

# SLOVENSKI STANDARD

## SIST EN 61158-6-2:2012

01-september-2012

Nadomešča:

SIST EN 61158-6-2:2008

---

### Industrijska komunikacijska omrežja - Specifikacije za procesna vodila - 600. del: Specifikacija protokola aplikacijskega nivoja (IEC 61158-6-2:2010)

Industrial communication networks - Fieldbus specifications - Part 600: Application layer protocol specification (IEC 61158-6-2:2010)

Industrielle Kommunikationsnetze - Feldbusse - Teil 600: Protokollspezifikation des Application Layer (Anwendungsschicht) (IEC 61158-6-2:2010)

Réseaux de communication industriels - Spécifications des bus de terrain - Partie 6-2: Spécification des protocoles des couches d'application - Elements de type 2 (CEI 61158-6-2:2010)

**Ta slovenski standard je istoveten z: EN 61158-6-2:2012**

#### **ICS:**

25.040.40	Merjenje in krmiljenje industrijskih postopkov	Industrial process measurement and control
35.100.70	Uporabniški sloj	Application layer
35.110	Omreževanje	Networking

**SIST EN 61158-6-2:2012**

**en**

**iTeh STANDARD PREVIEW**  
**(standards.iteh.ai)**

[SIST EN 61158-6-2:2012](#)

<https://standards.iteh.ai/catalog/standards/sist/8c4d0bdb-037e-47f6-9d9f-36a33dc56269/sist-en-61158-6-2-2012>

EUROPEAN STANDARD  
NORME EUROPÉENNE  
EUROPÄISCHE NORM

**EN 61158-6-2**

June 2012

ICS 25.040.40; 35.100.70; 35.110

Supersedes EN 61158-6-2:2008

English version

**Industrial communication networks -  
Fieldbus specifications -  
Part 6-2: Application layer protocol specification -  
Type 2 elements  
(IEC 61158-6-2:2010)**

Réseaux de communication industriels -  
Spécifications des bus de terrain -  
Partie 6-2: Spécification des protocoles  
des couches d'application -  
Éléments de type 2  
(CEI 61158-6-2:2010)

Industrielle Kommunikationsnetze -  
Feldbusse -  
Teil 6-2: Protokollspezifikation des  
Application Layer (Anwendungsschicht) -  
Typ 2-Elemente  
(IEC 61158-6-2:2010)

**ITeh STANDARD PREVIEW  
(standards.iteh.ai)**

[SIST EN 61158-6-2:2012](https://standards.iteh.ai/catalog/standards/sist/8c4d0bdb-037e-47f6-9d9f-61158-6-2)

<https://standards.iteh.ai/catalog/standards/sist/8c4d0bdb-037e-47f6-9d9f-61158-6-2>

This European Standard was approved by CENELEC on 2012-03-28. 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, 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 65C/607/FDIS, future edition 2 of IEC 61158-6-2, prepared by SC 65C, "Industrial networks", of IEC/TC 65, "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 61158-6-2:2012.

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) 2012-12-28
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2015-03-28

This document supersedes EN 61158-6-2:2008.

EN 61158-6-2:2012 includes the following significant technical changes with respect to EN 61158-6-2:2008:

- Clause 2 and Bibliography: update of normative and bibliographic references;
- 3.4: update of abbreviations;
- 4.1.2.1: update list of service request/response PDUs (Time Sync and Parameter ASEs/objects);
- 4.1.8.6: update of Time Sync ASE/object;
- new 4.1.8.7: new Parameter ASE/object;
- 4.1.9.4: minor update of a common parameter;
- 4.1.10.2: update/add object and services codes for Time Sync and Parameter ASEs;
- 4.1.11: minor updates of error codes;
- new 11.7: new QoS specification.

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.

## Endorsement notice

The text of the International Standard IEC 61158-6-2:2010 was approved by CENELEC as a European Standard without any modification.

