IEC 61162-450

Edition 2.0 2018-05

INTERNATIONAL STANDARD

Maritime navigation and radiocommunication equipment and systems – Digital interfaces –
Part 450: Multiple talkers and multiple listeners – Ethernet interconnection

IEC 61162-450:2018-05(en)

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

About the IEC
The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications
The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue
The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - webstore.iec.ch/advsearchform
The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing 21 000 terms and definitions in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary
67 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc
If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: sales@iec.ch.

IEC 61162-450

Edition 2.0 2018-05

INTERNATIONAL STANDARD

Maritime navigation and radiocommunication equipment and systems – Digital interfaces –
Part 450: Multiple talkers and multiple listeners – Ethernet interconnection

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 47.020.70

ISBN 978-2-8322-5636-7

Warning! Make sure that you obtained this publication from an authorized distributor.

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

MARITIME NAVIGATION AND RADIOCOMMUNICATION EQUIPMENT AND SYSTEMS – DIGITAL INTERFACES –

Part 450: Multiple talkers and multiple listeners – Ethernet interconnection

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 61162-450 has been prepared by IEC technical committee 80: Maritime navigation and radiocommunication equipment and systems.

This second edition of IEC 61162-450 cancels and replaces the first edition published in 2011 and Amendment 1:2016. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

a) network traffic filtering based on IGMP snooping added;
b) network traffic balancing added;
c) new encapsulation of IEC 61162-3 PGNs added;

d) new alternative for binary file transfer added: TCP/IP based on Annex H of IEC 62388:2007 on radars;

e) general authentication tag "a:" added to support managing of cyber security risk.

The text of this International Standard is based on the following documents:

FDIS	Report on voting
80/880/FDIS	80/885/RVD

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

This document has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in the IEC 61162 series, published under the general title *Maritime navigation and radiocommunication equipment and systems -Digital interfaces*, can be found on the IEC website.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under "http://webstore.iec.ch" in the data related to the specific document. At this date, the document will be

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

A bilingual version of this publication may be issued at a later date.

MARITIME NAVIGATION AND RADIOCOMMUNICATION EQUIPMENT AND SYSTEMS – DIGITAL INTERFACES –

Part 450: Multiple talkers and multiple listeners – Ethernet interconnection

1 Scope

This part of IEC 61162 specifies interface requirements and methods of test for high speed communication between shipboard navigation and radiocommunication equipment as well as between such systems and other ship systems that need to communicate with navigation and radio-communication equipment. This document is based on the application of an appropriate suite of existing international standards to provide a framework for implementing data transfer between devices on a shipboard Ethernet network.

This document specifies an Ethernet based bus type network where any listener can receive messages from any sender with the following properties.

- This document includes provisions for multicast distribution of information formatted according to IEC 61162-1, for example position fixes and other measurements, as well as provisions for transmission of general data blocks (binary file), for example between radar and VDR, and also includes provisions for multicast distribution of information formatted according to IEC 61162-3, for example position fixes and other measurements.
- This document is limited to protocols for equipment (network nodes) connected to a single Ethernet network consisting only of OSI level one or two devices and cables (Network infrastructure).
- This document provides requirements only for equipment interfaces. By specifying protocols for transmission of IEC 61162-1 sentences, IEC 61162-3 PGN messages and general binary file data, these requirements will guarantee interoperability between equipment implementing this document as well as a certain level of safe behaviour of the equipment itself.
- This document permits equipment using other protocols than those specified in this document to share a network infrastructure, provided that it is supplied with interfaces which satisfy the requirements described for ONF.
- This document includes provisions for filtering of the network traffic in order to limit the amount of traffic to manageable level for each individual equipment.

This document does not contain any system requirements other than the ones that can be inferred from the sum of individual equipment requirements. An associated standard, IEC 61162-460, further addresses system requirements.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 60825-2, *Safety of laser products – Part 2: Safety of optical fibre communication systems (OFCS)*

IEC 60945, *Maritime navigation and radiocommunication equipment and systems – General Requirements – Methods of testing and required test results*

IEC 61162-1:2016, *Maritime navigation and radiocommunication equipment and systems – Digital interfaces – Part 1: Single talker and multiple listeners*

IEC 61162-3:2008, *Maritime navigation and radiocommunication equipment and systems – Digital interfaces – Part 3: Serial data instrument network*

IEEE Std 802.3-2015, *IEEE Standard for Ethernet*

ISOC RFC 768, *User Datagram Protocol, Standard STD0006*

ISOC RFC 791, *Internet Protocol (IP), Standard STD0005 (and updates)*

ISOC RFC 792, *Internet Control Message Protocol (ICMP), Standard STD0005 (and updates)*

RFC 793:1981, *Transmission Control Protocol (TCP)*

ISOC RFC 826, *An ethernet Address Resolution Protocol*

ISOC RFC 1112, *Host Extensions for IP Multicasting, Standard STD0005 (and updates),* (include IGMP version 1)

ISOC RFC 1918, *Address Allocation for Private Internets, Best Current Practice BCP0005*

ISOC RFC 2236, *Internet Group Management Protocol, Version 2*

ISOC RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*

ISOC RFC 3376, *Internet Group Management Protocol, Version 3*

ISOC RFC 5000, *Internet Official Protocol Standards, Standard 0001*

ISOC RFC 5227, *IPv4 Address Conflict Detection*

ISOC RFC 5424, *The Syslog Protocol*

NMEA 0183:2008, *Standard for interfacing marine electronic devices, Version 4.00*

NOTE The standards of the Internet Society (ISOC) are available on the IETF websites http://www.ietf.org. Later updates can be tracked at http://www.rfc-editor.org/rfcsearch.html.

3 Terms and definitions

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

3.1
ASCII
printable 7 bit character encoded in one byte

3.2
binary file
data block without formatting known to this protocol, i.e., non IEC 61162-1 formatted data, that can be transmitted with the protocol defined in 7.3 or in 7.5

Note 1 to entry: The term "binary file" is used to differentiate the general data transfer protocol (which may or may not be in ordinary text format) from the transmission of sentences that is always in 7 bit ASCII format.

3.3
byte
group of 8 bits treated as one unit

Note 1 to entry: This corresponds to what is also sometimes called an octet.

3.4
command-response pair
CRP
messages exchanged between parties that synchronize state changes on both sides through the exchange

Note 1 to entry: CRP are defined in Annex A.

Note 2 to entry: Both the command and the reply message may also be used as a sensor broadcast message in some cases. Thus, the implementation of the semantics of the message exchange is somewhat different between different users of the exchange.

3.5
datagram
atomic UDP transmission unit on the Ethernet as defined in ISOC RFC 768 and as constrained elsewhere in this document

3.6
Ethernet
carrier sense, multiple access collision detect (CSMA/CD) local area network protocol standard as defined in IEEE Std 802.3 and later revisions and additions to IEEE 802

Note 1 to entry: The types of Ethernet media that can be used for implementation of this document are defined in Clause 5.

3.7
function block
specified functionality implemented by equipment

Note 1 to entry: Equipment normally implements multiple function blocks. Requirements to equipment are the sum of requirements to the function blocks it implements. Function blocks are defined in Clause 4.

3.8
Internet Group Management Protocol
IGMP
communications protocol used by hosts and adjacent routers on IPv4 networks to establish multicast group memberships

Note 1 to entry: The IGMP is an integral part of IP multicast.

3.9
IGMP snooping
process of listening to Internet Group Management Protocol (IGMP) network traffic

3.10
Internet assigned number authority
IANA
global coordination of the Domain Name Server (DNS) Root, IP addressing, and other Internet protocol resources, including UDP and TCP port numbers

Note 1 to entry: The currently assigned numbers are listed in http://www.iana.org/assignments/port-numbers.

3.11
Internet protocol
IP
signalling protocol used and defined in ISOC RFC 791 (and updates)

3.12
message
collection of one or more sentences that are grouped by mechanisms internal to the sentence, for instance by sequence numbers as in the TXT sentence

Note 1 to entry: A stand-alone sentence is a message.

3.13
message type
classification of IEC 61162-1 sentence formatters into SBM, MSM and CRP types

Note 1 to entry: SBM, MSM and CRP types are defined in Annex A.

Note 2 to entry: This document defines different requirements to the transmission of different message types.

3.14
multi-sentence message
MSM
logical group of messages and/or sentences where the full meaning of the group is dependent on the receiver reading the full group

Note 1 to entry: Multi-sentence messages that are grouped together with a TAG construct are also a sentence group.

Note 2 to entry: MSM are defined in Annex A.

3.15
network
physical Ethernet network with one Internet address space, consisting only of the network nodes, switches, cables and supporting equipment such as power supply units

3.16
network function block
NF
function block responsible for physical connectivity to the network and connectivity to the transport layer as described in 4.3

3.17
network infrastructure
part of the network that provides a transmission path between network nodes

Note 1 to entry: The network nodes are not part of the network infrastructure.

3.18
network node
physical device connected to the network and which have an Internet address

Note 1 to entry: It is also called an Internet host.

Note 2 to entry: A network node will normally correspond to equipment. “Equipment” is used in this document.

3.19
other network function block
ONF
function block that interfaces to the network, but which is not using the protocol definition in Clauses 5, 6 and 7

Note 1 to entry: For example, real time streaming of radar and CCTV image transfer, or VDR sound transfer.

Note 2 to entry: Requirements as defined in 4.7 ensure that an ONF can co-reside with SF network nodes and function blocks that make use of this document's protocol.

3.20
PGN to network gateway function block
PNGF
function block that enables transfer of sentences between the network and devices that are compliant with the IEC 61162-3 serial data instrument network interface

3.21
PGN message
parameter group number message
message consisting of an 8-bit or 16-bit number that identifies each parameter group

Note 1 to entry: The parameter group number (PGN) is analogous to the three-character sentence formatter in IEC 61162-1. By definition, parameter groups identified by 16-bit parameter group numbers are broadcast to all addresses on the network. Parameter groups identified by 8-bit parameter group numbers may be used to direct data for use by a specific address.

[SOURCE: IEC 61162-3:2008, 3.1.21, modified – The word "message" has been added to the term, and the definition has been rephrased.]

3.22
sensor broadcast message
SBM
message consisting of only one sentence

Note 1 to entry: SBMs are sent with a sufficiently high update rate to ensure that the receiver can maintain the correct status even in environments where some messages may be lost.

Note 2 to entry: SBMs are defined in Annex A.

3.23
sentence
standard information carrying unit as described in IEC 61162-1

3.24
sentence group
logical group of sentences (which may consist of only one) that need to be processed together to give full meaning to the information contained in the sentence(s)

Note 1 to entry: The grouping of sentences into sentence group is done by TAG block mechanisms.

Note 2 to entry: This document allows the explicit grouping of sentences by using coding in a datagram. This document does not enforce any relationship between datagram and sentence group. Thus a datagram may contain more than one sentence group, or a sentence group may be split over two or more datagrams.

3.25
serial to network gateway function block
SNGF
function block that enables transfer of sentences between the network and devices that are compliant with the IEC 61162-1 and IEC 61162-2 serial line interface

3.26
system function block
SF
function block, identified by a unique system function ID (SFI), that is the only function block that can send information in a datagram format as defined in Clause 7

3.27
system function ID
SFI
parameter string as defined in 4.4.2

3.28
transmission group
pair of a multicast address and a port number that are used by an SF to transmit sentences

Note 1 to entry: The transmission groups are defined in Table 4, and Annex A defines default transmission groups for the SF.

3.29
transport annotate and group
TAG
formatted block of data, defined in NMEA 0183, that adds parameters to IEC 61162-1 sentences

Note 1 to entry: Annex B gives an overview of the TAG blocks used in this document.

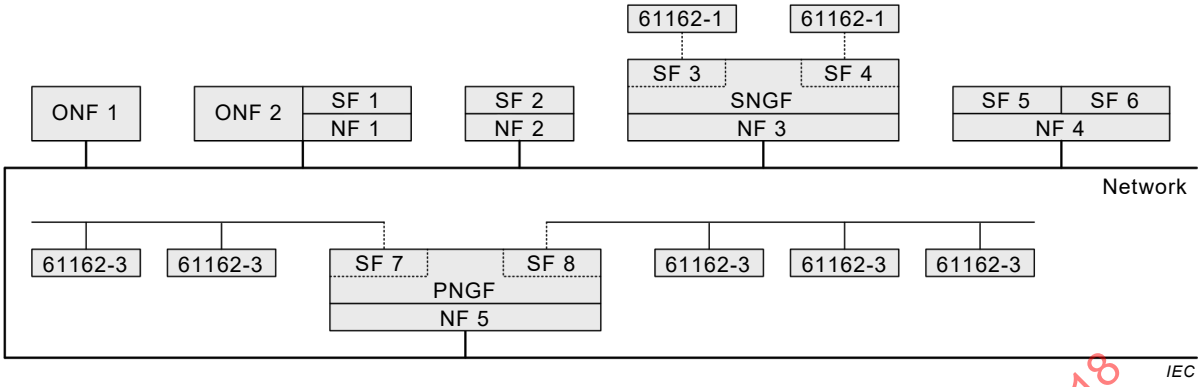
3.30
user datagram protocol
UDP
connection-less datagram protocol defined by ISOC RFC 768

Note 1 to entry: ISOC RFC 768 makes no provision for transport-layer acknowledgement of packets received.

