

DIN EN 61158-6-20

ICS 35.100.70; 35.160

Ersatz für
DIN EN 61158-6-20:2008-09
Siehe Anwendungsbeginn

**Industrielle Kommunikationsnetze –
Feldbusse –
Teil 6-20: Protokollspezifikation des Application Layer
(Anwendungsschicht) – Typ 20-Elemente
(IEC 61158-6-20:2010);
Englische Fassung EN 61158-6-20:2012, nur auf CD-ROM**

Industrial communication networks –

Fieldbus specifications –

Part 6-20: Application layer protocol specification – Type 20 elements
(IEC 61158-6-20:2010);

English version EN 61158-6-20:2012, only on CD-ROM

Réseaux de communication industriels –

Spécifications des bus de terrain –

Partie 6-20: Spécification des protocoles des couches d'application – Eléments de type 20
(CEI 61158-6-20:2010);

Version anglaise EN 61158-6-20:2012, seulement en CD-ROM

Gesamtumfang 60 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2012-03-28 angenommene Europäische Norm als DIN-Norm ist 2013-03-01.

Für DIN EN 61158-6-20:2008-09 gilt eine Übergangsfrist bis zum 2015-03-28.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN IEC 61158-600:2009-06.

Für dieses Dokument ist das nationale Arbeitsgremium K 956 „Feldbus“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom SC 65C „Industrial networks“ 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 eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

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

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen 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 61158-6-20 veröffentlicht. Zu deren 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.

Einführung

Ein Feldbus ist ein digitaler serieller Datenbus für die Kommunikation zwischen Geräten der industriellen Automatisierungstechnik wie beispielsweise Messeinrichtungen, Stellgliedern, Reglern und speicherprogrammierbaren Steuerungen. Die Normen der Reihe DIN EN 61158 legen eine Anzahl von Feldbus-Protokolltypen fest. Die vorliegende Norm legt das Protokoll des Application Layer für Typ 20 fest.

Änderungen

Gegenüber DIN EN 61158-6-20:2008-09 wurden folgende Änderungen vorgenommen:

- a) Identity FAL-PDU in 5.3.2 verbessert;

b) die read device-Variablen mit der Status-FAL-PDU in 5.3.9 verbessert.

Frühere Ausgaben

DIN V 19245-1: 1988-01
DIN EN 61158-6: 2005-05
DIN EN 61158-6-20: 2008-09

Nationaler Anhang NA (informativ)

3.2 Begriffe aus IEC/TR 61158-1

Es gelten die folgenden Begriffe aus IEC/TR 61158-1.

3.2.1

Anwendung

(en: application)

Funktion oder Datenstruktur, für die Daten verarbeitet oder erzeugt werden

3.2.2

Interoperabilität auf Anwendungsschicht

(en: application layer interoperability)

Fähigkeit von Anwendungseinheiten, mithilfe der Dienste der FAL koordinierte und kooperative Operationen auszuführen

3.2.3

Anwendungsobjekt

(en: application object)

Objektklasse, die den Nachrichtenaustausch über das Netzwerk und innerhalb des Netzwerkgerätes zur Laufzeit bereitstellt und verwaltet

ANMERKUNG Es können mehrere Typen von Anwendungsobjektklassen definiert werden.

3.2.4

Anwendungsprozess

(en: application process)

Teil einer verteilten Anwendung in einem Netzwerk, der sich auf einem Gerät befindet und eindeutig adressiert wird

3.2.5

Anwendungsprozesskennung

(en: application process identifier)

Kennung, die mehrere in einem Gerät verwendete Anwendungsprozesse voneinander unterscheidet

3.2.6

Anwendungsprozessobjekt

(en: application process object)

Komponente eines Anwendungsprozesses, die über eine FAL-Anwendungsbeziehung identifizierbar und zugänglich ist

ANMERKUNG Definitionen von Anwendungsprozessobjekten bestehen aus einem Satz von Werten für die Attribute ihrer Klasse (siehe Definition von „Klasse von Anwendungsprozessobjekten“). Auf die Definitionen von Anwendungsprozessobjekten kann mithilfe der Dienste des Object-Management-ASE der FAL von einem entfernten Standort aus zugegriffen werden. Object-Management-Dienste der FAL können zum Laden und Aktualisieren von Objektdefinitionen, zum Lesen von Objektdefinitionen und zur dynamischen Erstellung und Löschung von Anwendungsobjekten und deren Definitionen genutzt werden.

