _h – 1FFF _h	Communication profile area
2000 _h – 5FFF _h	Manufacturer-specific profile area
6000 _h – 67FF _h	Standardized profile area 1 st logical device
6800 _h – 6FFF _h	Standardized profile area 2 nd logical device
7000 _h – 77FF _h	Standardized profile area 3 rd logical device
7800 _h – 7FFF _h	Standardized profile area 4 th logical device
8000 _h – 87FF _h	Standardized profile area 5 th logical device
8800 _h – 8FFF _h	Standardized profile area 6 th logical device
9000 _h – 97FF _h	Standardized profile area 7 th logical device
9800 _h – 9FFF _h	Standardized profile area 8 th logical device
A000 _h – AFFF _h	Standardized network variable area
B000 _h – BFFF _h	Standardized system variable area
C000 _h – FFFF _h	reserved

7.6 Predefined CANopen communication objects

7.6.1 Content

This clause provides the basic CANopen communication capability for CANopen devices that participate in a CANopen-based consist network according to this specification.

7.6.2 Object 1000_h: Device type

This object provides the type of device and its functionality. The value 0000_h for the device profile number indicates a logical device that does not follow a standardized profile. In this case the additional information is 0000_h (if no further logical device is implemented) or FFFF_h (if a further logical device is implemented).

For multiple logical device modules the additional information parameter is FFFF_h and the device profile number referenced by object 1000_h is the profile of the first logical device in the object dictionary. All other profiles of a multiple logical device module identify their profiles at objects 67FF_h + x * 800_h with x = internal number of the logical device (from 1 to 8) minus 1. These objects describe the device type of the preceding logical device, having the very same value definition as object 1000_h.

Figure 8 illustrates the structure of the object. The value definition, the object description and the entry description are specified in EN 50325-4.

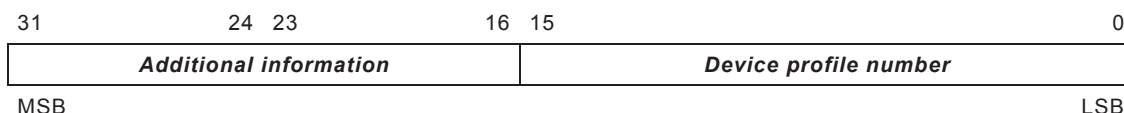


Figure 8 – Structure of the device type object

NOTE Devices connected to CANopen-based consist networks may follow the CANopen application profile for train vehicle control networks CiA 421. This CANopen application profile defines application data, based on UIC 556, that is exchanged in a CANopen-based consist network.

7.6.3 Object 1001_h: Error register

This object provides the error information. The CANopen device maps internal errors into this object. This error information is published as part of the emergency message. The value definition, object description and entry description are specified in EN 50325-4.

7.6.4 Object 1014_h: COB-ID emergency object

This object shall be implemented. It is specified in EN 50325-4. The CAN-ID, which is part of this object, shall not be changed.

7.6.5 Object 1017_h: Heartbeat producer

All CANopen devices that are connected to a CANopen-based consist network shall implement this object. It is specified in EN 50325-4.

The devices shall support heartbeat message transmissions from 100 ms to 1 000 ms.

7.6.6 Object 1018_h: Identity object

This object provides general information about the device as specified in EN 50325-4.

7.6.7 Object 1029_h: Error behavior

This object specifies to which state the device is set, in case a communication error or a device-internal error is detected. It is specified in EN 50325-4.

7.6.8 Object 67FF_h: Device type

This object shall describe the first logical device in a multiple device module according to EN 50325-4.

7.6.9 Service data objects (SDOs)

Any CANopen device supports the first SDO server channel. CANopen devices connected to a CANopen-based consist network may support additional SDO server or client channels.

In case an additional SDO channel is supported, the related SDO parameter set is supported in the CANopen object dictionary as defined in EN 50325-4.

There are no further SDO channels pre-defined by this specification.

7.6.10 Process data objects (PDOs)

CANopen devices operated in a CANopen-based consist network may support up to 512 PDOs in transmit as well as receive direction.

In case a CANopen device supports a PDO, the related PDO communication parameter and mapping entries are supported in the CANopen object dictionary as defined in EN 50325-4.