4 General network and equipment requirements

4.1 Network topology example

Figure 1 shows a possible IEC 61162-450 network topology consisting of one IP local area network (LAN) and a number of different network nodes, each containing different function blocks. This diagram is informal and does not imply any requirements other than the ones defined in Clause 4.



Key

SF system function block

NF network function block

SNGF serial to network gateway function block

ONF other network function block

PNGF PGN to network gateway function block

Figure 1 – Network topology example

Some examples of network nodes are (see Figure 1):

- a sensor, for example a GNSS receiver that is also a network node (SF2 and NF2);
- a device that sends or receives IEC 61162-450 compliant data (sentences and/or binary file) as well as other types of information onto the network, for example an ECDIS that can also load chart data from another device (SF1, ONF2 and NF1);
- two independent functions, such as a gyrocompass also approved as a rate of turn sensor that are implemented in one network node (SF5, SF6 and NF4);
- a system device function block represented by an IEC 61162-1 compliant equipment connected to a serial to network gateway function (SNGF); in this case, the SNGF will format outgoing sentences according to requirements in this document (SF3, SF4, SNGF and NF3);
- a system device function block presented by an IEC 61162-3 compliant equipment connected to network gateway function (PNGF); in this case, the PNGF will format outgoing sentences according to requirements of this document (SF7, SF8, PNGF and NF5);
- a device that does not send or receive IEC 61162-450 compliant data (sentences and/or binary file), but which satisfies minimum requirements for compatible use of the same network (ONF1).

4.2 Basic requirements

4.2.1 Requirements for equipment to be connected to the network

(see 8.2.1)

The requirements for equipment connected to the network are as follows.

- All equipment connected to the network, including network infrastructure equipment, shall satisfy the relevant physical and electrical requirements defined in 5.1.
- All equipment that implements one or more of SF and/or SNGF shall implement the NF. This equipment shall satisfy the requirements to the function blocks they implement as defined in 4.3 (NF), 4.4 (SF), 4.5 (SNGF) and 4.6 (PNGF).

- All other equipment that is not network infrastructure equipment and that shares the network infrastructure shall comply with requirements to an ONF as defined in 4.7.
- Network infrastructure equipment, i.e., switches, shall satisfy requirements in 4.2.2.
- All equipment connected to a network shall satisfy the requirements of IEC 60945.

  NOTE This requirement only applies to devices on the network when the network is in normal operation. During commissioning or maintenance, when the system is not being used for safety-related navigation, other equipment can be temporarily connected to the network that does not comply with IEC 60945.

Any other equipment is not allowed to be connected to the network.

4.2.2 Additional requirements for network infrastructure equipment

(see 8.2.2)

To avoid potential problems with certain network infrastructure equipment, repeater hubs shall not be used to interconnect components of an IEC 61162-450 network.

NOTE 1 Repeater hubs are network infrastructure devices without internal storage that repeat incoming datagrams onto all outgoing connections.

NOTE 2 Switches are network infrastructure devices that based on forwarding tables can process, and forward datagrams between nodes on the same network, using intermediate storage in the switch before retransmission.

Switches used in an IEC 61162-450 network shall have means to filter network traffic using IGMP snooping. When the IGMP snooping is enabled and when a multicast datagram is received, the switch shall forward it only to the ports which have joined the same multicast group. The means which shall be provided to support multicast data filtering using IGMP snooping are the following:

- IGMP snooping shall be provided based on IGMPv1, IGMPv2 or IGMPv3; the selection of the IGMP version shall be based on highest version supported by all the connected nodes;
- multicast traffic filtering shall be provided based on IP multicast address;
- multicast data filtering shall not be enabled for the address range of 224.0.0.1 to 224.0.0.255 as recommended in RFC 4541.

In addition to or instead of multicast filtering techniques, such as IGMP snooping, it is also permitted to configure manually individual ports of the switches to block unnecessary traffic flow (for example to isolate simple sensors from ECDIS and radar).

See Annex D for IGMP snooping compatibility issues of nodes based on IEC 61162-450:2011.

Another possible method to filter and control network traffic is described in Annex E.

4.3 Network function (NF) requirements

4.3.1 General requirements

All equipment that implements a NF shall satisfy the requirements in Clauses 5 and 6.

4.3.2 Maximum data rate requirements

(see 8.3.1)

The manufacturer shall specify the maximum input rate under which the equipment can still perform all functions required by its performance standards.

Maximum input rate shall be specified as

a) the maximum number of datagrams per second received, intended for and processed by the equipment,

b) the maximum number of datagrams per second received by, but not intended for, the equipment, and

c) the maximum number of datagrams per second received by, but not intended for, the equipment at 50 % of the maximum load for item a).

NOTE 1 "Received by" means datagrams that are received on all transmission groups that the equipment listens to.

NOTE 2 "Intended for" are datagrams that are processed by the equipment as part of its specified function.

The maximum data rates shall be the mean rate over a 10 s measurement period.

4.3.3 Error logging function

(see 8.3.2)

4.3.3.1 Internal logging

Means shall be provided in each NF to record errors that occur in the NF itself as well as SF and SNGF using it. Subclauses 4.5.2, 7.1.2, 7.2.5 and 7.3.9 give minimum requirements as to what shall be logged.

As a minimum, the manufacturer shall provide mechanisms by which error logs can be inspected by a human operator, for example by trained service engineer. It is allowed that the inspection is done through a simple network mechanism, such as a terminal emulator, as defined in this document or any other reasonable method.

The minimum requirements for the log are to count the number of each occurrence. The counter may reset itself by a manufacturer specified method.

4.3.3.2 External logging

A NF may be configured to support external logging, where non-trivial information is sent to a logging server. In this case, a "syslog" message as defined in ISOC RFC 5424 shall be used.

Syslog messages shall be formatted as ASCII text messages and sent as UDP packets on port 514 and the multicast address defined in Table 6. Error messages defined in this document shall be reported through a simplified message as described in Table 1, where italicised words are place-holders for data explained in the right hand column. Other characters shall be transmitted as shown, including spaces.

Table 1 – Syslog message format

Element	Description
<pri>	The combined priority and facility code (number from 0 to 199 inclusive) enclosed in pointed brackets. For the errors defined in this document, the value 131 shall be used (facility "local use 0" and priority "error condition").
Version	The version code. The code 1 (one) shall be used for messages from this document.
Space	One space character.
Timestamp	Timestamp, containing date and time and optional UTC offset, in a valid format, for example 1985-04-12T23:20:50-03:00. The example shows date, followed by upper case "T", then local time and finally offset from UTC (3 hours west – negative, east offsets shall be prefixed by a "+". UTC offset can be abbreviated to a single upper case "Z", without leading "-" or "+"). Alternatively, the timestamp field may be nil ("-", a single dash character).
Space	One space character.
Hostname	The host name of the network node, represented as the IP address in dotted decimal notation. Alternatively, this field may be nil ("-", a single dash character).
Space	A space character.
Appname	The application name. This shall be the string "450-" followed by the configured SFI code if the error originates in the SF or SNGF, "NF" if the error originates from the network function block or "ONF" if it originates in the ONF function block.
Space	A space character.
Procid	Normally, this field should be nil ("-" a dash character). Other values as defined in the syslog standard may be used.
Space	A space character.
Msgid	For errors defined in this document, this field shall be the error code as defined in Table 2.
Space	A space character.
Structured	This field can be nil ('-', a single dash character) or contain information as defined in ISOC RFC 5424.
Space	A space character.
Msg	A free format message in ASCII format.

A "syslog" packet shall not exceed 480 bytes and shall be sent as a single UDP datagram. The "syslog" packet for multiple occurrence of same message identity shall not be reported more often than once per minute. The "syslog" packet for any occurrence of message identity shall not be delayed more than 10 min.

This document does not specify requirements for equipment receiving syslog messages. This type of equipment would fall into the category of ONF. As Table 1 is a subset of the full ISOC RFC 5424 specification, implementers of such equipment shall refer to ISOC RFC 5424 and make sure that syslog messages from other ONF can be received and processed without problems.

To facilitate the use of the syslog protocol, the errors defined in this document have been assigned a message identity as defined in Table 2.

Table 2 – Syslog error message codes

Message identity	Description	Subclause
101	SNGF buffer overflow	4.5.2
102	Datagram header error	7.1.2
103	TAG or sentence format error	7.2.5
104	Binary file error	7.3.9
201	PNGF buffer overflow	4.6.2
202	PGN message errors	7.4.2 and 7.4.4
203	No available address for devices	7.4.3.2

Additional information can be given in the "Msg" field, if available.

4.3.4 Provisions for network traffic filtering – IGMP

NOTE The purpose of the IGMP for this document is to provide the possibility to perform network traffic filtering based on IGMP snooping.

The manufacturer shall specify the version of IGMP as defined in ISOC RFC 1112, RFC 2236 and RFC 3376 that the NF supports. At least version 1 as defined in ISOC RFC 1112 shall be implemented.

See Annex D for compatibility issues of nodes based on IEC 61162-450:2011.

4.4 System function block (SF) requirements

4.4.1 General requirements

(see 8.4.1 and 8.2.3)

Equipment that implements an SF shall satisfy the following requirements:

- requirements in 6.2 shall be satisfied for all equipment implementing SF;
- implements at least one of the datagram types defined in Clause 7, but does not have to implement all of them;
- implemented datagram types shall be specified in the manufacture's documentation (see 7.1.1);
- requirements in 7.2 shall be satisfied for all equipment implementing IEC 61162-1 sentence transmitting or receiving function blocks;
- requirements in 7.3 shall be satisfied for equipment that implements an SF that can transmit or receive binary file data;
- requirements in 7.4 shall be satisfied for all equipment implementing IEC 61162-3 PGN message transmitting or receiving function blocks.

4.4.2 Assignment of unique system function ID (SFI)

(see 8.4.2)

The format of the SFI parameter string shall be "ccxxxx"

where "cc" is two valid characters as defined in IEC 61162-1 and "xxxx" is four numeric characters.

An SF implementing the functionality of an equipment that has been given a talker mnemonic code in IEC 61162-1 shall use this talker mnemonic as the "cc" characters in the SFI. If the talker mnemonic is proprietary (i.e. consists of character "P" followed by a three-character

manufacturer's mnemonic code), then two first characters are used as the "cc" characters in the SFI.

Other SF may have their SFI string format defined in other standards or the manufacturer may have to choose a code. In the latter case, the already defined talker mnemonic codes shall be avoided.

The numeric character string "xxxx" will be an instance number in the range "0001" to "9999" The numeric character string "9999" is reserved for an un-configured SF and shall not be used by any transmitting SF during normal operation. However, all receiving equipment shall accept the "9999" string.

During normal operation, the SFI parameter string shall be unique for all SF in an IEC 61162-450 network.

It is recommended that all SF on a ship, independent on whether they are residing on one common network or not, are given a ship unique SFI.

There may be multiple SF, each communication with their own SFI, assigned to a single IP address or MAC address.

Means shall be provided by the manufacturer to configure the SFI for each SF (see 7.2.3.4).

4.4.3 Implementing configurable transmission groups

(see 8.4.3)

As default, each SF shall be assigned a single transmission group/multicast address for all outgoing messages. The default for this transmission group is determined by the SFI as described in Annex A.

For each SF that the equipment implements, the manufacturer shall document the default transmission groups the SF listens to and what sentences it expects to receive on each group. The default transmission groups can be selected by the manufacturer from the list of groups in 6.2.2.

Means shall be provided to configure all transmission groups and the SFs which are assigned to them within the valid range of multicast addresses defined in 5.4. A system integrator may, for example, split an SF into different transmission groups to support optimal load balancing for a given system. Where non-default configurations of SF and transmission groups are utilised, the details should be documented by the system integrator.

4.5 Serial to network gateway function (SNGF) requirements

4.5.1 General requirements

(see 8.5.1)

The SNGF shall implement all relevant functionality defined in 4.4 for each SF it supports.

The SNGF may support one or more serial ports. Unless the SNGF implements multi-SF serial port, each serial port shall be implemented as a separate SF and assigned a separate SFI. If practical, the "cc" part of the SFI shall be based on the talker identifier in use by the serial port.

All sentences, including those with unidentified or illegal content, as well as proprietary sentences shall be transmitted from the SF associated with the serial port. Sentences with unidentified or illegal content shall be sent with a legal transport annotate and group (TAG) block as defined in 7.2.3, but with the raw received serial data following the TAG block.

As a destination, each serial port shall be associated with the corresponding SFI. Outgoing sentences shall be transmitted exactly as received in the datagram.