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

- |                     |  |
|---------------------|--|
| IEC 61131-3         | NOTE Harmonized as EN 61131-3.                         |
| IEC/TR 61158-1:2010 | NOTE Harmonized as CLC/TR 61158-1:2010 (not modified). |

## Annex ZA (normative)

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

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

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

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC 61158-3-2	2007	Industrial communication networks - Fieldbus specifications - Part 3-2: Data-link layer service definition - Type 2 elements	EN 61158-3-2	2008
IEC 61158-4-2	2010	Industrial communication networks - Fieldbus specifications - Part 4-2: Data-link layer protocol specification - Type 2 elements	EN 61158-4-2	2012
IEC 61158-5-2	2010	Industrial communication networks - Fieldbus specifications - Part 5-2: Application layer service definition - Type 2 elements	EN 61158-5-2	2012
IEC 61588	2009	Precision clock synchronization protocol for networked measurement and control systems	-	-
IEC 61784-3-2	-	Industrial communication networks - Profiles - Part 3-2: Functional safety fieldbuses Additional specifications for CPF 2	EN 61784-3-2	-
IEC 62026-3	2008	Low-voltage switchgear and controlgear - Controller-device interfaces (CDIs) - Part 3: DeviceNet	EN 62026-3	2009
ISO/IEC 7498-1	-	Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model	-	-
ISO/IEC 8802-3	-	Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications	-	-
ISO/IEC 8824	Series	Information technology - Abstract Syntax Notation One (ASN.1)	-	-
ISO/IEC 8825-1	-	Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)	-	-
ISO/IEC 9545	-	Information technology - Open Systems Interconnection - Application Layer structure	-	-
ISO/IEC 10646	-	Information technology - Universal multiple-octet coded character set (UCS)	-	-

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
ISO/IEC 10731	-	Information technology - Open Systems Interconnection - Basic reference model - Conventions for the definition of OSI services	-	-
ISO 11898	1993	Road vehicles - Interchange of digital information - Controller area network (CAN) for high-speed communication	-	-
IEEE 802.1Q	-	IEEE Standard for Local and Metropolitan Area Networks - Virtual Bridged Local Area Networks	-	-
IEEE 802.1D	-	IEEE Standard for Local and Metropolitan Area Networks - Media Access Control (MAC) Bridges	-	-
IETF RFC 791	-	Internet Protocol - DARPA Internet Program Protocol Specification	-	-
IETF RFC 1122	-	Requirements for Internet Hosts - Communication Layers	-	-
IETF RFC 2475	-	An architecture for differentiated services	-	-
IETF RFC 1759	-	Printer MIB	-	-
IETF RFC 1035	-	Domain Names - Implementation and Specification	-	-

## iTeh STANDARD PREVIEW (standards.iteh.ai)

[SIST EN 61158-6-2:2012](#)

<https://standards.iteh.ai/catalog/standards/sist/8c4d0bdb-037e-47f6-9d9f-36a33dc56269/sist-en-61158-6-2-2012>



IEC 61158-6-2

Edition 2.0 2010-08

# INTERNATIONAL STANDARD

---

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

[SIST EN 61158-6-2:2012  
https://standards.iteh.ai/catalog/standards/sist/8c4d0bdb-037e-47f6-9d9f-36a33dc56269/sist-en-61158-6-2-2012](https://standards.iteh.ai/catalog/standards/sist/8c4d0bdb-037e-47f6-9d9f-36a33dc56269/sist-en-61158-6-2-2012)