There are no PDOs pre-defined.

8 Application data

8.1 Content

This clause provides the representation of application data that is communicated between a CANopen-based consist network and a Train Backbone via a Gateway.

NOTE 1 The application data is described in application profile of TCN.

In general application data is managed in the CANopen object dictionary index ranges 2000_h to $5FFF_h$ and 6000_h to $9FFF_h$. A manufacturer specific device behavior is controlled via the index range 2000_h to $5FFF_h$. A standardized CANopen device behavior is controlled via objects in the index range 6000_h to $9FFF_h$.

NOTE 2 Object dictionary entries in the index range 6000_h to $9FFF_h$ are specified in CANopen device and application profiles and available at CAN in Automation.

8.2 CANopen application data representation

As the application data for TCN is not defined yet, the representation of the TCN application data in the standardized object dictionary index range is not defined.

8.3 Recommended representation principle of application data

8.3.1 Content

In this clause a representation principle of application data is provided for the representation of application data in CANopen-based consist networks. Therefore it is possible to map the process to CANopen process data objects (PDOs) for process data transfer.

8.3.2 Application data for door control

This clause provides the recommended representation principle by means of door control information, which is communicated between a CANopen-based consist network and a Train Backbone via a Gateway. The represented process data is therefore available in the CANopen-based consist network and can be communicated via the CANopen communication objects.

The access type is given for the interface of the gateway device to the CANopen-based consist network. Other devices such as e.g. door controllers or door units, which are connected to a CANopen-based consist network manage the data as defined in this clause, at the given CANopen object dictionary index. The access type of the object may be changed to the appropriate access type as defined in EN 50325-4.

NOTE The illustrated application data is derived from the document UIC 556 and is given in the representation of the CANopen application profile CiA 421 train vehicle control network.

8.3.3 Consumed door control application objects

8.3.3.1 Object 6007_h : External door status word export

This object shall indicate the status of the external doors of the local rail vehicle. Via the gateway to the Train Backbone, this information is available in the TCN. Figure 9 specifies the object structure and Table 5 specifies the value definition. Table 6 specifies the object description and Table 7 specifies the entry description.

NOTE The object shall correspond to R3-telegram octet 20, which is defined in UIC 556.

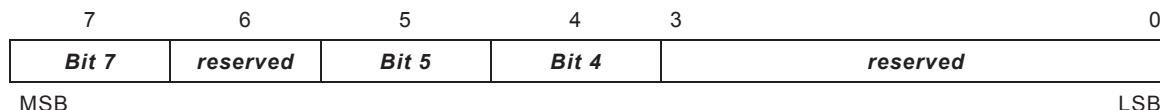


Figure 9 – Object structure

Table 5 – Value definition

Bit	Value	Value definition
Bit4	0	At least one left door is open
	1	All left doors locked
Bit 5	0	At least one right door is open
	1	All right doors locked
Bit 7	0	Side selective door blocking is not in operation
	1	Side selective door blocking is in operation
Reserved		Reserved (shall be ignored)

Table 6 – Object description

Attribute	Value
INDEX	6007 _h
Name	External doors status word export
Object code	Variable
Data type	UNSIGNED8
Category	Optional

Table 7 – Entry description

Attribute	Value
Sub-index	00 _h
Access	rw
PDO mapping	Optional
Value range	See value definition
Default value	Manufacturer specific

8.3.4 Produced door control application objects

8.3.4.1 Object 6006_h: External door status word import

This object shall provide the status of the external doors of other rail vehicles. Each sub-index shall provide the door status of that vehicle, which corresponds with the UIC vehicle number. Therefore the external door status of other vehicles is available within the CANopen-based consist network. The overall door status of all vehicles is provided in sub-index 21_h. Figure 10 specifies the object structure and Table 8 specifies the value definition. Table 9 specifies the object description and Table 10 specifies the entry description.

NOTE The object shall correspond to R3-telegram octet 20, which is defined in UIC 556.