3.2.7

Klasse von Anwendungsprozessobjekten

(en: **application process object class**)

Klasse von Anwendungsprozessobjekten, die in Bezug auf den Satz der über das Netzwerk zugänglichen Attribute und Dienste der Anwendungsprozessobjekte definiert ist

3.2.8

Anwendungsbeziehung

(en: **application relationship**)

kooperative Verbindung zwischen zwei oder mehr Aufrufen von Anwendungseinheiten zum Zweck des Informationsaustausches und der Koordination des gemeinsamen Betriebs

ANMERKUNG Diese Beziehung wird entweder durch den Austausch von Protokolldateneinheiten der Anwendung oder als Ergebnis von Vorkonfigurationsaktivitäten aktiviert.

3.2.9

Anwendungsdienstelement für Anwendungsbeziehungen

(en: **application relationship application service element**)

Anwendungsdienstelement, das die alleinigen Mittel für den Aufbau und die Beendigung aller Anwendungsbeziehungen bereitstellt

3.2.10

Endpunkt der Anwendungsbeziehung

(en: **application relationship endpoint**)

Kontext und Verhalten einer Anwendungsbeziehung entsprechend der Sicht und Verwaltung durch einen der in die Anwendungsbeziehung eingeschlossenen Anwendungsprozesse

ANMERKUNG Jeder Anwendungsprozess, der in die Anwendungsbeziehung eingeschlossen ist, verwaltet seinen eigenen Endpunkt der Anwendungsbeziehung.

3.2.11

Attribut

(en: **attribute**)

Beschreibung eines von außen sichtbaren Merkmals oder einer von außen sichtbaren Eigenschaft eines Objekts

ANMERKUNG Die Attribute eines Objekts enthalten Informationen zu veränderlichen Teilen eines Objekts. Üblicherweise stellen sie Statusinformationen bereit oder steuern den Betrieb eines Objekts. Attribute können auch das Verhalten eines Objekts beeinflussen. Attribute werden in Klassenattribute und Instanzattribute unterteilt.

3.2.12

Verhalten

(en: **behaviour**)

Angabe, wie das Objekt auf bestimmte Ereignisse reagiert. Seine Beschreibung schließt die Beziehung zwischen Attributwerten und Diensten ein.

3.2.13

Klasse

(en: **class**)

Satz von Objekten, die allesamt den gleichen Typ von Systemkomponente repräsentieren

ANMERKUNG Eine Klasse ist eine Verallgemeinerung des Objektes; eine Vorlage für die Festlegung von Variablen und Methoden. Alle Objekte in einer Klasse sind in ihrer Form und in ihrem Verhalten identisch, enthalten aber üblicherweise unterschiedliche Daten in ihren Attributen.

3.2.14

Klassenattribute

(en: **class attributes**)

Attribut, das von allen Objekten innerhalb derselben Klasse gemeinsam genutzt wird

3.2.15**Klassencode****(en: class code)**

eindeutige Kennung, die jeder Objektklasse zugewiesen ist

3.2.16**klassenspezifischer Dienst****(en: class specific service)**

Dienst, der durch eine bestimmte Objektklasse definiert ist, eine erforderliche Funktion, die nicht von einem allgemeinen Dienst ausgeführt wird, auszuführen

ANMERKUNG Ein klassenspezifisches Objekt ist für die Objektklasse, die es definiert, eindeutig.

3.2.17**Client****(en: client)**

- a) Objekt, das die Dienste eines anderen Objekts (Servers) nutzt, um eine Aufgabe auszuführen
- b) Initiator einer Nachricht, auf die ein Server reagiert, wie die Rolle eines AR-Endpunktes, in der dieser bestätigte Dienstanforderungs-APDUs an einen einzelnen AR-Endpunkt, der als Server agiert, ausgibt

3.2.18**Übertragungspfad****(en: conveyance path)**

unidirektionaler Strom von APDUs über eine Anwendungsbeziehung

3.2.19**zyklisch****(en: cyclic)**

Benennung zur Beschreibung von Ereignissen, die sich regelmäßig wiederholen

3.2.20**zugeordnete AR****(en: dedicated AR)**

AR, die direkt vom FAL-Benutzer verwendet wird. Bei zugeordneten Anwendungsbeziehungen werden nur der FAL-Header und die Benutzerdaten übertragen.