INTERNATIONAL  
ELECTROTECHNICAL  
COMMISSION

PRICE CODE

**XL**

---

ICS 25.04.40; 35.100.70; 35.110

ISBN 978-2-88912-118-2

## CONTENTS

FOREWORD.....	12
INTRODUCTION.....	14
1 Scope.....	15
1.1 General.....	15
1.2 Specifications.....	15
1.3 Conformance.....	16
2 Normative references.....	16
3 Terms, definitions, symbols, abbreviations and conventions.....	17
3.1 Terms and definitions from other ISO/IEC standards.....	17
3.2 Terms and definitions from IEC 61158-5-2.....	18
3.3 Additional terms and definitions.....	18
3.4 Abbreviations and symbols.....	24
3.5 Conventions.....	24
3.6 Conventions used in state machines.....	29
4 Abstract syntax.....	30
4.1 FAL PDU abstract syntax.....	30
4.2 Data abstract syntax specification.....	120
4.3 Encapsulation abstract syntax.....	124
5 Transfer syntax.....	138
5.1 Compact encoding.....	138
5.2 Data type reporting.....	145
6 Structure of FAL protocol state machines.....	150
7 AP-Context state machine.....	150
7.1 Overview.....	150
7.2 Connection object state machine.....	150
8 FAL service protocol machine (FSPM).....	160
8.1 General.....	160
8.2 Primitive definitions.....	160
8.3 Parameters of primitives.....	164
8.4 FSPM state machines.....	165
9 Application relationship protocol machines (ARPMs).....	166
9.1 General.....	166
9.2 Connection-less ARPM (UCMM).....	166
9.3 Connection-oriented ARPMs (transports).....	176
10 DLL mapping protocol machine 1 (DMPM 1).....	247
10.1 General.....	247
10.2 Link producer.....	247
10.3 Link consumer.....	248
10.4 Primitive definitions.....	248
10.5 DMPM state machine.....	250
10.6 Data-link Layer service selection.....	251
11 DLL mapping protocol machine 2 (DMPM 2).....	251
11.1 General.....	251
11.2 Mapping of UCMM PDUs.....	252
11.3 Mapping of transport class 0 and class 1 PDUs.....	257



11.4 Mapping of transport class 2 and class 3 PDU's .....	258
11.5 Mapping of transport classes 4 to 6 .....	258
11.6 IGMP Usage .....	258
11.7 Quality of Service (QoS) for CP 2/2 messages .....	259
11.8 Management of an encapsulation session .....	262
12 DLL mapping protocol machine 3 (DMPM 3) .....	263
Bibliography .....	264
Figure 1 – Attribute table format and terms .....	25
Figure 2 – Service request/response parameter .....	25
Figure 3 – Example of an STD .....	29
Figure 4 – Network connection parameters .....	49
Figure 5 – Time tick .....	51
Figure 6 – Connection establishment time-out .....	53
Figure 7 – Transport Class Trigger attribute .....	86
Figure 8 – CP2/3_initial_comm_characteristics attribute format .....	89
Figure 9 – Segment type .....	97
Figure 10 – Port segment .....	98
Figure 11 – Logical segment encoding .....	100
Figure 12 – Extended network segment .....	104
Figure 13 – Encapsulation message .....	124
Figure 14 – FixedLengthBitString compact encoding bit placement rules .....	142
Figure 15 – Example compact encoding of a SWORD FixedLengthBitString .....	142
Figure 16 – Example compact encoding of a WORD FixedLengthBitString .....	142
Figure 17 – Example compact encoding of a DWORD FixedLengthBitString .....	143
Figure 18 – Example compact encoding of a LWORD FixedLengthBitString .....	143
Figure 19 – Example 2 of formal encoding of a structure type specification .....	147
Figure 20 – Example of abbreviated encoding of a structure type specification .....	148
Figure 21 – Example 1 of formal encoding of an array type specification .....	148
Figure 22 – Example 2 of formal encoding of an array type specification .....	149
Figure 23 – Example 1 of abbreviated encoding of an array type specification .....	149
Figure 24 – Example 2 of abbreviated encoding of an array type specification .....	150
Figure 25 – I/O Connection object state transition diagram .....	151
Figure 26 – Bridged Connection object state transition diagram .....	155
Figure 27 – Explicit Messaging Connection object state transition diagram .....	157
Figure 28 – State transition diagram of UCMM client .....	169
Figure 29 – State transition diagram of high-end UCMM server .....	170
Figure 30 – State transition diagram of low-end UCMM server .....	172
Figure 31 – Sequence diagram for a UCMM with one outstanding message .....	174
Figure 32 – Sequence diagram for a UCMM with multiple outstanding messages .....	175
Figure 33 – TPDU buffer .....	176
Figure 34 – Data flow diagram using a client transport class 0 and server transport class 0 .....	179
Figure 35 – Sequence diagram of data transfer using transport class 0 .....	179