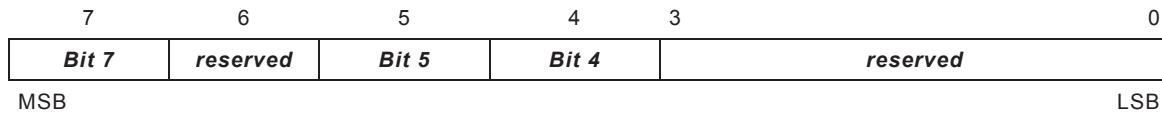


Figure 10 – Object structure

Table 8 – Value definition

Bit	Value	Value definition
Bit4	0	At least one left door is open
	1	All left doors locked
Bit 5	0	At least one right door is open
	1	All right doors locked
Bit 7	0	Side selective door blocking is not in operation
	1	Side selective door blocking is in operation
Reserved		Reserved (shall be ignored)

Table 9 – Object description

Attribute	Value
INDEX	6006 _h
Name	External doors status word import
Object code	ARRAY
Data type	UNSIGNED8
Category	Optional

Table 10 – Entry description

Attribute	Value
Sub-index	00 _h
Description	Highest subindex supported
Entry category	Mandatory
Access	ro
PDO mapping	No
Value range	01 _h to 21 _h
Default value	No
Sub-index	01 _h
Description	Door status UIC vehicle 1
Entry category	Optional
Access	ro
PDO mapping	Optional
Value range	see Table 46
Default value	No
to	
Sub-index	20 _h
Description	Door status UIC vehicle 32
Entry category	Optional
Access	ro
PDO mapping	No
Value range	Optional
Default value	No
Sub-index	21 _h
Description	Door status train
Entry category	Optional
Access	ro
PDO mapping	No
Value range	Optional
Default value	No

8.3.4.2 Object 6001_h: External door command

This object shall provide the first control word for the external doors. Figure 11 specifies the object structure and Table 11 specifies the value definition. Table 12 specifies the object description and Table 13 specifies the entry description.

NOTE The object shall correspond to R3-telegram octet 20, which is defined in UIC 556.

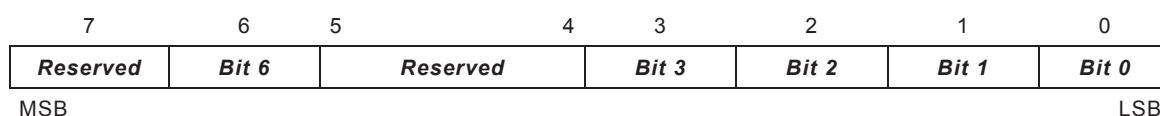

Figure 11 – Object structure

Table 11 – Value definition

Bit	Value	Value definition
Bit 0	0	Inactive
	1	All doors close
Bit 1	0	Inactive
	1	Interrupt door closing
Bit 2	0	Release all left doors
	1	Lock all left doors
Bit 3	0	Release all right doors
	1	Lock all right doors
Bit 6	0	Not extend footstep
	1	Extend footstep
Reserved		Reserved (shall be ignored)

Table 12 – Object description

Attribute	Value
INDEX	6001 _h
Name	External door command 1
Object code	Variable
Data type	UNSIGNED8
Category	Optional

Table 13 – Entry description

Attribute	Value
Sub-index	00 _h
Access	ro
PDO mapping	Optional
Value range	See value definition
Default value	No

9 CANopen network management

9.1 Content

This clause provides the network management for CANopen-based consist networks. The definition of the network management includes the definition of the network startup behavior as well as definitions that are related to networks that operate without NMT master, networks with one CANopen device capable of the NMT master mode. These definitions are intended to be an add-on to the CANopen application layer and communication profile provided in EN 50325-4.

CANopen consist networks with more than one CANopen device capable of the NMT master mode (NMT flying master) are not in the scope of this specification. However such networks are not excluded.

NOTE With regard to an increased availability, within CANopen-based consist networks several CANopen devices with NMT master functionality may reside. As at a certain time, only one active NMT master is allowed, mechanisms are required, that enable overtaking of the NMT master functionality from one CANopen device to another. These services are part of the CANopen flying master functionality that is described in CiA 302.

9.2 CANopen NMT slave functionality

Devices operated in a CANopen-based consist network shall provide CANopen NMT-slave functionality and shall implement all mandatory communication services and protocols as well as all mandatory objects as specified in EN 50325-4. The minimum communication objects are defined in 7.6.

Optionally further communication objects may be supported. In addition the CANopen manager functionality may be supported.