The SNGF may support one or more sources distinguished by different talker mnemonics at each serial port. Each source in a shared serial port shall be implemented as a separate SF and assigned a separate SFI. If practical, the "cc" part of the SFI shall be based on the talker mnemonic in use by each source in a shared serial port. As a destination, each source in a shared serial port shall be based on the SFI. Proprietary sentences include no talker identifier and based on setup parameters they shall use the same SFI as standardized sentences from the same source. The STN sentence is an additional qualifier for the following sentence. The STN sentence and the following sentence belong to the same SF and shall use the same SFI.

Proprietary sentences belong to the default SF of the associated serial port or to the SF determined by the preceding STN sentence.

The TAG block for source identification (s:) shall be based on the SFI. If available, routing from a 450-network to serial ports shall be based on the TAG block for destination identification (d:).

The default SFI of an SNGF used for administrative purposes, such as syslog, shall use the talker mnemonic "SI".

The SNGF may implement different types of filtering with regard to what serial line sentences are retransmitted as datagrams and what datagrams will result in a serial line sentence being sent. Any filtering methods shall be described in the manufacturer's documentation.

NOTE A typical filtering method would be to use the destination TAG "d" to determine what sentences in incoming datagrams are to be sent on the serial line.

4.5.2 Serial line output buffer management

(see 8.5.2)

An SNGF function block shall provide an independent buffer for each separate SF implemented for each serial port it can send sentences onto. The manufacturer shall specify the maximum buffer capacity for each port. The maximum capacity may be configurable at installation.

The buffer shall be implemented as a FIFO (first in, first out) buffer. In case of a full buffer, newly arrived sentences shall be discarded, unless these sentences are specified as prioritized (see below). Newly arrived sentences will be inserted into the buffer when buffer space is available. The method of treatment of sentences grouped by the TAG g (see 7.2.3.3) may be configurable or specified in the manufacturer's documentation.

The SNGF may implement a priority-based functionality for some sentences with specified sentence formatters. The prioritised formatters may be configurable or specified in the manufacturer's documentation.

Processing of prioritized sentences shall be as follows.

- Only one sentence with identical talker ID and sentence formatter shall exist in the buffer. Exception is a multi sentence message or a TAG block group of sentences: they shall only be replaced in their entirety.

  NOTE When prioritizing AIS VDM and VDO sentences, the string beginning with the "!" character and ending with the 7th character of the encapsulation field is used for comparison to identify identical sentences. A match of this string from a newly arrived sentence with one in the buffer means the sentence contains the same ITU-R M.1371 message from the same MMSI as the sentence already in the buffer, and can then replace the older sentence at its position in the queue.

- If a single sentence, multi sentence message or a TAG block grouped sentences, with identical talker ID and sentence formatter exists in the buffer, the new sentence or

sentences will replace the existing sentence or sentences at its position in the queue. This replacing shall not cause logging of an error nor sending anything to syslog.

When prioritizing TAG block grouped sentences, several fields within the TAG block need to be compared as well as the sentence comparisons. All of the compared components should match those of the current TAG block group in order to the replace TAG block group in the queue. The components to compare are: the TAG block source parameter code value, the "number of lines" portion of the TAG block group parameter code, and the sentences within the TAG block group.

- Otherwise, the new sentence shall follow the FIFO principle as described above.

If a sentence is discarded from the queue, this event shall be logged as an error internally in the equipment as defined in 4.3.3. The equipment shall have separate error counts for each serial port.

4.5.3 Datagram output requirements

(see 8.5.3)

The SNGF shall format outgoing datagrams as defined in 7.2.

The SNGF shall either transmit one IEC 61162-1 sentence or, if part of a multi sentence sequence, may transmit multiple IEC 61162-1 sentences per outgoing IEC 61162-450 datagram. The multi sentence sequence includes the case described in IEC 61162-1:2016, 7.3.9, and the cases for which IEC 61162-1 requires a sentence sent prior sending another sentence. The datagram shall include the correct SFI, source identification (s:) and, if required, destination identification (d:).

4.5.4 Multi SF serial port

(see 8.5.4)

The SNGF is allowed to implement more than one SFs for any single serial line. Received sentences on this serial line with a valid talker mnemonic will be transmitted from one of the associated SFs dependent on the talker mnemonics. Each SF shall be assigned a separate SFI and, as a destination, transmit outgoing sentences on the serial line according to the rules in 4.5.1.

Proprietary sentences received on the serial line include no talker identifier. It shall be determined by setup parameters from what SF they shall be transmitted.

Unidentified data from the serial line shall be sent from all SFs associated with the serial port. This sending of unidentified data shall not cause logging of an error nor sending anything to syslog.

4.5.5 Handling malformed data received on serial line

(see 8.5.5)

The SNGF is intended as a remote serial data converter with minimum data processing. For each of the cases below, the SNGF shall send a datagram with the malformed data as required by 4.5.1 and 4.5.4. If the formatted message exceeds the maximum datagram length, the data shall be truncated from the end. The following cases shall cause a message containing the malformed data to be sent:

1) if data has been received before a start character;
2) if data has been received after a valid start character and the maximum sentence and TAG block length has been exceeded;
3) if data has been received after a valid start character and end of line (CR,LF) has not been received after 1 s;

4) if a reserved character has been received and not having been appropriately escaped;

5) if random binary data is received on the serial line.

"Start character" is a valid start of sentence ("$", "!") or TAG block start character.

4.6 PGN to network gateway function (PNGF) requirements

(see 8.12)

4.6.1 General requirements

(see 8.12)

The PNGF shall implement all relevant functionality for each SF it supports as defined in 4.4.

The default SFI of a PNGF used for administrative purposes, such as syslog, shall use the talker mnemonic "SI".

The PNGF may implement different types of filtering based on the PGN messages from and to IEC 61162-3 network. Any filtering methods shall be described in the manufacturer's documentation.

NOTE The accurate timing between PGN messages available in the IEC 61162-3 network is not supported when the same is converted into IEC 61162-450 network.

4.6.2 Output buffer management from IEC 61162-450 network to IEC 61162-3 network

(see 8.12)

A PNGF function block shall provide an independent buffer for each IEC 61162-3 network it can send into. The manufacturer shall specify the maximum buffer capacity for each port. The maximum capacity may be configurable at installation.

PNGF buffer management shall be based on the IEC 61162-3 priority included into each message. The manufacturer shall describe the method in documentation.

If the buffer is full and a PGN message is discarded, it shall be recorded as specified in 4.3.3.

4.6.3 Datagram output requirements

(see 8.12)

The PNGF shall format outgoing messages as defined in 7.4.1.

The PNGF shall transmit one IEC 61162-3 PGN message per outgoing IEC 61162-450 datagram to minimise delays.

4.6.4 PGN group number

(see 8.12)

A PGN group is defined as a logical group of devices that can share the information and message. A message from a device is broadcasted to all devices that belong to the same PGN group. A device may belong to more than one PGN groups. The maximum number of PGN groups is no more than four. The PGN group may be used for filtering of messages (see 4.6.1).

4.7 Other network function (ONF) requirements

(see 8.6)

The ONF represents a function that is allowed to share the same network infrastructure as the network function blocks (NF) on an IEC 61162-450 network.

The ONF shall conform to the requirements given in 4.2.1.

The ONF equipment shall not use any IP multicast address reserved by this document as defined in 5.4.

Documentation shall be provided describing the network protocols used by the ONF to send datagrams or byte steams, for instance UDP, TCP/IP or other.

Documentation shall be provided describing the impact of the ONF to the network.

5 Low level network requirements

5.1 Electrical and mechanical requirements

(see 8.7.1)

The cable and connectors used shall at least meet the specifications listed in Table 3 when used in protected environment as defined in IEC 60945.

The safety requirements and installation practices specified in IEEE Std 802.3™-2015, 14.7 and Clause 27, shall be followed. Also refer to IEEE Std 802.3-2015, informative Annex 67.

Fibre optic interfaces shall comply with the laser safety requirements for Class 1 devices specified in IEC 60825-2.

The physical layer requirement for IEC 61162-3 ports of the PNGF shall be compliant with IEC 61162-3:2008, Clause 4.

Table 3 – Interfaces, connectors and cables

IEEE 802.3 Interface	Max network segment link distance	Mechanical device interface connector type (protected environment)	Pin assignment	Cable category, minimum
100BASE-TXS IEEE Std 802.3-2015, 14.7 and Clauses 24 and 25	100 m	IEC 60603-7-3, 8-way shielded modular connector Refer to IEEE Std 802.3-2015, Clause 3, IEC 60603-7, Figures 1 through 5, and IEEE Std 802.3/25	[b]	CAT5 STP Two shielded twisted pairs ANSI/TIA/EIA-568-A, ANSI/TIA/EIA-568-B or ISO/IEC 11801 (Class D).
(not specified)	[a]	Terminal block	[b]	CAT5 STP Two shielded twisted pairs
100BASE-SX IEEE Std 802.3-2015, Clauses 24 and 26	550 m	IEC 61754-20 LC type duplex optical connector[d]		Two multimode optical fibres Short wavelength 850 nm
1000BASE-T IEEE Std 802.3:2015, Clause 40	100 m	IEC 60603-7-7, 8-way shielded modular connector Refer to IEEE Std 802.3-2015, Clause 3, and IEC 60603-7, Figures 1 through 5. See IEEE Std 802.3/25	[c]	CAT5 STP Four shielded twisted pairs ANSI/TIA/EIA-568-A, ANSI/TIA/EIA-568-B or ISO/IEC 11801 (Class D).
1000BASE-SX IEEE Std 802.3-2015, Clause 38	220 m (62/125 µm, low modal bw) 550 m (50/125 µm, high modal bw)	IEC 61754-20 LC type duplex optical connector[d]		Two multimode optical fibres Short wavelength 850 nm
For use in exposed environments, additional provisions are necessary. Consideration should be given to the M12-type specified in IEC 61076-2-101 for copper network cable. And similar rugged connector should be considered for external fibre optic connections.				

[a] In this case, the maximum operating distance should be specified by the manufacturer.

[b] The 8-way modular connector specified in IEC 60603-7 is the "8P8C" type that has commonly been used in desktop computer LAN connections and incorrectly but widely referred to as "RJ45". Wires are in the order 1, 2, 3, 6, 4, 5, 7, 8 on the modular jack; the same at each end of a cable. The color-order from wire 1 to 8 shall be green/white, green, orange/white, blue, blue/white, orange, brown/white, brown; the same at both ends of the cable. Refer to IEEE Std 802.3-2015, 25.4.3, and IEC 60603-7-3.

[c] The 8-way modular connector specified in IEC 60603-7 is the "8P8C" type that has commonly been used in desktop computer LAN connections and incorrectly but widely referred to as "RJ45". Wires are in the order 1, 2, 3, 6, 4, 5, 7, 8 on the modular jack; the same at each end of a cable. The color-order from wire 1 to 8 shall be green/white, green, orange/white, blue, blue/white, orange, brown/white, brown; the same at both ends of the cable. Refer to IEEE Std 802.3-2015, 40.8.1 and IEC 60603-7-7.

[d] See TIA/EIA-604-10-A:2002.

5.2 Network protocol requirements

(see 8.7.2)

Equipment shall implement IPv4 as generally described in ISOC RFC 5000 with a minimum requirement of support for the following specific network protocols:

- ARP – Address Resolution Protocol as described in ISOC RFC 826 and as updated in ISOC RFC 5227;

- IP – Internet Protocol as described in ISOC RFC 791 and as updated in ISOC RFC 2474;
- UDP – User datagram Protocol as described in ISOC RFC 768;
- UDP Multicast – Host groups as described in ISOC RFC 966 and Host extensions as described in ISOC RFC 1112;
- TCP – Transmission Control Protocol as described in ISOC RFC 793;
- ICMP – Internet Control Message Protocol as described in ISOC RFC 792;
- IGMP – Internet Group Management Protocol as described in ISOC RFC 1112, RFC 2236 or RFC 3376;

5.3 IP address assignment for equipment

(see 8.7.3)

Means shall be provided to configure the equipment to any of the addresses reserved for use in private networks as described in ISOC RFC 1918 with any valid network address mask. The default sub-net mask shall be set appropriately for 192.168.0.0/24 (legacy Class C). The assigned IP address shall remain fixed during normal operation of the equipment, including powering the equipment down and up.

Specific 450-Nodes may choose to exclude a few sub-nets to facilitate internal sub-nets (internal to the equipment) which shall be documented. The following sub-nets shall always be available to the IEC 61162-450 network: 192.168.0.0/24 – 192.168.10.0/24 and 172.16.0.0/16 (Class B).

5.4 Multicast address range

(see 8.7.4)

The range 239.192.0.1 to 239.192.0.64 is reserved for current and future use in the application layer protocols (see 6.2.2).

The multicast address range 239.192.0.57 to 239.192.0.64 is used for interconnection with IEC 61162-3 networks.

ONF equipment shall not use multicast addresses in the range 239.192.0.1 to 239.192.0.64.

NOTE 1 ISOC RFC 2365 defines the multicast address range 239.192.0.0 to 239.192.63.255 as the IPv4 Organization Local Scope, and is the space from which an organization should allocate sub-ranges when defining scopes for private use.