Figure 36 – Class 0 client STD .....	180
Figure 37 – Class 0 server STD .....	181
Figure 38 – Data flow diagram using client transport class 1 and server transport class 1 .....	182
Figure 39 – Sequence diagram of data transfer using client transport class 1 and server transport class 1 .....	183
Figure 40 – Class 1 client STD .....	185
Figure 41 – Class 1 server STD .....	186
Figure 42 – Data flow diagram using client transport class 2 and server transport class 2 .....	188
Figure 43 – Diagram of data transfer using client transport class 2 and server transport class 2 <i>without</i> returned data .....	189
Figure 44 – Sequence diagram of data transfer using client transport class 2 and server transport class 2 <i>with</i> returned data .....	190
Figure 45 – Class 2 client STD .....	191
Figure 46 – Class 2 server STD .....	193
Figure 47 – Data flow diagram using client transport class 3 and server transport class 3 .....	196
Figure 48 – Sequence diagram of data transfer using client transport class 3 and server transport class 3 <i>without</i> returned data .....	197
Figure 49 – Sequence diagram of data transfer using client transport class 3 and server transport class 3 <i>with</i> returned data .....	198
Figure 50 – Class 3 client STD .....	200
Figure 51 – Class 3 server STD .....	203
Figure 52 – Data flow diagram using transport classes 4 and 5 .....	205
Figure 53 – Sequence diagram of message exchange using transport classes 4 and 5 .....	206
Figure 54 – Sequence diagram of messages overwriting each other .....	207
Figure 55 – Sequence diagram of queued message exchange using transport classes 4 and 5 .....	208
Figure 56 – Sequence diagram of retries using transport classes 4 and 5 .....	208
Figure 57 – Sequence diagram of idle traffic using transport classes 4 and 5 .....	209
Figure 58 – Classes 4 and 5 basic structure .....	210
Figure 59 – Class 6 basic structure .....	211
Figure 60 – Classes 4 to 6 general STD .....	212
Figure 61 – Class 4 sender STD .....	214
Figure 62 – Class 4 receiver STD .....	217
Figure 63 – Sequence diagram of three fragments using transport class 5 .....	220
Figure 64 – Sequence diagram of fragmentation with retries using transport class 5 .....	221
Figure 65 – Sequence diagram of two fragments using transport class 5 .....	221
Figure 66 – Sequence diagram of aborted message using transport class 5 .....	222
Figure 67 – Class 5 sender STD .....	223
Figure 68 – Class 5 receiver STD .....	226
Figure 69 – Data flow diagram for transport class 6 .....	231
Figure 70 – Sequence diagram of message exchange using transport class 6 .....	233
Figure 71 – Sequence diagram of retries using transport class 6 .....	233
Figure 72 – Sequence diagram of idle traffic using transport class 6 .....	234