3.2.21**Gerät****(en: device)**

physikalische Hardware-Verbindung zum Verbindungszweig. Ein Gerät kann mehr als einen Knoten umfassen.

3.2.22**Geräteprofil****(en: device profile)**

Zusammenstellung geräteabhängiger Informationen und Funktionalitäten, um Konsistenz zwischen ähnlichen Geräten desselben Gerätetyps herzustellen

3.2.23**Endpunkt****(en: endpoint)**

eine der Kommunikationseinheiten, die zu einer Verbindung gehören

3.2.24**Fehler****(en: error)**

Nichtübereinstimmung zwischen einem berechneten, beobachteten oder gemessenen Wert oder Zustand und dem festgelegten oder theoretisch richtigen Wert oder Zustand

3.2.25

Fehlercode

(en: **error code**)

Kennzeichnung eines bestimmten Fehlertyps innerhalb einer Fehlerklasse

3.2.26

Verwaltungsinformationen

(en: **management information**)

über das Netzwerk zugängliche Informationen, die die Verwaltung des Betriebs des Feldbussystems, einschließlich der Anwendungsschicht, unterstützen

ANMERKUNG Zur Verwaltung gehören Funktionen wie die Steuerung, die Überwachung und die Diagnose.

3.2.27

Netzwerk

(en: **network**)

Reihe von Knoten, die über ein bestimmtes Kommunikationsmedium miteinander verbunden sind

ANMERKUNG Die Verbindungspfade zwischen beliebigen Paaren von Knoten können Repeater, Router und Gateways einschließen.

3.2.28

vordefinierter AR-Endpunkt

(en: **pre-defined AR endpoint**)

AR-Endpunkt, der lokal innerhalb eines Gerätes ohne Nutzung des Erstellungsdienstes definiert ist

ANMERKUNG Vordefinierte ARs, die nicht bereits bestehen, werden vor ihrer Nutzung hergestellt.

3.2.29

vorher aktivierter AR-Endpunkt

(en: **pre-established AR endpoint**)

AR-Endpunkt, der während der Konfiguration der AEs, die ihre Endpunkte steuern, in einen aktiven Zustand versetzt wird

3.2.30

Server

(en: **server**)

- a) Rolle eines AREP, in der dieser eine bestätigte Dienstanwort-APDU an den Client, der die Anforderung initiiert hat, zurückgibt
- b) Objekt, das einem anderen Objekt (Client) Dienste bereitstellt

3.2.31

Dienst

(en: **service**)

Operation oder Funktion, die ein Objekt und/oder eine Objektklasse auf Aufforderung eines anderen Objekts und/oder einer anderen Objektklasse ausführt

ANMERKUNG Es ist eine Reihe von allgemeinen Diensten definiert und es sind Richtlinien für die Definition von objektspezifischen Diensten angegeben. Objektspezifische Dienste sind durch eine bestimmte Objektklasse definierte Dienste, die eine erforderliche Funktion, die nicht von einem allgemeinen Dienst ausgeführt wird, ausführen sollen.

3.3 Spezielle Begriffe für die Feldbus-Anwendungsschicht Typ 20

Für diesen Teil sind zusätzliche Begriffe definiert.

3.3.1**analoger Kanal****(en: analog channel)**

fortlaufend wechselndes elektrisches Signal, das ein Feldgerät mit dem Rest des Datenerfassungs- oder Steuerungssystems verbindet. Einige Feldgeräte unterstützen mehrere analoge Kanäle (Eingangs- oder Ausgangskanäle).

ANMERKUNG Jeder analoge Kanal überträgt eine einzelne dynamische Variable an das Feldgerät oder vom Feldgerät.

3.3.2**Broadcast-Adresse****(en: broadcast address)**

eine Broadcast-Adresse wird von einem Master verwendet, um einen Befehl an alle Geräte zu senden

ANMERKUNG Die Broadcast-Adresse ist fünf Oktette lang und weist Nullen als Werte auf.

3.3.3**busy****(en: busy)**

das Gerät ist ausgelastet und kann derzeit keinen Befehl ausführen; ein Gerät gibt seine Auslastung an, indem es den Antwortcode 32 zurückgibt, sofern dies nach der Befehlsspezifikation zulässig ist; der angeforderte Befehl wird nicht ausgeführt, wenn eine BUSY-Antwort zurückgegeben wird

3.3.4**Geräte-ID****(en: device ID)**

Seriennummer für ein Gerät

ANMERKUNG Der Hersteller muss jedem Gerät, das über die gleichen Werte für die Hersteller-ID und den Gerätetyp verfügt, einen eindeutigen Wert zuweisen.