NOTE 2 The default TTL (i.e. number of hops) is 1 for multicast. The sub-net mask is set appropriately for class C (local area network).

5.5 Device address for instrument networks

Means shall be provided to assign a device address range from 0 to 251 when the PNGF transmits to an IEC 61162-3 network. The device address may be set automatically.

6 Transport layer specification

(see 8.8)

6.1 General

Clause 6 specifies how UDP multicast messages are used to communicate between equipment over an Ethernet network.

Equipment may implement functionality for sending, receiving or both. The provisions of Clause 6 applies to both, but shall be tested independently as described in 7.6.

An example of the structure of an Ethernet frame with a IEC 61162-450 sentence is given in Figure 2. The uppermost block shows the full Ethernet frame with the UDP user available data block shown in white. The IP and UDP headers are included in the grey blocks. The lower block shows the UDP user available data block with an IEC 61162-450 formatted sentence included. The numbers above the Ethernet frame gives the size of each block. The numbers in front of the UDP user data block gives the offset from the start of the block (0 – zero).

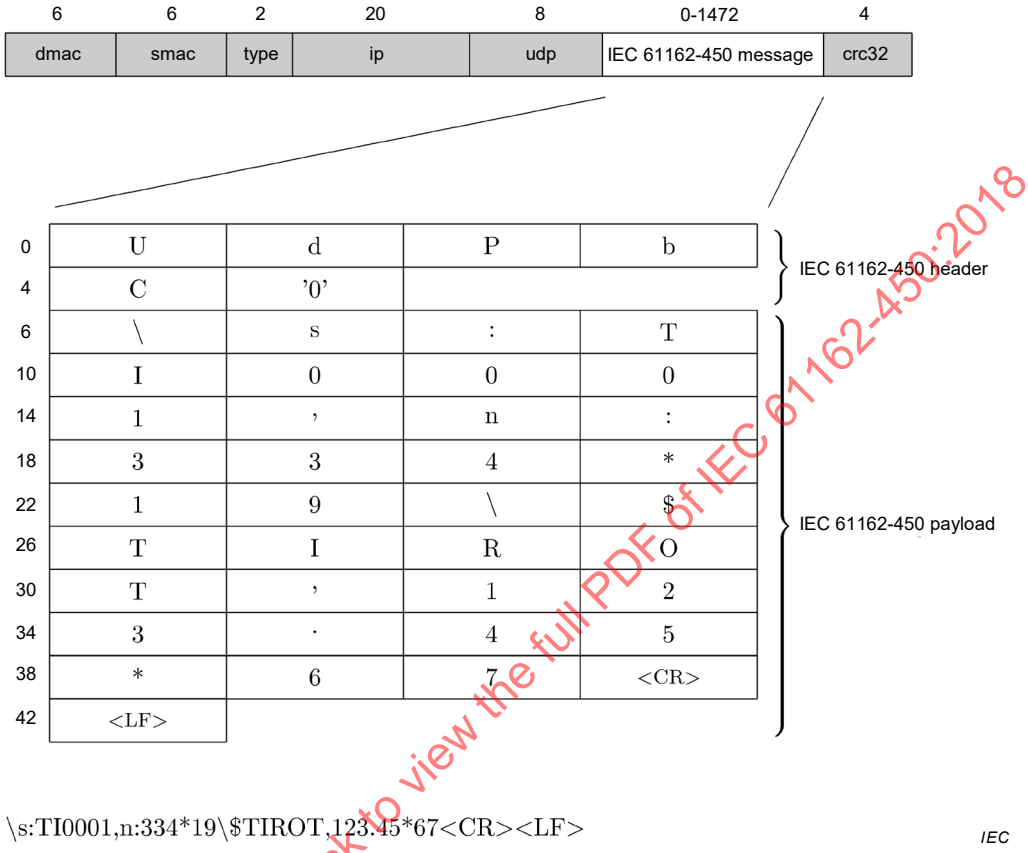


Figure 2 – Ethernet frame example for a SBM from a rate of turn sensor

6.2 UDP messages

6.2.1 UDP multicast protocol

UDP Multicast – IP multicast is a technique for many-to-many communication over an IP infrastructure in a network. The destination nodes send join and may send leave messages. IP multicast scales to a larger receiver population by not requiring prior knowledge of who or how many receivers there are. Multicast uses network infrastructure efficiently by requiring the source to send a packet only once, even if it needs to be delivered to a large number of receivers. The nodes in the network take care of replicating the packet to reach multiple receivers only when necessary. The most common transport layer protocol to use multicast addressing is User Datagram Protocol (UDP).

Senders and receivers shall as a minimum be able to use UDP as defined by ISOC RFC 768 and as further specified in this document.

6.2.2 Use of multicast addresses and port numbers

Port numbers shall be allocated from the dynamic port range that the internet assigned number authority (IANA) has reserved for dynamic and/or private port numbers (range 49152 to 65535, inclusive).

Table 4 defines multicast addresses and destination port numbers that shall be used when transmitting sentences from a system function block. The mapping of SFI to default transmission group is described in Annex A. If provided by the equipment, the default multicast address or destination port number can be changed by the parameter setup system of the equipment to the multicast addresses or destination port numbers of the transmission groups USR1 to USR8, RCOM, PROP in Table 4 or any in Table 5 (for example to support use of same transmission group for both "binary file" and related sentences, for example ECDIS route exchange and RRT-sentence).

NOTE The purpose of the port differentiation is to provide a mechanism that allows a certain level of load reduction for the receiving equipment.

Table 4 – Destination multicast addresses and port numbers

Transmission group	Category	Multicast address	Destination port
MISC	SF not explicitly listed below	239.192.0.1	60001
TGTD	Target data (AIS), tracked target messages (Radar)	239.192.0.2	60002
SATD	High update rate, for example ship heading, attitude data.	239.192.0.3	60003
NAVD	Navigational output other than that of TGTD and SATD groups	239.192.0.4	60004
VDRD	Data required for the VDR according to IEC 61996	239.192.0.5	60005
RCOM	Radio communication equipment	239.192.0.6	60006
TIME	Time transmitting equipment	239.192.0.7	60007
PROP	Proprietary and user specified SFs	239.192.0.8	60008
USR1 to USR8	User defined transmission group 1 to 8	239.192.0.9 to 239.192.0.16	60009 to 60016
BAM1 to BAM2	Optionally, BAM compliant alert source reporting to CAM	239.192.0.17 to 239.192.0.18	60017 to 60018
CAM1 to CAM2	CAM of the BAM	239.192.0.19 to 239.192.0.20	60019 to 60020
NETA	Network administration, e.g. SFI collision detection	239.192.0.56	60056
PGP1 to PGP4	Primary PGN Group 1 to PGN Group 4	239.192.0.57 to 239.192.0.60	60057 to 60060
PGB1 to PGB4	Backup PGN Group 1 to PGN Group 4	239.192.0.61 to 239.192.0.64	60061 to 60064
NOTE 1 The USR1 to USR8 transmission groups can be used, for example, for proprietary data in binary format. NOTE 2 BAM1/BAM2 and CAM1/CAM2 are available for system integrators to balance the traffic, for example higher volume radar in BAM1/CAM1 and low volume sensor, for example gyro, in BAM2/CAM2.			

Table 5 defines multicast addresses and destination port numbers that shall be used when transmitting binary file data. If provided by the equipment, the default multicast address or destination port number can be changed by the parameter setup system of the equipment to the multicast addresses or destination port numbers of the transmission groups USR1 to USR8, RCOM, PROP in Table 4 or any in Table 5 (for example to support use of same transmission group for both "binary file" and related sentences).

Table 5 – Destination multicast addresses and port numbers for binary data transfer

Category	Multicast address	Destination port
Non re-transmittable binary file transfer [a]	239.192.0.21 to 239.192.0.25	60021 to 60025
Re-transmittable binary file transfer [b]	239.192.0.26 to 239.192.0.30	60026 to 60030
[a] Address 239.192.0.25, port 60025 is the default for ECDIS route transfer (see IEC 61174). [b] Address 239.192.0.26, port 60026 is the default for VDR image transfer (see IEC 61996-1). Address 239.192.0.30, port 60030 is the default for ECDIS re-transmittable data blocks for route transfer (see IEC 61174).		

Table 6 lists other multicast addresses and ports reserved by this document.

Table 6 – Destination multicast addresses and port numbers for other services

Category	Multicast address	Destination port
Syslog	239.192.0.254	514
Sending to syslog can use multicast or UDP unicast. Some switches can support only UDP unicast.		

The addresses 239.192.0.31 to 239.192.0.55 are reserved for future expansion.

It may be noted that IANA has defined that port range 49152 to 65535 is reserved for dynamic and private use. The specific ports for this document are within this IANA range. It should be noted that operating systems also use this IANA range for their internal use as ephemeral ports. This double use may cause port number conflicts resulting for lost communication of IEC 61162-450 messages. It is recommended to consider limiting the ephemeral port range of the operating system of equipment connected to an IEC 61162-450 network to avoid port number conflicts.

6.2.3 UDP checksum

All devices shall calculate and check the UDP checksum as defined by ISOC RFC 768. It is not permitted to set the checksum field to zero (no checksum).

A datagram that has an incorrect or missing checksum shall be discarded by the receiver.

6.2.4 Datagram size

The network function block shall not transmit more than 1 472 bytes of data in each datagram, including header as defined in Clause 7.

Receiving equipment is allowed to discard datagrams that have a size larger than the maximum specified size.

NOTE UDP datagrams can be up to 64 kB in size when they are sent as a number of IP fragments.

7 Application layer specification

7.1 Datagram header

(see 8.9.2)

7.1.1 Valid header

All UDP multicast datagrams shall contain one of the following strings, followed by a null character (all bits set to zero) as the first six bytes of the datagram:

- "UdPbC" for transmission of IEC 61162-1 formatted sentences as described in 7.2;
- "RaUdP" for transmission of binary files as described in 7.3;
- "RrUdP" for transmission of re-transmittable binary files as described in 7.3;
- "NkPgN" for transmission of IEC 61162-3 PGN messages as described in 7.4;

All TCP/IP datagrams shall contain the following string, followed by a null character (all bits set to zero) as the first six bytes of the datagram:

- "RrTcP" for transmission of binary files as described in 7.6;

NOTE 1 Datagram means packet in this context.

Incoming datagrams with an unknown header should be discarded without processing the content beyond the header.

NOTE 2 Future editions of this document can define other header codes. Any such header code will be different from the ones already in use and will at least contain six bytes, possibly including a trailing null character.

7.1.2 Error logging

The equipment shall maintain a count of received datagrams that do not have a valid header and make this available as defined in 4.3.3.

7.2 General IEC 61162-1 sentence transmissions

7.2.1 Application of this protocol

(see 8.9.1)

This protocol provides a mechanism by which IEC 61162-1 sentences can be sent to one or more receivers on the network. The protocol allows several sentences to be merged into one datagram.

7.2.2 Types of messages for which this protocol can be used

(see 8.9.3)

This protocol shall be used for SBM and MSM (see Annex A) type messages. The protocol shall also be used for CRP message exchanges with provisions specified in Annex C.

7.2.3 TAG block parameters for sentences transmitted in the datagram

(see 8.9.4)

7.2.3.1 Valid TAG block

Each sentence shall be preceded with one or more TAG blocks as defined in NMEA 0183:2008, Section 7 (see also Annex B), containing the parameter codes described in 7.2.3.3 to 7.2.3.8. Adding of TAG blocks with parameter codes happens between existing TAG blocks with parameter codes and the start of IEC 61162-1 sentence. If a parameter code is assigned a value more than once in the TAG blocks and only one value is expected, the last

parameter value (i.e. parameter value closest to the start of IEC 61162-1 sentence) shall be used.

In this document, all identities are set at the time of installation and shall not be dynamically configurable during normal operation. The control sentences for changing parameter codes in NMEA 0183 shall not be used during normal operation.

7.2.3.2 TAG block checking

Only sentences preceded by valid TAG blocks as defined in 7.2.3.1 shall be processed by the receiver.

7.2.3.3 Grouping control – g

The g parameter code shall be used by talkers to group TAG blocks and/or sentences. As a minimum, it shall be used to group sentences that are classified as belonging to message type "MSM" in Table A.2, when the multi-sentence group consists of more than one message. It is not required to include the g parameter code for single line sentences.

Receivers shall accept the g parameter code for all message types.

A valid MSM type sentence where internal data fields specifies that it belongs to a group of more than one message shall be discarded if the g group is missing or contains inconsistent information.

The group code is determined by the sending device. The initial group code value shall be one ("1") and the group code increment value shall be one ("1"). The group code shall be reset to one ("1") after it reaches 100, i.e., the valid range is 1-99, inclusive.