Figure 73 – Sequence diagram of request overwriting null .....	235
Figure 74 – Sequence diagram of response overwriting ACK of null.....	235
Figure 75 – Sequence diagram of three fragments using transport class 6.....	236
Figure 76 – Sequence diagram of fragmentation with retries using transport class 6.....	237
Figure 77 – Sequence diagram of two fragments using transport class 6 .....	237
Figure 78 – Sequence diagram of aborted fragmented sequence using transport class 6 .....	238
Figure 79 – Class 6 client STD .....	239
Figure 80 – Class 6 server STD .....	242
Figure 81 – Data flow diagram for a link producer and consumer .....	247
Figure 82 – State transition diagram for a link producer .....	250
Figure 83 – State transition diagram for a link consumer.....	251
Figure 84 – DS field in the IP header .....	261
Figure 85 – IEEE 802.1Q tagged frame.....	261
Table 1 – Get_Attribute_All response service rules .....	26
Table 2 – Example class level object/service specific response data of Get_Attribute_All .....	26
Table 3 – Example Get_Attribute_All data array method .....	27
Table 4 – Set_Attribute_All request service rules .....	28
Table 5 – Example Set_Attribute_All attribute ordering method.....	28
Table 6 – Example Set_Attribute_All data array method.....	28
Table 7 – State event matrix format.....	30
Table 8 – Example state event matrix.....	30
Table 9 – UCMM_PDU header format .....	33
Table 10 – UCMM command codes.....	33
Table 11 – Transport class 0 header .....	34
Table 12 – Transport class 1 header .....	35
Table 13 – Transport class 2 header .....	35
Table 14 – Transport class 3 header .....	35
Table 15 – Classes 4 to 6 header format.....	35
Table 16 – Real-time data header – exclusive owner .....	36
Table 17 – Real-time data header– redundant owner .....	37
Table 18 – Forward_Open request format .....	39
Table 19 – Forward_Open_Good response format .....	40
Table 20 – Forward_Open_Bad response format .....	40
Table 21 – Large_Forward_Open request format .....	41
Table 22 – Large_Forward_Open_Good response format .....	42
Table 23 – Large_Forward_Open_Bad response format.....	42
Table 24 – Forward_Close request format .....	43
Table 25 – Forward_Close_Good response format.....	43
Table 26 – Forward_Close_Bad response format.....	44
Table 27 – Unconnected_Send request format.....	45
Table 28 – Unconnected_Send_Good response format.....	45

Table 29 – Unconnected_Send_Bad response format .....	46
Table 30 – Get_Connection_Data request format .....	46
Table 31 – Get_Connection_Data response format .....	47
Table 32 – Search_Connection_Data request format .....	47
Table 33 – Get_Object_Owner request format .....	48
Table 34 – Forward_Open_Good response format .....	48
Table 35 – Time-out multiplier .....	51
Table 36 – Time tick units .....	51
Table 37 – Selection of connection ID .....	55
Table 38 – Transport class, trigger and Is_Server format .....	55
Table 39 – MR_Request_Header format .....	56
Table 40 – MR_Response_Header format .....	56
Table 41 – Structure of Get_Attribute_All_ResponsePDU body .....	57
Table 42 – Structure of Set_Attribute_All_RequestPDU body .....	57
Table 43 – Structure of Get_Attribute_List_RequestPDU body .....	57
Table 44 – Structure of Get_Attribute_List_ResponsePDU body .....	57
Table 45 – Structure of Set_Attribute_List_RequestPDU body .....	57
Table 46 – Structure of Set_Attribute_List_ResponsePDU body .....	58
Table 47 – Structure of Reset_RequestPDU body .....	58
Table 48 – Structure of Reset_ResponsePDU body .....	58
Table 49 – Structure of Start_RequestPDU body .....	58
Table 50 – Structure of Start_ResponsePDU body .....	58
Table 51 – Structure of Stop_RequestPDU body .....	59
Table 52 – Structure of Stop_ResponsePDU body .....	59
Table 53 – Structure of Create_RequestPDU body .....	59
Table 54 – Structure of Create_ResponsePDU body .....	59
Table 55 – Structure of Delete_RequestPDU body .....	59
Table 56 – Structure of Delete_ResponsePDU body .....	59
Table 57 – Structure of Get_Attribute_Single_ResponsePDU body .....	60
Table 58 – Structure of Set_Attribute_Single_RequestPDU body .....	60
Table 59 – Structure of Set_Attribute_Single_ResponsePDU body .....	60
Table 60 – Structure of Find_Next_Object_Instance_RequestPDU body .....	60
Table 61 – Structure of Find_Next_Object_Instance_ResponsePDU body .....	60
Table 62 – Structure of Apply_Attributes_RequestPDU body .....	61
Table 63 – Structure of Apply_Attributes_ResponsePDU body .....	61
Table 64 – Structure of Save_RequestPDU body .....	61
Table 65 – Structure of Save_ResponsePDU body .....	61
Table 66 – Structure of Restore_RequestPDU body .....	61
Table 67 – Structure of Restore_ResponsePDU body .....	61
Table 68 – Structure of Group_Sync_RequestPDU body .....	62
Table 69 – Structure of Group_Sync_ResponsePDU body .....	62
Table 70 – Identity object class attributes .....	62
Table 71 – Identity object instance attributes .....	62