3.3.5**Gerätetyp****(en: device type)**

vom Hersteller festgelegter Typ eines Gerätes, z. B. sein Produktname

ANMERKUNG Der Wert dieses Attributes wird vom Hersteller zugewiesen. Der Wert des Gerätetyps legt den Satz von Befehlen und Datenobjekten fest, die vom Gerät unterstützt werden. Der Hersteller muss jedem Gerätetyp einen eindeutigen Wert zuweisen.

3.3.6**Gerätevariable****(en: device variable)**

eindeutig definiertes Datenelement innerhalb eines Feldgerätes, das sich immer auf zyklische Prozessinformationen bezieht; der Wert einer Gerätevariablen ändert sich in Folge von Änderungen und Abweichungen in dem Prozess, mit dem das Gerät verbunden ist

3.3.7**dynamische Variable****(en: dynamic variable)**

Verbindung zwischen einem Prozess und einem analogen Kanal

ANMERKUNG Ein Gerät kann primäre, sekundäre, tertiäre und quartäre Variablen enthalten. Diese werden zusammen als dynamische Variablen bezeichnet.

3.3.8**langes Tag****(en: long tag)**

Zeichenfolge mit 32 ISO-Latin-1-Zeichen zur Kennzeichnung eines Feldgerätes

3.3.9

Schleifenstrom

(en: loop current)

Wert, der mit einem Milliampereometer, der in Reihe mit dem Feldgerät geschaltet ist, gemessen wird

ANMERKUNG Der Schleifenstrom ist ein analoges niederfrequentes Gleichstromsignal von 4 mA bis 20 mA, das verwendet wird, um einen einzelnen Wert zwischen dem Steuerungssystem und dem Feldgerät zu übertragen. Geräte zur Steuerung des Spannungsbetriebs verwenden „Volt (Gleichstrom)“ als Einheit für Schleifenstromwerte.

3.3.10

Hersteller-ID

(en: manufacturer ID)

Zeichenfolge zur Identifizierung des Herstellers, der ein Gerät produziert hat

ANMERKUNG Ein Hersteller muss den ihm zugewiesenen Wert verwenden und darf keinen einem anderen Hersteller zugewiesenen Wert verwenden.

3.3.11

Master

(en: master)

Gerät, das eine Kommunikationsaktivität durch Senden von Anforderungs-PDUs an andere Geräte initiiert

3.3.12

Polling-Adresse

(en: polling address)

ganze Zahl zur Kennzeichnung eines Gerätes

ANMERKUNG Die Polling-Adresse wird verwendet, um eine ein Oktett lange Adresse zu erstellen.

3.3.13

Slave

(en: slave)

Gerät, das eine Kommunikationsaktivität nur initiiert, nachdem es eine Anforderungs-PDU von einem Master-Gerät empfangen und eine Antwort auf diese Anforderung gesendet hat

3.3.14

Tag

(en: tag)

Zeichenfolge mit 8 ASCII-Zeichen zur Kennzeichnung eines Feldgerätes

3.3.15

eindeutige Adresse

(en: unique address)

fünf Oktette lange Adresse eines Gerätes, die das Gerät unter allen anderen Geräten, die diese Norm unterstützen, eindeutig identifiziert, und durch Verknüpfung einer Hersteller-ID, eines Gerätetyps und einer Geräte-ID gebildet wird

English version

**Industrial communication networks -
Fieldbus specifications -
Part 6-20: Application layer protocol specification -
Type 20 elements
(IEC 61158-6-20:2010)**

Réseaux de communication industriels -
Spécifications des bus de terrain -
Partie 6-20: Spécification des protocoles
des couches d'application -
Éléments de type 20
(CEI 61158-6-20:2010)

Industrielle Kommunikationsnetze -
Feldbusse -
Teil 6-20: Protokollspezifikation des
Application Layer (Anwendungsschicht) -
Typ 20-Elemente
(IEC 61158-6-20:2010)

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 61158-6-20, 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 61158-6-20:2012.

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 61158-6-20:2008.