The following example shows the g parameter code used to group sentences in two different groups, each consisting of two sentences.

```
\g:1-2-34,s:IN0001*3A\!ABVDM,1,1,1,B,100000?0?wJm4:`GMUrf40g604:4,0*04
\g:2-2-34,s:IN0001*39\$ABVSI,r3669961,1,013536.96326433,1386,-98,,*14
\g:1-2-46,s:IN0001*3F\!ABVDM,1,1,1,B,15N1u<PP1cJnFj:GV4>:MOw:0<02,0*2D
\g:2-2-46,s:IN0001*3C\$ABVSI,r3669962,1,013538.05654921,1427,-101,,*20
```

7.2.3.4 Source identification – s

The s parameter code is mandatory for talkers and shall contain the system function ID (SFI, see 4.4.2) corresponding to the function block from where the sentence originates.

7.2.3.5 Destination identification – d

The d parameter code is optional and shall, if used, contain the system function ID (SFI, see 4.4.2) corresponding to the intended recipient of the sentence. If no destination parameter code is present, then all devices that receive this sentence shall process it.

Multiple d parameters may be specified, if more than one receiver exists.

NOTE This can be the case for redundant control functions.

For CRP type sentences, the destination code shall be read and processed to ensure that only the intended recipients take action on the content of the sentence. Other receivers may also read the message, for example for voyage data recording purposes, but shall not take any further action on the contents.

7.2.3.6 Line-count parameter – n

The n parameter code may be used to assign a sequence number to each sentence transmitted from a system function block. The format of the parameter value is a positive integer. The value shall start at one ("1") and shall be incremented by one ("1") for each sentence or TAG block transmitted from this system function block. The parameter value shall be reset to one ("1") when it reaches 1 000, i.e., the valid range is 1-999, inclusive.

For function blocks that transmit datagram to more than one transmission group destination, separate line counters shall be maintained for each transmission group (see 6.2.2).

7.2.3.7 Text string parameter – t (proprietary data)

The t parameter code is a free text field. This document reserves coding for proprietary TAG-codes with the fields defined below where the leading p and the three letter manufacturer mnemonic code is required for this type of text string.

```
t:p<manufacturer mnemonic code in lower case><proprietary data>
```

An example used for proprietary authentication of lines using grouping and source for manufacturer "mmm" might be

```
\g:1-2-34,s:TI0001,n:333*6B\$TIROT,123.45*67
\g:2-2-34,s:TI0001,n:334,t:pmmma;MD5;0x12345678*0D\
```

7.2.3.8 General authentication – a

(see 8.9.5)

The authentication parameter code is used to sign a message with a password. Just sending a password with the message would reveal the password to anyone listening to the traffic. Sending a signature digest instead keeps the password secret.

Any kind of messages may be signed using the authentication parameter code. The authentication parameter code does not change the original message in any way. It is always possible to ignore this tag and use the rest of the message.

NOTE 1 For example, one can sign configuration commands for devices or commands to the autopilot.

The authentication parameter code provides a standardized mechanism for passing the digest with the message. Password management is outside of the scope of this document. One way is to use pre-shared keys (PSK) on the participating devices.

NOTE 2 The pre-shared key could be 32 alphanumeric characters, for example "Alea iacta est 1234567890".

This parameter code is optional and should only be used where special safety concerns make it useful. If this tag is provided then the manufacturer's documentation shall describe which of the optional types of methods to calculate a signature are supported by the equipment and shall describe how to share keys.

The format of the tag block is:

\a:c-h--h*hh\

In which

c

Type of optional method to calculate signature

1) MD5

P) Proprietary

h-h

Hexadecimal representation of the signature, for example 32 hexacodes for MD5

An example of the tag block is:

\a:1-123456789abcdef67890123456789012*hh\

Types of methods to calculate signature:

1) MD5

The signature is a MD5 digest of the password plus the message. MD5 is a one-way message-digest algorithm (RFC 1321). The full length of the signature is 128 bits or 32 hexadecimal codes. The MD5 is commonly used for storing passwords in Unix-systems. Revealing the digest does not expose the password.

NOTE 3 See http://tools.ietf.org/html/rfc1321 and http://en.wikipedia.org/wiki/MD5.

P) Proprietary

The signature is a proprietary digest of the password plus the message. This alternative requires that both parties use the same manufacturer specified proprietary method.

The authentication parameter code value is calculated by concatenating pre-shared key and all TAG blocks and sentences in the message as a single string to be used by the method of the signature calculation to produce the signature digest. "Carriage returns" and "line feeds" from the sentences are not included into the input string.

When the authentication parameter code "a:" is used, it shall be in its own authentication TAG block, with no other parameter codes. For a grouped message consisting of several lines of TAG blocks and sentences, the authentication TAG block shall be placed on the first line of the group. Within the first line, the authentication TAG block shall be placed as the last TAG block, and before any sentence on that line. This also applies to a single line TAG block and sentence with no grouping.

An example of use of authentication TAG block:

Message consisting of two grouped sentences to be protected by authentication:

\g:1-2-23,s:IN0001*3C\!ABVDM,1,1,1,B,15N1u<PP1cJnFj:GV4>:MOw:0<02,0*2D

\g:2-2-23,s:IN0001*3F\$ABVSI,r3669962,1,013538.05654921,1427,-101,,*20

Pre-shared key to be used for signature calculation:

Alea iacta est 1234567890

Resulting input string for signature calculation:

Alea iacta est 1234567890\g:1-2-23,s:IN0001*3C\!ABVDM,1,1,1,B,15N1u<PP1cJnFj:GV4>:MOw:0<02,0*2D\g:2-2-23,s:IN0001*3F\$ABVSI,r3669962,1,013538.05654921,1427,-101,,*20

Message to be sent including signature, method MD5:

\g:1-2-23,s:IN0001*3C\\a:2-851E40CC1CB7E3B39D961D7CF10BD8D3*44\!ABVDM,1,1,1,B,15N1u<PP1cJnFj:GV4>:MOw:0<02,0*2D

\g:2-2-23,s:IN0001*3F\$ABVSI,r3669962,1,013538.05654921,1427,-101,,*20

Messages without authentication parameter codes are accepted unless the set-up parameters of the receiver are explicitly set to require authentication on incoming packets.

If the device is set to require authentication on incoming packets, then packets without valid authentication shall be dropped.

NOTE 4 SNGF are advised to avoid transmitting passwords in the clear from SPW sentences received over a serial connection.

7.2.4 Requirements for processing incoming datagrams

For datagrams intended for processing by the SF, any syntax error in a TAG block or in a sentence shall make the receiving equipment discard the complete datagram without any other further processing than specified in 7.2.5. The exception is an SNGF which shall retransmit the faulty sentences to the appropriate serial SF, if it can be determined from a valid destination field, or to all connected serial SFs, if no destination field is specified.

7.2.5 Error logging for processing incoming datagrams

(see 8.10)

The equipment shall maintain counts of errors detected in processing datagrams containing IEC 61162-1 sentences. As a minimum, the following errors shall be counted and made available as defined in 4.3.3:

- any TAG block formatting errors as defined in 7.2.3.1;
- TAG checksum error;
- TAG syntax error (line length, use of delimiters, invalid characters);
- TAG framing error (incorrect start or termination of TAG block);
- any sentence syntax errors, including formatting, length or checksum as defined in 7.2.4.

7.3 Binary file transfer using UDP multicast – Single transmitter, multiple receivers

(see 8.11)

7.3.1 Application of this protocol

This protocol provides a mechanism by which non IEC 61162-1 formatted data, for instance radar images as files, can be transmitted to one or more receivers. This protocol supports the transmission of files from zero bytes up to 4 billion files blocks.

Equipment using this mechanism shall be able to use one or both of the following forms of binary file transfer:

- non re-transmittable transfers where sender sends the complete binary file without any feed-back from receiver;
- re-transmittable transfers where limited feed-back from one receiver identified by DestID can be used to re-transmit certain parts of the binary file while other parallel receivers operate as passive receive-only receivers of the binary file.

NOTE The advantage of non-re-transmittable and re-transmittable binary file transfer methods over the TCP/IP is the possibility of multiple parallel receivers of the same transmission.

Table 7 gives a description of terms used in this application.

Table 7 – Description of terms

Term	Description
DWORD	Double Word. One unsigned 32-bit integer (in range 0 to 4294967295). The DWORD is constructed from four consecutively transmitted BYTE, where the transmission order on the network is most significant BYTE first followed by next most significant BYTE until the least significant BYTE.
Null character	A BYTE with the value zero.
Reserved bytes	A number of bytes in the datagram that may be ignored by the receiver. The reserved bytes may be additional header information that only has meaning for newer versions of the protocol.
WORD	One unsigned 16-bit integer (in range 0 to 65535). The WORD is constructed from two consecutively transmitted BYTES, where the transmission order on the network is the most significant BYTE followed by the least significant BYTE.
STRING[n]	A sequence of exactly *n* BYTE, interpreted as a string of characters. The transmission order on the network is left-most character first. If the string is shorter than *n*, additional trailing bytes shall be set to null character. All strings in the header are encoded in ISO/IEC 8859-1 (ISO Latin 1).

7.3.2 Binary file structure

7.3.2.1 General

The binary files are transmitted over the network in one or more datagrams. The binary file structure is a sequential and unpadded stream of bytes divided into three main groups: header, binary file descriptor and binary file data (see Table 8 and Table 9). The header is needed for synchronisation and data integrity validation. The binary file descriptor is needed for the description of the binary file data and is only used in the first datagram for each binary file transfer.

7.3.2.2 Non re-transmittable and re-transmittable transfers

Table 8 – Binary file structure

61162-450 header (see 7.3.4)
Binary file descriptor (only in first datagram) (see 7.3.6)
Binary file data fragment (see 7.3.7)
61162-450 header (zero or more)
Binary file data fragment (zero or more)

A minimum binary file transmission using non re-transmittable or re-transmittable transfer will consist of the three first blocks where the binary file fragment may have zero length.

The header shall be repeated as the first element of any datagram that contains binary file data fragments.

7.3.3 61162-450 header

7.3.3.1 Header format

The purpose of the header is to provide the data transfer status to receivers. This allows a receiver to identify if there is any data loss during binary file transfers, and how much data loss occurs. In addition, the header is used to provide a re-transmission mechanism for re-transmittable binary file transfer.

The 61162-450 header format is defined in Table 9.

Table 9 – 61162-450 header format

Data item	TYPE	Description
Token	STRING[6]	Identifier as ASCII string with a length of 5 bytes followed by a null character (see 7.1.1).
Version	WORD	Defines the header version. The header version with value 2 is defined in this document. Extensions and/or modified versions may update this value.
HeaderLength	WORD	Defines the length of the header in bytes. This is at least the length of the header. Future editions of this document may append additional fields to this header as long as these additional fields are compatible with the definition of the header in this document. Receivers which are not aware of these additional fields shall ignore them.
SrcID	STRING[6]	Define the source system identifier in format "ccxxxx" (see 4.4.2).
DestID	STRING[6]	For re-transmittable, defines the destination system identifier in format "ccxxxx", for example "VR0001" for VDR (see 4.4.2). When Destid = "XXXXXX", then there is no assigned destination.
Type	WORD	Identifies the information in the Header.
BlockID	DWORD	Binary file block identifier. The initial value is randomly generated within a range 0 to (2^{32} – 1 = 4294967295) and is incremented by 1 after a whole block is transmitted.
SequenceNum	DWORD	Defines the sequence number of the binary file block. In ACK, this is used to inform the sender what block was last received.
MaxSequence	DWORD	The number of datagrams needed for the transmission of this binary file data block. When SequenceNum is equal to MaxSequence, it means that this datagram is the last datagram of the data block. The Maxseq is used only for DATA type message. For other messages (QUERY,ACK), this field shall be 0.
Device	BYTE	Data source (device) as binary value, 1 => equipment 1, 2 => equipment 2, etc. The value can be between 1 and 255
Channel	BYTE	Subdivision according to data source (device), values from 1 to 255, default = 1

The Device and Channel fields are defined by the application and may be used by receivers to determine how to process the binary file data.

7.3.3.2 Use of header token

Header token is used to identify both the type of data block and transfer mode not be used to accept or reject transmissions. Two tokens are defined in 7.1.1:

- "RaUdP" – Simple binary file transfer service with UDP Multicast;
- "RrUdP" – Re-transmittable binary file transfer service with UDP Multicast.

7.3.3.3 Version

Defines the header version. It shall be set to 2 for this document.

7.3.3.4 Destination identifier

For transmissions to one specific receiver, the field shall contain the destination SFI. The field shall be "XXXXXX" for no specific destination.

7.3.3.5 Message type

Message type gives the information about which information is contained in the datagram:

- DATA (0x01) – This type is used for transmission of binary file data including file descriptor.
- QUERY (0x02) – This type is used by the sender to query the reception status from the receiver. The length of this message payload is always zero (0). It is recommended that a

binary file sender sends a QUERY message if there is no ACK message for 1 s after a last datagram of the binary file block is sent or after a QUERY message is sent.

- ACK (0x03) – This message is used as an acknowledgement from the receiver. This message is transmitted by the receiver either when a whole binary file is received without any error or when errors occurred during the binary file reception, for example one sequence number is skipped. Also, when a receiver receives a QUERY message from the sender, it also responds with an ACK message.

Non re-transmittable transfer makes use of only DATA message but re-transmittable transfer uses all messages.