9.3 CANopen manager functionality

9.3.1 General

This subclause provides the definition of the CANopen manager functionality for CANopen-based consist networks.

Besides the application process several different additional functionalities exist in a CANopen network. These functionalities are referred to by different terms. This subclause is intended to clarify these terms.

Within a distributed system the application process is divided into several parts running on different CANopen devices. From the applications point of view usually one CANopen device is responsible for the control of the system. This CANopen device is called *application master*.

From the network's point of view there are several additional functionalities, which not directly deal with the application but provide application-supporting functions. These additional functionalities are based on a master/slave, client/server or producer/consumer relationship.

As it is common to combine several of the additional functionalities in one CANopen device the term CANopen Manager is introduced.

A CANopen device is referred to as CANopen manager, if it provides the NMT master functionality and at least one of the functionalities SDO manager or Configuration manager.

9.3.1.1 NMT master

The network management (NMT) provides services for controlling the network behavior of CANopen devices as defined in EN 50325-4. All CANopen devices that are participating in a CANopen consist network, referred to as NMT slaves, are controlled by services provided by an NMT master. Usually the NMT master application is also part of the application master.

The device, in which the NMT master functionality resides is a fully CANopen device. In addition to the NMT master functionality it supports all functions and objects, which are indicated as mandatory in EN 50325-4.

9.3.1.2 Flying master

The flying master mechanism provides services for a hot stand-by NMT master within a CANopen network. The flying master is an optional functionality within a CANopen device. The CANopen device implementing the flying master implements the NMT master functionality. The definition of the responsibilities, functionalities, and services of the flying master is not in the scope of this document.

NOTE Definitions for the flying master functionality are provided in CiA 302.

9.3.1.3 SDO manager

The SDO manager is an optional functionality, responsible for handling of the dynamic establishment of SDO connections. If an SDO manager is present in a CANopen network it resides together with the NMT master on the same CANopen device.

NOTE Definitions for the SDO manager functionality are provided in CiA 302.

9.3.1.4 Configuration manager

The Configuration Manager is an optional functionality, which provides mechanisms for configuration of CANopen devices in a CANopen network during boot-up. The mechanisms are called Configuration Management (CMT). If the Configuration Manager is present in a CANopen network it resides together with the NMT master on the same CANopen device.

NOTE Definitions for the Configuration manager functionality are provided in CiA 302.

9.3.1.5 SYNC producer

The SYNC producer is an optional functionality, which is responsible for transmitting the SYNC object. It may reside on any CANopen device in the CANopen-based consist network. The relevant CANopen object dictionary entries are defined in EN 50325-4.

9.3.1.6 TIME producer

The TIME producer is an optional functionality, which is responsible for transmitting the TIME STAMP object. It may reside on any CANopen device in the CANopen-based consist network. The time stamp object is defined in EN 50325-4.

9.3.1.7 LSS master

The layer setting services (LSS) provide services for configuring layer 2 (bit timing) and NMT (CANopen Node-ID) via CAN. The LSS master configures LSS slaves by executing LSS services. The definition of the layer setting services is not in the scope of this document.

NOTE Definitions for the layer setting services are provided in CiA 305.

9.3.2 Object dictionary usage

Several objects relating to the configuration and validation of CANopen devices are of object type ARRAY. The sub-index of the entries of such an ARRAY corresponds to the CANopen Node-ID of the CANopen device. Those objects may have less than 127 entries. In that case all of these objects shall have the very same set of supported entries. Sub-index 00_h shall provide the highest sub-index supported at this index. Related object description in Clause 9 refer to this as *SupportedNodeID* condition.

NOTE For a CANopen manager to be universally usable it is recommended to support the full sub-index range of 01_h to 7F_h.

9.3.3 Redundant networks

Redundant networks are required for high-availability applications. Redundant networks consist of at least two CAN lines. This allows communication between CANopen devices, in case of a single failure within the physical interconnection between the CANopen devices. The definition of the responsibilities, functionalities, and services for redundant networks is not in the scope of this document.

NOTE Definitions for setting up redundant CANopen networks are provided in CiA 302.