EN 61158-6-20:2012 includes the following significant technical changes with respect to EN 61158-6-20:2008:

- revised Identify FAL PDU, see 5.3.2;
- revised Read device variables with status FAL PDU, see 5.3.9.

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 61158-6-20:2010 was approved by CENELEC as a European Standard without any modification.

CONTENTS

INTRODUCTION	6
1 Scope	7
1.1 General	7
1.2 Specifications	7
1.3 Conformance	7
2 Normative references	8
3 Terms, definitions, symbols, abbreviations and conventions	8
3.1 Terms and definitions from other ISO/IEC standards	8
3.2 IEC/TR 61158-1 terms	9
3.3 Type 20 fieldbus application-layer specific definitions	12
3.4 Abbreviations and symbols	13
3.5 Conventions	14
3.6 Conventions used in state machines	15
4 Abstract syntax	16
5 Transfer syntax	16
5.1 General	16
5.2 Common APDU structure	16
5.3 Service-specific APDU structures	19
5.4 Data coding rules	32
6 Structure of FAL protocol state machines	37
7 AP-context state machines	38
8 FAL service protocol machine (FSPM)	38
8.1 General	38
8.2 FSPM state tables	39
8.3 Functions used by FSPM	44
8.4 Parameters of FSPM/ARPM primitives	44
9 Application relationship protocol machines (ARPMs)	45
9.1 AREP mapping to data link layer	45
9.2 Application relationship protocol machines (ARPMs)	46
9.3 AREP state machine primitive definitions	48
9.4 AREP state machine functions	48
10 DLL mapping protocol machine (DMPM)	48
10.1 DMPM states	48
10.2 DMPM state machines	49
10.3 Primitives exchanged between data link layer and DMPM	49
10.4 Functions used by DMPM	50
Bibliography	51
Annex ZA (normative) Normative references to international publications with their corresponding European publications	52
Figure 1 – APDU format	16
Figure 2 – Normal response from slave to master	17
Figure 3 – Command error response from slave to master	18
Figure 4 – Communication error response from slave to master	18

Figure 5 – Coding without identification.....	32
Figure 6 – Coding of Integer type data	33
Figure 7 – Coding of Integer16 type data	33
Figure 8 – Coding of Unsigned type data	33
Figure 9 – Coding of Unsigned16 type data	33
Figure 10 – Coding of single precision Floating Point type data	34
Figure 11 – Coding of double precision Floating Point type data	34
Figure 12 – Coding of Date type data	35
Figure 13 – Relationships among protocol machines and adjacent layers	38
Figure 14 – State transition diagram of FSPM	39
Figure 15 – State transition diagram of the client ARPM	46
Figure 16 – State transition diagram of the server ARPM	47
Figure 17 – State transition diagram of DMPM	49
Table 1 – Conventions used for state machines	15
Table 2 – Response code values	17
Table 3 – Device status values.....	17
Table 4 – Response code values	18
Table 5 – Communication error codes.....	18
Table 6 – Identify request APDU	19
Table 7 – Identify response value field	19
Table 8 – Identify command specific response codes.....	20
Table 9 – Read primary variable response value field	20
Table 10 – Read primary variable command specific response codes	20
Table 11 – Read loop current and percent of range value field	21
Table 12 – Read loop current and percent of range command specific response codes	21
Table 13 – Read dynamic variables and loop current value field	21
Table 14 – Read dynamic variables and loop current command specific response codes.....	22
Table 15 – Write polling address value field	22
Table 16 – Loop current mode codes	22
Table 17 – Write polling address command specific response codes.....	23
Table 18 – Read loop configuration value field	23
Table 19 – Read loop configuration command specific response codes	23
Table 20 – Read dynamic variable families classifications value field	24
Table 21 – Read dynamic variable families classifications command specific response codes.....	24
Table 22 – Read device variables with status request value field	24
Table 23 – Read device variables with status value field	24
Table 24 – Variable status values.....	26
Table 25 – Read device variables with status command specific response codes.....	26
Table 26 – Read message response value field	27
Table 27 – Read message command specific response codes	27
Table 28 – Read tag, descriptor, date response value field	27
Table 29 – Read tag, descriptor, date command specific response codes.....	28