7.3.3.6 Binary file block identifier

Block identifier is used to identify each binary file block. Since a binary file block is fragmented into several datagrams, the block identifier is used to assemble one or more datagrams into a binary file block in a receiver.

7.3.3.7 Sequence number and maximum sequence number

Sequence number (SequenceNum) and maximum sequence number (MaxSequence) is used for segmentation and re-assembly purposes. When a receiver gets a datagram, it checks the sequence number and maximum sequence number to determine if any errors have occurred or if it has received a whole message.

The sequence number is also used in ACK messages. In ACK messages, the sequence number identifies the last message the receiver receives without any error. The maximum sequence number is not used for control (Query) messages.

7.3.3.8 Identification of separate binary file transfer

Each single binary file transfer shall be identified by a unique combination of SrcID, Device, Channel and BlockID (see Table 9).

NOTE If a single SrcID has multiple needs to send binary files (e.g. ECDIS sending screen image, chart source information and route exchange), then each single binary file transfer is identified, for example: ECDIS number 1 send screen image as Device = 1 and Channel = 1, and Chart source information as Device = 1 and Channel = 2.

7.3.4 Binary file descriptor structure

The binary file descriptor format is defined in Table 10.

Table 10 – Binary file descriptor format

Data item	TYPE	Description
Length	DWORD	Defines the binary file descriptor length in bytes. This is at least the length of the header including the reserved bytes. Future editions of this document may append additional fields to this file descriptor as long as these additional fields are compatible with the definition of the file descriptor in this document. Receivers which are not aware of these additional fields shall ignore them.
fileLength	DWORD	Defines the length of the full binary file content in bytes, excluding headers and descriptor.
Status of acquisition	WORD	The status for the data return. A zero is returned for normal operation. Non-zero value is used to indicate an error condition. A descriptive text may be put in the status and information text field.
AckDestPort	WORD	Port number to be used to acknowledge. Allowed port numbers are within the range from 60006, 60008 to 60016, 60021 to 60030 (see 7.3.8.9).
TypeLength	BYTE	The length of the DataType field.
DataType	STRING[n]	This string defines the data block encoding by assigning a MIME content type to the data block for the server followed by null character. For example, "image/jpeg" is used for JPEG image type.
StatusLength	WORD	The length of the "Status and information text" field in bytes.
Status and information text	STRING[n]	Status information (e.g. successful operation or error codes). This may be one or more strings, each terminated by a binary null
NOTE 1 There is no error check for the binary file header contents as this is handled by the UDP layer. In this document, UDP header checksum is mandatory. NOTE 2 MIME is Multipart Internet Mail Extensions. The MIME content type was originally used for email services but is widely used for many other applications including Web. Also, it has flexibility to support new media types. The specification of the MIME content type and registration is defined in ISOC RFC 4288 and ISOC RFC 4289.		

DataType shall be encoded by the MIME content-type which is "type/sub-type", and is defined by IANA. Table 11 illustrates some examples of MIME content type for binary file and compressed data. More updated information is available on the IANA web site, http://www.iana.org/assignments/media-types/.

Table 11 – Examples of MIME content type for DataType codes

Content type	File extension	MIME type/sub-type
GIF	gif	image/gif
Microsoft Windows bitmap	bmp	image/x-ms-bmp
Gnu tar format	gtar	application/x-gtar
4.3BSD tar format	tar	application/x-tar
DOS/PC – Pkzipped archive	zip	application/zip
XML	xml	application/xml

7.3.5 Binary file data fragment

The package data format is defined in Table 12.

Table 12 – Binary file data fragment format

Data item	TYPE	Description
Datablock	BYTE[datalength]	This item is the data either split into pieces or in one block.

The length of the binary file fragment is the length of the UDP datagram (as obtained from the UDP header) minus any headers that are inserted in front of the binary file fragment. All datagrams, except the first datagram of the binary file which requires two headers (Header + binary File Descriptor), carry only one header (Header).

The binary file fragment length is allowed to be zero for one or more datagrams.

NOTE There is no error check for the data contents as this is handled by the UDP layer.

7.3.6 Sender process for binary file transfer

7.3.6.1 Non re-transmittable sender process

The following steps are performed for the basic sending process (see Figure 3):

a) a sender process waits until it gets a binary file block;

b) a block identifier is assigned for the binary file block (if this is the first binary file, then it is assigned randomly; otherwise, the instance identifier of the previous binary file block + 1 is used). The BlockID shall be unique for each binary file transfer from the same SrcID, Device and Channel combination;

c) a binary file descriptor is composed according to Table 10;

d) a binary file block is split into datagrams whose size is not more than 1 472 bytes and each datagram is put into the sending buffer;

e) get the first datagram of the binary file block;

f) assign a sequence number, which is assigned to one initially;

g) compose a header including token, source ID, destination ID and maximum sequence number according Table 9;

h) send a datagram to the network;

i) if all datagrams of the binary file block are not transmitted, get the next datagram and go to step f);

j) otherwise, then go to step a).

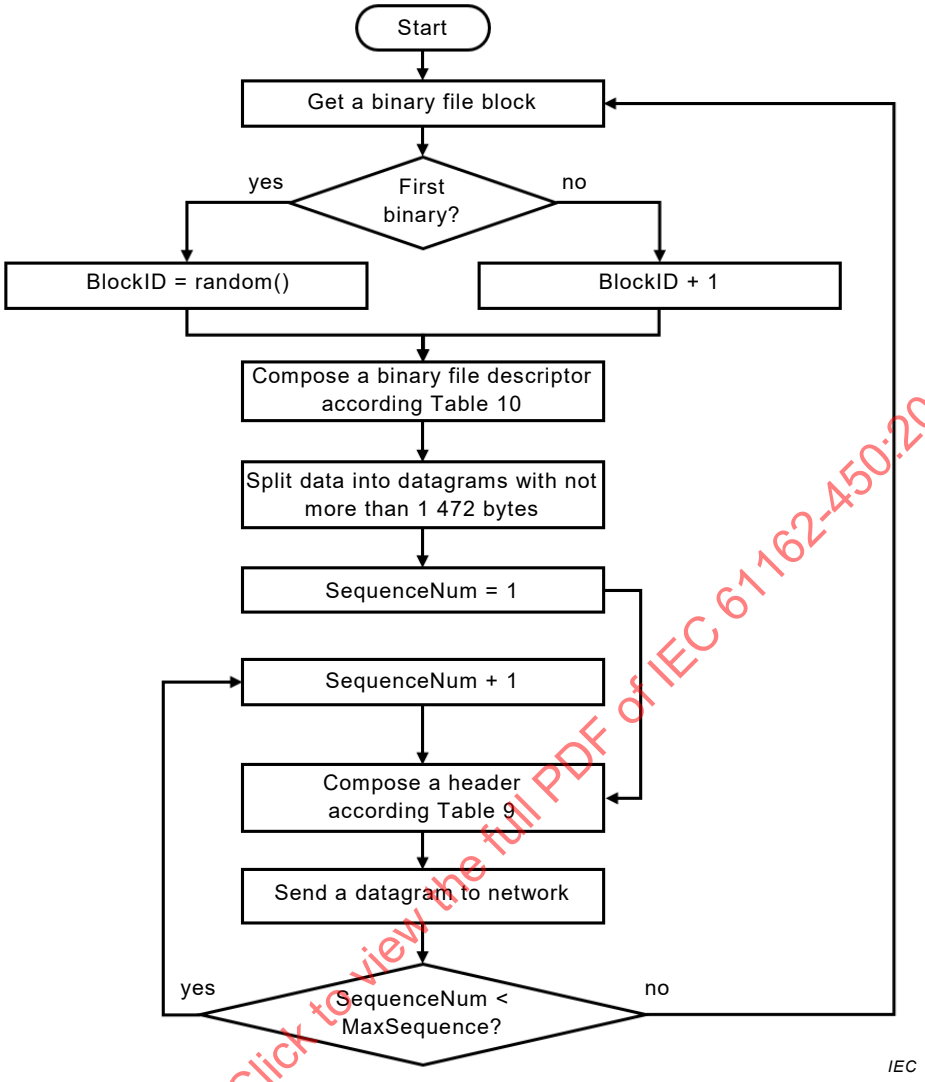


Figure 3 – Non re-transmittable sender process

7.3.6.2 Re-transmittable sender process

The sender processing steps for re-transmittable binary file transfer is as follows (see Figure 4):

a) a sender process waits until it gets a binary file block;

b) a block identifier (BlockID) is assigned for the binary file block (if this is the first binary file, then it is assigned randomly; otherwise, the block identifier of the previous binary file block + 1 is used). The BlockID shall be unique for each binary file transfer from the same SrcID;

c) a binary file descriptor is composed according to Table 10;

d) set re-transmission counter to zero(0), set query counter to zero(0), set query sequence number to 1, set final counter to zero(0);

e) a binary file block is split into datagrams whose size is less than 1 472 bytes and each datagram is put into the sending buffer;

f) get the first datagram of the binary file;

g) assign a sequence number, which is set to one initially;

h) compose a header according Table 9;

i) send a datagram to the network and set an ACK timer;

j) if the sender receives an ACK message, whose DestID is not equal to own SFI and whose SourceID is not equal to own actual DestID, go to step k);

   1) if the sequence number of ACK message is less than the maximum sequence number and lower than the sequence number of the last transmitted datagram, increase re-transmission count by one;

   2) if re-transmission count is greater than the maximum number of retransmissions (see 7.3.8.7), go to step k);

   3) if sequence number in ACK message is identical to query sequence number, increase query counter with 1, and if query counter is more than 3, set query counter to zero (0) and go to k);

   4) if sequence number in ACK message is not identical to query sequence number, set query counter to 1;

   5) get a datagram whose sequence number is sequence number in ACK message plus one;

   6) set query sequence number to sequence number;

   7) go to step h);

k) if all datagrams of the binary file block have not been transmitted,

   1) get a next datagram and increase sequence number by one,

   2) go to step h);

l) otherwise, wait for an ACK message;

m) if the sender receives an ACK message whose DestID is not equal to own SFI and whose SourceID is not equal to own actual DestID , then go to step (n);

   1) if the sequence number of the ACK message is less than the maximum sequence number, then go to step j);

   2) if the sequence number of the ACK message is equal to the maximum sequence number (i.e. transfer successful), then go to step a);

n) if ACK Timer expires and final counter is not more than three, then

   1) increase the final counter,

   2) compose a QUERY datagram and send it to the network,

   3) set an ACK timer

   4) go to step l); clear the sending buffer,

   5) go to step a).

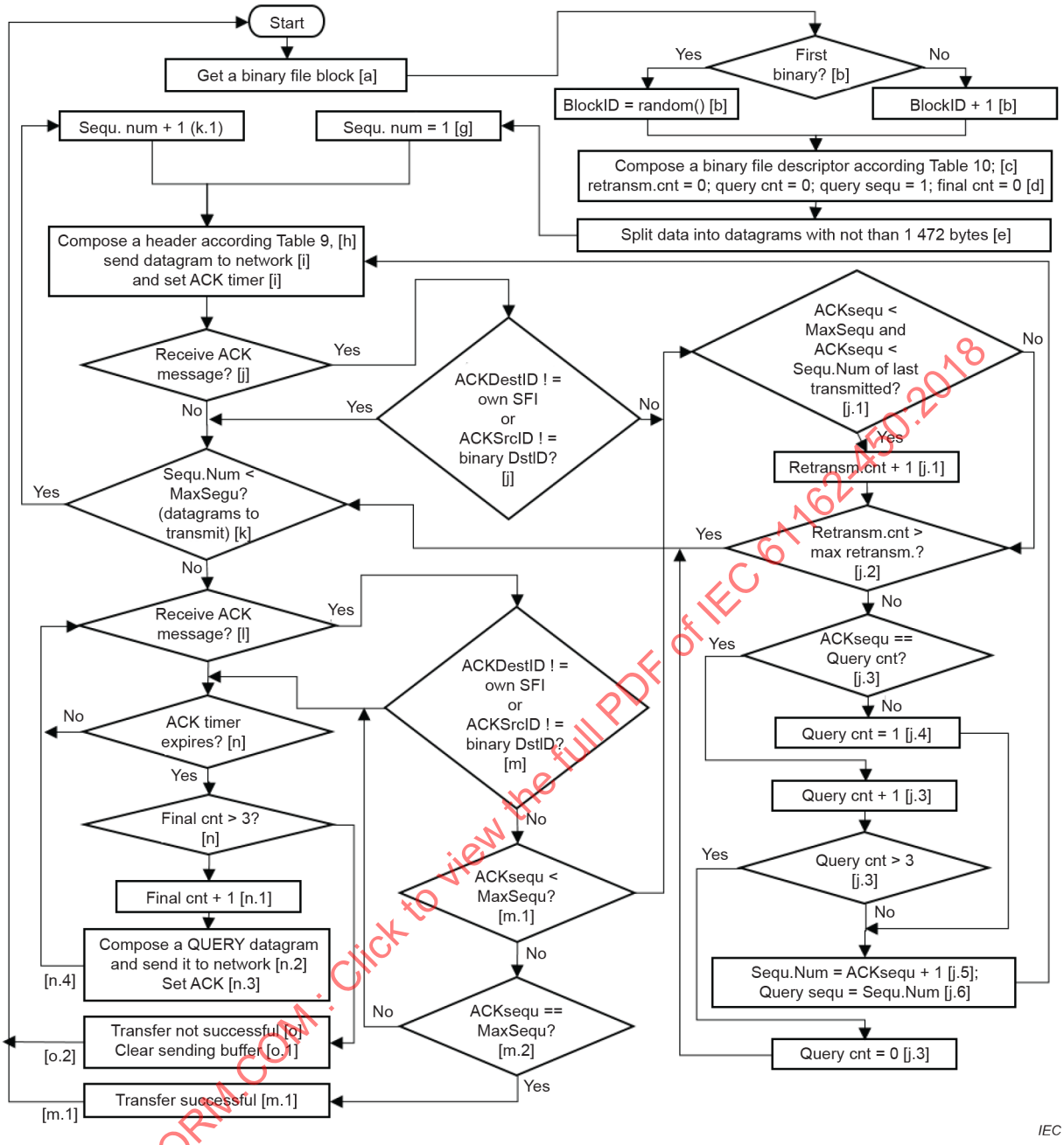


Figure 4 – Re-transmittable sender process

7.3.7 Receiver process for binary file transfer

7.3.7.1 Non re-transmittable receiver process

The receiver process steps of the non re-transmittable binary file transfer, including passive receivers of a re-transmittable binary file transfer, is as follows:

a) waits for receiving new datagram;

b) if the BlockID of the received datagram for same source identified by the combination of SrcID, Device and Channel is not equal to that of the previous datagram,