Table 72 – Identity object bit definitions for status instance attribute .....	63
Table 73 – Bits 4 – 7 of status instance attribute .....	64
Table 74 – Class level object/service specific response data of Get_Attribute_All .....	64
Table 75 – Instance level object/service specific response data of Get_Attribute_All .....	65
Table 76 – Modified instance level object/service specific response data of Get_Attribute_All .....	65
Table 77 – Object-specific parameter for Reset .....	66
Table 78 – Message Router object class attributes .....	66
Table 79 – Message Router object instance attributes .....	66
Table 80 – Class level object/service specific response data of Get_Attribute_All .....	67
Table 81 – Instance level object/service specific response data of Get_Attribute_All .....	67
Table 82 – Assembly object class attributes .....	67
Table 83 – Assembly object instance attributes .....	67
Table 84 – Acknowledge Handler object class attributes .....	68
Table 85 – Acknowledge Handler object instance attributes .....	69
Table 86 – Structure of Add_AckData_Path_RequestPDU body .....	69
Table 87 – Structure of Remove_AckData_Path_RequestPDU body .....	69
Table 88 – Time Sync object instance attributes .....	70
Table 89 – ClockIdentity encoding for different network implementations .....	73
Table 90 – ClockClass values .....	73
Table 91 – TimeAccuracy values .....	74
Table 92 – TimePropertyFlags bit values .....	74
Table 93 – TimeSource values .....	75
Table 94 – Types of Clock .....	75
Table 95 – Parameter object class attributes .....	75
Table 96 – Parameter Class Descriptor bit values .....	76
Table 97 – Parameter object instance attributes .....	77
Table 98 – Semantics of Descriptor Instance attribute .....	77
Table 99 – Minimum and Maximum Value semantics .....	79
Table 100 – Scaling Formula attributes .....	80
Table 101 – Scaling links .....	80
Table 102 – Structure of Get_Enum_String_RequestPDU body .....	81
Table 103 – Structure of Get_Enum_String_ResponsePDU body .....	81
Table 104 – Enumerated strings Type versus Parameter data type .....	81
Table 105 – Connection Manager object class attributes .....	82
Table 106 – Connection Manager object instance attributes .....	82
Table 107 – Connection object class attributes .....	83
Table 108 – Connection object instance attributes .....	84
Table 109 – Values assigned to the state attribute .....	85
Table 110 – Values assigned to the instance_type attribute .....	85
Table 111 – Possible values within Direction Bit .....	86
Table 112 – Possible values within Production Trigger Bits .....	86
Table 113 – Possible values within Transport Class Bits .....	87