Table 30 – Read primary variable transducer information response value field	28
Table 31 – Read primary variable transducer information command specific response codes.....	28
Table 32 – Read device information response value field	29
Table 33 – Read device information command specific response codes	29
Table 34 – Read final assembly number response value field.....	29
Table 35 – Read final assembly number command specific response codes	30
Table 36 – Write message value field	30
Table 37 – Write message command specific response codes.....	30
Table 38 – Write tag, descriptor, date value field.....	30
Table 39 – Write tag, descriptor, date command specific response codes.....	31
Table 40 – Write final assembly number value field.....	31
Table 41 – Write final assembly number command specific response codes	31
Table 42 – Read long tag response value field	32
Table 43 – Read long tag command-specific response codes	32
Table 44 – Write long tag value field	32
Table 45 – Write long tag command specific Response codes	32
Table 46 – Coding for Date type	35
Table 47 – Coding for one octet Enumerated Type	35
Table 48 – One octet bit field	36
Table 49 – Packed ASCII character set	36
Table 50 – Acceptable subset of ISO Latin-1 characters.....	37
Table 51 – FSPM state Table – client transactions.....	39
Table 52 – FSPM state Table – server transactions	43
Table 53 – Function Command ().....	44
Table 54 – Function CommErr ().....	44
Table 55 – Function CommandErr ().....	44
Table 56 – Function Resp ().....	44
Table 57 – Function Device ()	44
Table 58 – Parameters used with primitives exchanged between FSPM and ARPM	44
Table 59 – Client ARPM states	46
Table 60 – Client ARPM state table	47
Table 61 – Server ARPM states.....	47
Table 62 – Server ARPM state table	47
Table 63 – Primitives issued from ARPM to DMPM.....	48
Table 64 – Primitives issued by DMPM to ARPM	48
Table 65 – Parameters used with primitives exchanged between ARPM and DMPM	48
Table 66 – DMPM state descriptions	48
Table 67 – DMPM state Table – Client transactions.....	49
Table 68 – DMPM state Table – Server transactions	49
Table 69 – Primitives exchanged between data-link layer and DMPM.....	49

INTRODUCTION

This part of IEC 61158 is one of a series produced to facilitate the interconnection of automation system components. It is related to other standards in the set as defined by the “three-layer” fieldbus reference model described in IEC/TR 61158-1.

The application protocol provides the application service by making use of the services available from the data-link or other immediately lower layer. The primary aim of this standard is to provide a set of rules for communication expressed in terms of the procedures to be carried out by peer application entities (AEs) at the time of communication. These rules for communication are intended to provide a sound basis for development in order to serve a variety of purposes:

- as a guide for implementors and designers;
- for use in the testing and procurement of equipment;
- as part of an agreement for the admittance of systems into the open systems environment;
- as a refinement to the understanding of time-critical communications within OSI.

This standard is concerned, in particular, with the communication and interworking of sensors, effectors and other automation devices. By using this standard together with other standards positioned within the OSI or fieldbus reference models, otherwise incompatible systems may work together in any combination.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 6-20: Application layer protocol specification – Type 20 elements

1 Scope

1.1 General

The Fieldbus Application Layer (FAL) provides user programs with a means to access the fieldbus communication environment. In this respect, the FAL can be viewed as a “window between corresponding application programs.”

This standard provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 20 fieldbus. The term “time-critical” is used to represent the presence of a time-window, within which one or more specified actions are required to be completed with some defined level of certainty. Failure to complete specified actions within the time window risks failure of the applications requesting the actions, with attendant risk to equipment, plant and possibly human life.

This standard defines in an abstract way the externally visible behavior provided by the Type 20 of the fieldbus Application Layer in terms of

- a) the abstract syntax defining the application layer protocol data units conveyed between communicating application entities,
- b) the transfer syntax defining the application layer protocol data units conveyed between communicating application entities,
- c) the application context state machine defining the application service behavior visible between communicating application entities; and
- d) the application relationship state machines defining the communication behavior visible between communicating application entities; and.

The purpose of this standard is to define the protocol provided to define

- a) the wire-representation of the service primitives defined in IEC 61158-5-20, and
- b) the externally visible behavior associated with their transfer.

This standard specifies the protocol of the Type 20 IEC fieldbus application layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498) and the OSI Application Layer Structure (ISO/IEC 9545).

1.2 Specifications

The principal objective of this standard is to specify the syntax and behavior of the application layer protocol that conveys the application layer services defined in IEC 61158-5-20.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of protocols standardized in IEC 61158-6.