- if there is any data in the receiver buffer, it is delivered to the SF,
- the receiver buffer is cleared;

c) put a datagram into the receiver buffer;

d) if the sequence number is the same as the maximum sequence number,

- the all data in the received buffer is delivered to the SF,
- the receiver buffer is cleared;

e) go to step a).

7.3.7.2 Re-transmittable receiver process

The re-transmittable receiver process steps are performed only by the receiver whose SFI is same as the DestID in the Header as follows (see Figure 5):

a) waits for receiving a new datagram;

b) if the token is not "RrUdP" or if DestID of received datagram is not equal to own SFI, go to step a);

c) if the received datagram is a QUERY message, then,

1) if no previous datagram is available, go to step a),

2) if the sequence number of the previous datagram is not equal to maximum sequence number, go to step a),

3) if all sequences flag is false (not all sequences of the previous binary block are received), go to step a),

4) compose a Header with type = ACK, the BlockID and sequence number of the previous datagram,

5) send an acknowledge datagram to the sender,

6) go to step a);

d) if the sequence number is 1,

1) analyse file image descriptor and header,

2) if file image descriptor or header or token is invalid, go to step a),

3) set re-transmission counter and query counter to zero (0), set query sequence number to 1, set all sequences flag to true,

4) go to step g);

e) if no previous datagram is available, go to step g);

f) if the BlockID of the received datagram for same source identified by the combination of SrcID, Device and Channel is not equal to that of the previous datagram,

1) if there is any data in the receiver buffer, it is delivered to the SF,

2) the receiver buffer is cleared;

g) if the sequence number is not 1 and not the same as the sequence number of the previous datagram plus one,

1) increase re-transmission count by one,

2) if re-transmission count is greater than the maximum number of retransmissions (see 7.3.8.7), set all sequences flag to false and go to step (a),

3) if previous sequence number is identical to query sequence number increase query counter with 1,

4) if query counter is more than 3 set query counter to zero (0),

5) set all sequences flag to false and go to step (a),

6) if previous sequence number is not identical to query sequence number, set query counter to 1,

7) set query sequence number to previous sequence number,

8) compose a Header with type = ACK, the block identifier and sequence number of the previous datagram,

9) send an acknowledge datagram to the sender,

10) go to step (a);

h) put a datagram into the receiver buffer;

i) if the sequence number is same as the maximum sequence number,

   1) all the data in the receive buffer is delivered to the SF,

   2) the receiver buffer is cleared,

   3) the re-transmission count is set to zero (0);

j) if all sequences flag are true,

   1) compose a Header with type = ACK, the block identifier and maximum sequence number,

   2) send an acknowledge datagram to the sender;

k) go to step a).

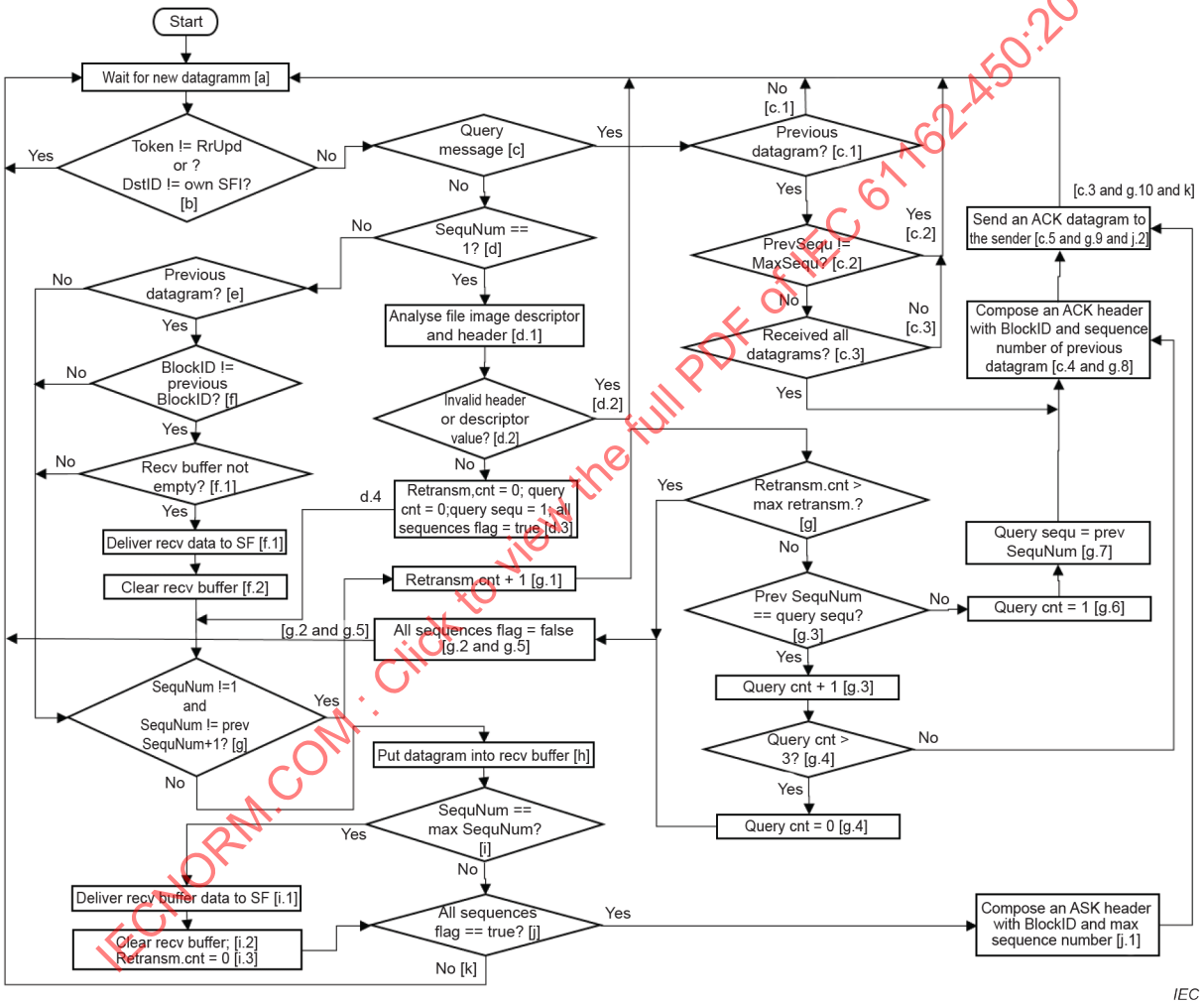


Figure 5 – Re-transmittable receive process

7.3.8 Other requirements

7.3.8.1 Re-transmittable messages that cannot be processed

Both receiver and sender shall silently ignore messages that are related to the retransmit process that they cannot process themselves.

7.3.8.2 Multiple binary file blocks

A receiver that receives a binary file block more than once shall ignore all but one of the transmissions.

It is allowed both to ignore the first (overwrite buffer) or the last (ignore).

7.3.8.3 Retransmissions size

If a sender retransmits one or more binary file blocks, each of the blocks shall have the same size and same header information.

7.3.8.4 Maximum outgoing rate

The data volume for each binary file source shall not exceed 2 MBytes per second.

NOTE This provision is included to guarantee spare network capacity for other transmissions in between the blocks of a large binary file. When the binary file is transmitted as multicast, it will flood the network and can inhibit transmissions of other data.

7.3.8.5 End of transmission for non re-transmittable and re-transmittable binary file transfer

The receiver shall assume that a transmission has ended unsuccessfully when it gets a binary file block from the same source identified by the combination of SrcID, Device and Channel (see Table 9 and Table 10) with a new BlockID. Then the receiver stops the current receiving process and becomes ready for the new binary file block being received. The transmission shall also be considered finished when the last block is signaled by the SequenceNum from the sender. When a re-transmittable receiver identified by the DestID gets the last block after successful reception of all previous blocks, then it sends an ACK message to the sender to indicate successful transfer and so as to start new binary file block transmission. The receiver of non-re-transmittable binary file transfer and a receiver of re-transmittable binary file transfer not identified by DestID shall not send ACK message to the sender.

The re-transmittable sender assumes that the transmission is successfully finished only if it receives an ACK message with the SequenceNum which is equal to the MaxSequence; otherwise, a transmission has ended unsuccessfully. When a transmission is ended, a sender starts a new transmission if necessary.

7.3.8.6 Gaps between ACK messages for re-transmittable binary file transfer

In general, a receiver shall, immediately after loss detection, transmit an ACK message to the sender if a binary file block has been lost by having a gap in sequence numbers. Since there is a time delay between the reception of the ACK message and re-transmission of lost data at the sender, a receiver waits for the sender's response. For this purpose, a receiver should wait at least 200 ms before it sends another ACK message for the following datagrams. However, when a receiver receives all messages correctly, it shall send an ACK message immediately to the sender.

NOTE ACK message is used both for positive and negative acknowledge. See 7.3.3.5 for the description of the ACK message.

7.3.8.7 Maximum retransmissions for re-transmittable binary file transfer

The sender shall not retransmit the same datagram identified by sequence number of a binary file more than three times. After three retransmissions, the sender shall ignore any additional retransmission requests for this datagram identified by sequence number and continue transmitting the next datagram identified by sequence number. The receiver shall not query the same datagram identified by sequence number of a binary file more than three times.

The maximum number of re-transmission requests for a binary file shall be limited to 10 % of the maximum sequence number for the binary file but shall be not lower than three times. If the sender of a binary file receives more re-transmission queries than the maximum number, it shall ignore all further retransmission queries and continue to transmit the binary file until the last datagram identified by sequence number. In the case the receiver did not successfully receive all datagrams identified by sequence numbers of binary file, the receiver shall not

acknowledge the last received datagram identified by sequence number with sequence number equal to maximum sequence number. The receiver shall not query datagrams in the same binary file more than the maximum number of retransmissions.

In addition to data message re-transmission, control (Query) messages can be re-transmitted in case the control message is lost. The re-transmission counter increases whenever the control message is transmitted.

7.3.8.8 Timer management for re-transmittable binary file transfer

The re-transmission timer is managed at the sender. A sender sets the re-transmission timer when either a whole binary file block is transmitted and waits for an ACK message, or a control message (QUERY) message is transmitted. When the re-transmission timer expires, the sender (re-) transmits a QUERY message and sets the timer again unless the re-transmission counter reaches three.

7.3.8.9 UDP port and IP addresses for non re-transmittable and re-transmittable binary file transfer

Multicast addresses and ports for the service type are given in Table 5. As a default, addresses for non re-transmittable and re-transmittable binary file transfer service shall be 239.192.0.21 and 239.192.0.26 respectively. As a default, the port for non re-transmittable and re-transmittable binary file transfer shall be 60021 and 60026 respectively.

The receiver shall reply with ACK to the sender using the incoming datagram's AckDestPort and multicast address corresponding to this port number.

7.3.9 Error logging

Equipment shall maintain a count of the events of invalid binary file structures processed and make the count available. As a minimum, the following events shall be logged:

- the number of binary file blocks where errors occur;
- missing datagrams;
- unrecognized header.

7.4 General IEC 61162-3 PGN message transmissions

(see 8.12)

7.4.1 Message structure

The message structure for transporting IEC 61162-3 PGN messages into IEC 61162-450 networks is illustrated in Table 13.