Table 114 – Transport Class_Trigger attribute .....	88
Table 115 – Values defined for the CP2/3_produced_connection_id attribute .....	88
Table 116 – Values defined for the CP2/3_consumed_connection_id attribute .....	89
Table 117 – Values for the Initial Production Characteristics nibble .....	90
Table 118 – Values for the Initial Consumption Characteristics nibble.....	91
Table 119 – Values for the watchdog_timeout_action.....	94
Table 120 – Structure of Connection_Bind_RequestPDU body.....	95
Table 121 – Object specific status for Connection_Bind service.....	96
Table 122 – Structure of Producing_Application_Lookup_RequestPDU body .....	96
Table 123 – Structure of Producing_Application_Lookup_ResponsePDU body .....	96
Table 124 – Producing_Application_Lookup Service status codes .....	96
Table 125 – Possible port segment examples .....	98
Table 126 – TCP/IP link address examples.....	99
Table 127 – Electronic key segment format.....	101
Table 128 – Logical segments examples.....	102
Table 129 – Network segments.....	102
Table 130 – Extended subtype definitions .....	104
Table 131 – Data segment.....	104
Table 132 – ANSI_Extended_Symbol segment.....	105
Table 133 – Addressing categories.....	105
Table 134 – Class code ID ranges .....	106
Table 135 – Attribute ID ranges.....	106
Table 136 – Service code ranges.....	106
Table 137 – Class codes.....	107
Table 138 – Reserved class attributes for all object class definitions .....	107
Table 139 – Common services list .....	108
Table 140 – Acknowledge Handler object specific services list .....	109
Table 141 – Parameter object specific services list.....	109
Table 142 – Services specific to Connection Manager .....	109
Table 143 – Services specific to Connection object.....	109
Table 144 – Device type numbering .....	110
Table 145 – Connection Manager service request error codes .....	111
Table 146 – General status codes.....	116
Table 147 – Identity object status codes .....	118
Table 148 – Encapsulation header.....	125
Table 149 – Encapsulation command codes .....	125
Table 150 – Encapsulation status codes .....	126
Table 151 – Options flags .....	127
Table 152 – Nop request encapsulation header .....	127
Table 153 – RegisterSession request encapsulation header .....	127
Table 154 – RegisterSession request data portion .....	128
Table 155 – Options flags .....	128
Table 156 – RegisterSession reply encapsulation header .....	128

Table 157 – RegisterSession reply data portion .....	128
Table 158 – UnRegisterSession request encapsulation header .....	129
Table 159 – ListServices request encapsulation header .....	129
Table 160 – ListServices reply encapsulation header .....	130
Table 161 – ListServices reply data portion.....	130
Table 162 – Service type codes .....	130
Table 163 – Communications capability flags.....	130
Table 164 – ListIdentity request encapsulation header.....	131
Table 165 – ListIdentity reply encapsulation header.....	131
Table 166 – ListIdentity reply data portion .....	132
Table 167 – ListInterfaces request encapsulation header.....	132
Table 168 – ListInterfaces reply encapsulation header.....	133
Table 169 – SendRRData request encapsulation header .....	133
Table 170 – SendRRData request data portion .....	133
Table 171 – SendRRData reply encapsulation header .....	134
Table 172 – SendUnitData request encapsulation header .....	134
Table 173 – SendUnitData request data portion.....	135
Table 174 – Common packet format.....	135
Table 175 – Address and data item structure.....	135
Table 176 – Address type ID's .....	135
Table 177 – Data type ID's .....	136
Table 178 – Null address type .....	136
Table 179 – Connected address type .....	136
Table 180 – Sequenced address type .....	137
Table 181 – UCMM data type.....	137
Table 182 – Connected data type .....	137
Table 183 – Sockaddr info items.....	137
Table 184 – BOOLEAN encoding .....	139
Table 185 – Example compact encoding of a BOOL value .....	139
Table 186 – Encoding of SignedInteger values .....	139
Table 187 – Example compact encoding of a SignedInteger value .....	139
Table 188 – UnsignedInteger values .....	139
Table 189 – Example compact encoding of an UnsignedInteger.....	140
Table 190 – FixedLengthReal values .....	140
Table 191 – Example compact encoding of a REAL value.....	140
Table 192 – Example compact encoding of a LREAL value.....	140
Table 193 – FixedLengthReal values .....	140
Table 194 – STRING value .....	141
Table 195 – STRING2 value .....	141
Table 196 – STRINGN value.....	141
Table 197 – SHORT_STRING value .....	141
Table 198 – Example compact encoding of a STRING value.....	141
Table 199 – Example compact encoding of a STRING2 value.....	142