1.3 Conformance

This standard does not specify individual implementations or products, nor does it constrain the implementations of application layer entities within industrial automation systems. Conformance is achieved through implementation of this application layer protocol specification.

2 Normative references

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

IEC 60559, *Binary floating-point arithmetic for microprocessor systems*

IEC/TR 61158-1:2010¹, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-5-20, *Industrial communication networks – Fieldbus specifications – Part 5-20: Application layer service definition – Type 20 elements*

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

ISO/IEC 8824-1, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*

ISO/IEC 8859-1, *Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*

ISO/IEC 9545, *Information technology – Open Systems Interconnection – Application Layer structure*

3 Terms, definitions, symbols, abbreviations and conventions

For the purposes of this document, the following definitions apply.

3.1 Terms and definitions from other ISO/IEC standards

3.1.1 Terms and definitions from ISO/IEC 7498-1

- a) abstract syntax
- b) application entity
- c) application process
- d) application protocol data unit
- e) application service element
- f) application entity invocation
- g) application process invocation
- h) application transaction
- i) presentation context
- j) real open system
- k) transfer syntax

3.1.2 Terms and definitions from ISO/IEC 9545

- a) application-association
- b) application-context
- c) application context name
- d) application-entity-invocation
- e) application-entity-type
- f) application-process-invocation
- g) application-process-type
- h) application-service-element
- i) application control service element

¹ To be published.

3.1.3 Terms and definitions from ISO/IEC 8824-1

- a) object identifier
- b) type
- c) value
- d) simple type
- e) structured type
- f) component type
- g) tag
- i) true
- j) false
- k) integer type
- m) octet string type
- n) null type
- o) sequence type
- p) sequence of type
- q) choice type
- r) tagged type
- s) any type
- t) module
- u) production

3.1.4 Terms and definitions from ISO/IEC 8825

- a) encoding (of a data value)
- b) data value
- c) identifier octets (the singular form is used in this standard)
- d) length octet(s) (both singular and plural forms are used in this standard)
- e) contents octets

3.2 IEC/TR 61158-1 terms

The following IEC/TR 61158-1 terms apply.

3.2.1

application

function or data structure for which data is consumed or produced

3.2.2

application layer interoperability

capability of application entities to perform coordinated and cooperative operations using the services of the FAL

3.2.3

application object

object class that manages and provides the run time exchange of messages across the network and within the network device

NOTE Multiple types of application object classes may be defined.

3.2.4

application process

part of a distributed application on a network, which is located on one device and unambiguously addressed

3.2.5

application process identifier

identifier that distinguishes among multiple application processes used in a device

3.2.6

application process object

component of an application process that is identifiable and accessible through an FAL application relationship

NOTE Application process object definitions are composed of a set of values for the attributes of their class (see the definition for Application Process Object Class Definition). Application process object definitions may be accessed remotely using the services of the FAL Object Management ASE. FAL Object Management services can be used to load or update object definitions, to read object definitions, and to dynamically create and delete application objects and their corresponding definitions

3.2.7

application process object class

class of application process objects defined in terms of the set of their network-accessible attributes and services

3.2.8

application relationship

cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation

NOTE This relationship is activated either by the exchange of application-protocol-data-units or as a result of pre-configuration activities

3.2.9

application relationship application service element

application-service-element that provides the exclusive means for establishing and terminating all application relationships

3.2.10

application relationship endpoint

context and behavior of an application relationship as seen and maintained by one of the application processes involved in the application relationship

NOTE Each application process involved in the application relationship maintains its own application relationship endpoint.

3.2.11

attribute

description of an externally visible characteristic or feature of an object

NOTE The attributes of an object contain information about variable portions of an object. Typically, they provide status information or govern the operation of an object. Attributes may also affect the behaviour of an object. Attributes are divided into class attributes and instance attributes.

3.2.12

behaviour

indication of how the object responds to particular events. Its description includes the relationship between attribute values and services

3.2.13

class

set of objects, all of which represent the same kind of system component

NOTE A class is a generalisation of the object; a template for defining variables and methods. All objects in a class are identical in form and behaviour, but usually contain different data in their attributes.

3.2.14

class attributes

attribute that is shared by all objects within the same class

3.2.15

class code

unique identifier assigned to each object class

3.2.16

class specific service

service defined by a particular object class to perform a required function which is not performed by a common service

NOTE A class specific object is unique to the object class which defines it.