9.4 CANopen NMT start-up

9.4.1 NMT startup

CANopen managers shall behave according to the NMT slave state machine as defined in EN 50325-4. Before transition from NMT state *Pre-operational* to NMT state *Operational* of the CANopen manager, all assigned NMT slaves shall be booted. The main flow chart for the procedure is defined in Figure 12 and Figure 13. The simplest flow chart for the startup is defined in Figure 14.

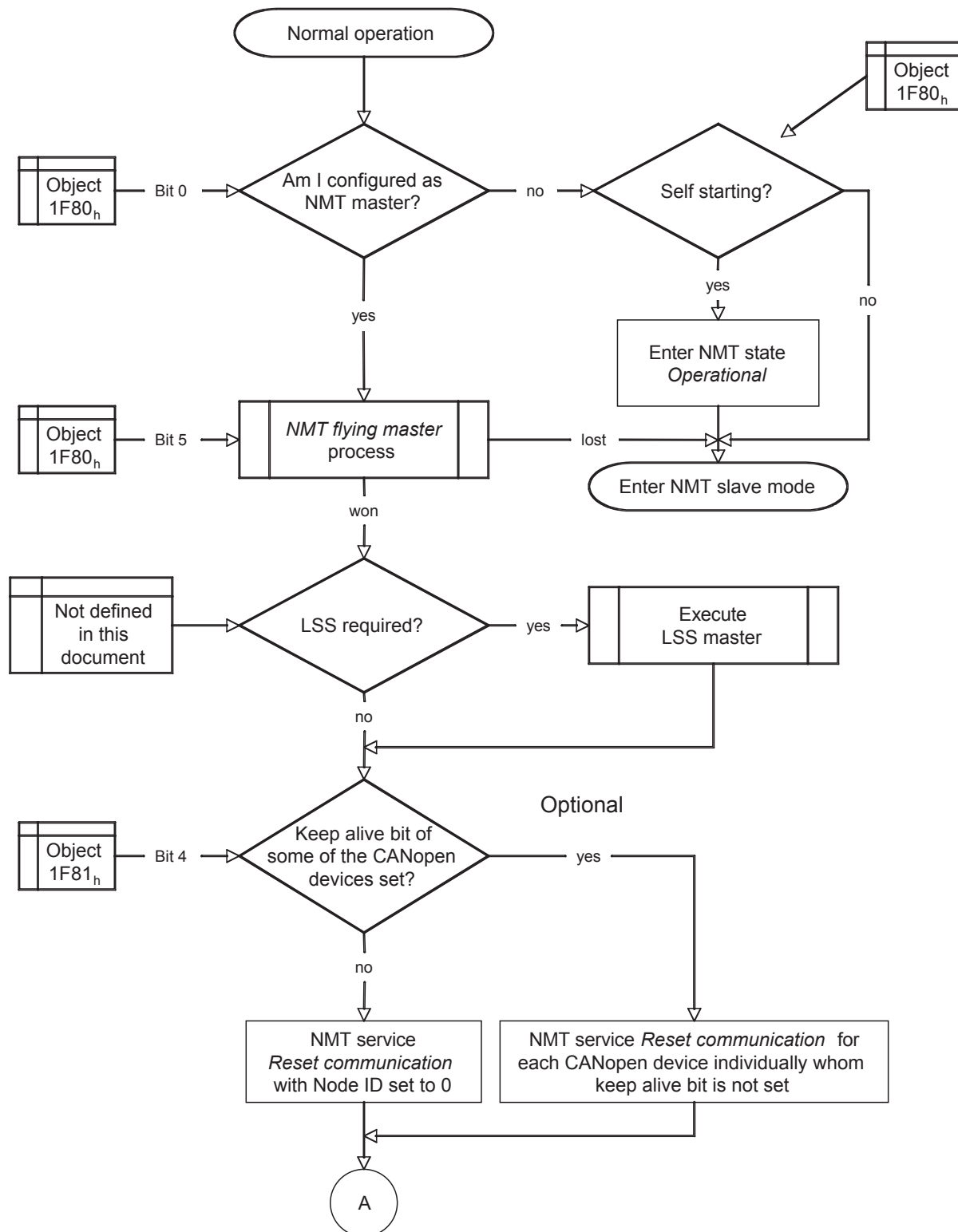


Figure 12 – NMT startup, part 1