Table 13 – Structure for PGN message

Header (see Table 9)
PGN message descriptor
IEC 61162-3 message fragment
IEC 61162-3 message fragment (zero or more)

The maximum message size of the PGN is 1 785 bytes. The PGN message shall be transmitted using one or two IEC 61162-450 datagrams. When there is a missing datagram, then the PGN message will be ignored as an error since the re-transmission of the lost datagram is not required.

7.4.2 Message format

The message format for transporting IEC 61162-3 PGN messages into IEC 61162-450 networks is illustrated in Table 14. The PGN message descriptor length for PGN messages is 32 bytes.

Table 14 – PGN message descriptor

Field Name	Size	Description
Source NAME (SNAME)	8 bytes of characters	Name of source. NAME shall be compliant with IEC 61162-3.
Source Device Identifier (SDID) [a]	2 byte of numeric number	Address of source device which is compliant with IEC 61162-3.
Destination NAME (DNAME)	8 bytes of characters	Name of destination. NAME shall be compliant with IEC 61162-3.
Destination Device Identifier (DDID) [a]	2 byte of numeric number	Address of destination device which is compliant with IEC 61162-3.
PGN number	4 byte of numeric number	PGN number of IEC 61162-3.
Priority	1 byte of numeric number	Priority of IEC 61162-3. Bit 0-2 are used and Bit 3 to Bit 7 are reserved.
Reserved (REVD)	7 bytes	Reserved bytes.
[a] Two bytes are specified to allow for future expansion.		

7.4.3 Address translation requirements

7.4.3.1 PGN group identification

A PGN group is defined as a logical group of devices that can share the information and message. Each PNGF shall be assigned a PGN group to communicate with devices in the group. The device address in a PGN group shall be unique in the network.

A PNGF may be registered with more than one PGN group if some of devices are required to communicate with devices in different PGN groups.

Means shall be provided to configure PGN groups at each PNGF.

7.4.3.2 Device identification

The PNGF shall represent all IEC 61162-3 equipment which are uniquely identified in the network.

A virtual device in an IEC 61162-3 network is identified by the source address. Each virtual device shall be identified by SFI of PNGF where it is connected, its PGN group number, its IEC 61162-3 source address and NAME. When there is no address available, then the address cannot be mapped until a new address is available. When a new address is not available, this event shall be recorded as specified in 4.3.3.

7.4.3.3 Address resolution

When a PNGF receives a query (i.e. Address Claim Message) about the device address with NAME and it has the information about the device, it shall respond with the address without forwarding the message to the IEC 61162-3 network.

7.4.4 Message processing

7.4.4.1 From IEC 61162-3 to IEC 61162-450

The PNGF shall have the capability of representing IEC 61162-3 devices as gateway device address and PNGF's SFI except for device address 0 which is always mapped to the device address 256. This is because the PNGF's device address of 0 represents PNGF itself on the IEC 61162-450.

The PNGF shall have the capability to represent it as at least an IEC 61162-3 device by obtaining its corresponding IEC 61162-3 source address from the IEC 61162-3 network.

When a PGN message is received from the IEC 61162-3, the PNGF extracts the information of source address, destination address, priority and PGN and it creates a message and fills up the corresponding fields. It also looks up the field of SFI and NAME of the destination address, and fills it out. When the received PGN message is not valid, then it will be discarded, and this event shall be recorded as specified in 4.3.3.

7.4.4.2 From IEC 61162-450 to IEC 61162-3

The PNGF shall have the capability to map 251 IEC 61162-3 source addresses to IEC 61162-450 device address and vice versa. The value of 251 is based on IEC 61162-3 address where the PNGF consumes 1 for the PNGF itself leaving 251 for mapping.

The address at IEC 61162-3 is source and destination device address. When a PNGF receives a message from an IEC 61162-450 network, it extracts the SDID and DDID information and puts it in the IEC 61162-3 PGN message and transmits into the IEC 61162-3 network. When the received PGN message is not valid, then it will be discarded, and this event shall be recorded as specified in 4.3.3.

7.4.4.3 Address conflicts

When there is a PNGF assigned address conflict in the address translation table of PNGF for mapping IEC 61162-3 network devices available as 450-Nodes in the 450-Network (i.e. when the same device address is assigned to more than one device), it shall be resolved. When a PNGF finds out that there are address conflicts, it re-assigns IEC 61162-3 device address mapping.

The address re-assignment process shall be done within 1 min.

7.4.5 Additional management requirements

7.4.5.1 Field configurable capability

The PNGF may also have the field configurable capability to change this default address so that the address is not claimed for a particular IEC 61162-3 device.

7.4.5.2 Non-volatile memory

The PNGF shall maintain configuration data in non-volatile memory. This ensures that field configurable settings are maintained across power cycles.

7.5 System function ID resolution

(see 8.13)

7.5.1 General

At the construction of a 450-Network of a ship, the assignment of SFI (system function ID) may be clearly defined. However, as the equipment of the ship is amended, replaced,

repaired and serviced, the assignment of SFIs may not be as clear. This protocol assists in the detection of SFI collisions.

NOTE The receiver functions are covered in IEC 61162-460.

7.5.2 Transmitter functions

These functions apply to all 450-Nodes.

A transmitter in a 450-Network shall, as a minimum after boot up, 1 min after boot up, 5 min after boot up and after reconfiguration which changes any fields in an SRP sentence, send on address 239.192.0.56 port 60056 an SRP sentence to assist detection of collision of the SFI (see Annex F). On receiving an SRP sentence with null fields, equipment shall respond with an SRP sentence with the fields populated.

Multiple sending of SRP is needed as different devices can have faster boot up time than the network monitoring performing the collision detection based on SRP sentences.

7.6 Binary file transfer using TCP point-to-point

(see 8.14)

7.6.1 Definition

This protocol provides a mechanism by which non IEC 61162-1 formatted data can be transmitted from a sender to a single receiver. The protocol emphasizes the reliability of the data transmission between two linked systems by using the TCP protocol.

NOTE The TCP standard is RFC 793. The IP standard is RFC 894. The Ethernet standard is IEEE Std 802.3.

Table 15 describes the terminology used.

Table 15 – Description of terms

Term	Description
BYTE	The lowest level data element consisting of 8 ordered bits (sometimes called an octet). Bit order is as determined by the computer implementation. Note that the implementation shall make any necessary conversion between network bit order and computer bit order.
Data packet	A number of bytes that contains a header, an optional sequence of reserved bytes and the actual message content. The header specifies the length of header itself, of reserved bytes and data and will also contain information that allows a number of data packets to be re-assembled into a presentation.
Data element	One or more bytes that forms a stand-alone information carrier, i.e. a time stamp, an integer or a character.
DWORD	Double word. One unsigned 32-bit integer (in range 0 to 4294967295). The DWORD is constructed from four consecutively transmitted BYTES, where the transmission order on the network is the most significant BYTE first followed by the next most significant BYTE until the least significant BYTE.
File	One group of bytes that forms a stand-alone data set.
Message data	The data contents of a data package.
Reserved bytes	A number of bytes in the data packet that may be ignored by the receiver. The reserved bytes may be additional header information that only has meaning for newer versions of the protocol or they may also be used for manufacturer specific purposes.
WORD	One unsigned 16-bit integer (in the range 0 to 65535). The WORD is constructed from two consecutively transmitted BYTES, where the transmission order on the network is the most significant BYTE followed by the least significant BYTE.
STRING[N]	A sequence of exactly n BYTES, interpreted as a string of characters. The transmission order on the network is the left-most character first. If the string is shorter than n, additional trailing bytes shall be set to zero. All strings in the header are encoded in ISO/IEC 18859-1 (ISO Latin 1).

7.6.2 Data field structure for transfer of files

7.6.2.1 General

The files are transmitted over the network in packets. The data field is defined as a sequential and unpadded stream of octets divided into two main groups – header and package data, as shown in Table 16. The header is needed for synchronisation and data integrity validation.

Table 16 – Binary file structure

Header (see 7.6.2.2)
Package data (see 7.6.2.3)

7.6.2.2 Elements of the header structure

The header format is defined in the Table 17. The first column specifies the name of the data item inside the header (starting from offset zero). The second column specifies the data type and size. The third column describes the data item and its purpose.

Table 17 – Header structure

Data item	Type	Description
token	STRING[6]	It shall always contain the string "RrTcP" including a trailing NULL character. Identifier as ASCII string with a length of 5 bytes. This token defines the beginning of a new data block.
crcHeader	WORD	Cyclic redundancy check for the header according to CRC-16-CCITT. The CRC is calculated from and including headerversion to and including any reserved bytes. The CRC is calculated from the sequence of bytes after formatting into transmission byte order. The CRC polynomial is: $x^{16} + x^{12} + x^5 + 1$.
headerversion	WORD	Defines the header version. The headerversion with value 1 is defined in this document. Extensions and/or modified versions will update this value.
headerlength	DWORD	Defines the binary file descriptor length in bytes. This is at least the length of the header including the reserved bytes. Future editions of this document may append additional fields to this file descriptor without incrementing the header version as long as these additional fields are compatible with the definition of the file descriptor in this document. Receivers which are not aware of these additional fields shall ignore them.
srcID	STRING[6]	Define the source system identifier in format "ccxxxx" (see 4.4.2).
datalength	DWORD	Defines the data content of this data package in octets. This may be the full (oversized) data in one package or a typical size for network transfer (1 280 octets). In the latter case, maxnum, actnum and streamlength will be used to synchronize data packets into a complete data transfer.
timeSec	DWORD	Seconds part of time stamp. Timestamp is constructed both of time in seconds and nanoseconds at the grabbing instant. If required by the application (e.g. image transmission to the VDR), The timestamp shall be made at the source immediately at data recording. If the application allows the timestamp to be optional and no timestamp is available, the value 0 shall be used for timeSec and timeNsec. The time representation is the number of seconds since January 1st 1970, not including leap seconds (i.e. in astronomic/GMT representation). This information is only needed with the first packet of each file or data stream. Time stamps in the following data packages belonging to the same data transfer shall be discarded by the receiver. It is only practicable to use this value if the synchronization between the destination device (e.g. VDR) and the source device (e.g. Radar unit) is sufficiently precise (in the range of milliseconds). The difftime data item

Data item	Type	Description
		may be used as an alternative method for synchronisation. If difftime is non-zero, this field shall be ignored.
timeNsec	DWORD	Nanosecond part of time stamp. See timeSec for details.
difftime		Time difference in milliseconds between data recording instant (e.g. grabbing instant) and transmission of the first packet of the file. A timestamp with a resolution of at least in the millisecond range is made immediately before the source generation (e.g. screenshot) and the second timestamp is made immediately before the first packet is transmitted. The difference is entered as "difftime" and the packet is then sent. The destination device (e.g. VDR) uses this difftime value together with its system time to determine the timestamp for the transmitted data. Time tolerances between destination device and source device may be neglected, because the time reference of the destination device is always the system time of the destination device.
maxnum	DWORD	Number of packets needed for transmission of the corresponding file or data stream. The value can be 1 or more
actnum	DWORD	This packet number (range from 1 to maxnum)
streamlength	DWORD	Defines the length of the (full) stream/presentation content in octets
device	BYTE	Data source (device) as binary value, 1 ≥ equipment 1, 2 ≥ equipment 2, etc. The value can be between 1 and 255.
channel	BYTE	Subdivision according to data source (device), values from 1 to 255, default = 1.
deviceip	DWORD	IP of transmitting device; optionally used. The IP address is entered in Network Byte Order Format (DWORD).
deviceport	WORD	That port the transmitting device has used. It may be used optionally.
typelength	BYTE	The length of the datatype field.
datatype	STRING[16]	This string defines the datablock encoding by assigning a MIME content type to the datablock for the server followed by a null character. For example, image/png is used for PNG image files and application/zip is used for zip-files.
Status of acquisition	WORD	The status for the data return. A zero is returned for normal operation. Non-zero value is used to indicate an error condition. A descriptive text may be put in the status and information text field.
StatusLength	WORD	The length of the "Status and information text" field in bytes.
Status and information text	STRING[n]	Status information (e.g. successful operation or error codes). This may be one or more strings terminated by a binary null.

7.6.2.3 Elements of the package data structure

The package data format is defined in Table 18. The first column specifies the name of the data item. The second column specifies the data type and size. The third column describes the data item and its purpose.

The package data structure size is set to zero if only status information is transmitted.

Table 18 – Package data structure

Data item	Type	Description
datablock	BYTE[datalength]	This item is the data either split into pieces or in one block. Size is defined by datalength in the header.

There is no CRC for the data contents as this is partly handled by the TCP/IP layer or by other mechanisms in the contents format. The header has a separate CRC as it is deemed more critical for the correct operation of the system.

7.6.3 Structure of the transfer stream

7.6.3.1 General

The complete binary file is split into a number of datablocks. Each header and datablock is transmitted in increasing order, beginning with the first datablock and ending with the last datablock. Synchronisation is achieved with data items actnum and maxnum.

7.6.3.2 Unknown data types

A receiver that does not understand an incoming data type shall ignore all incoming data without closing the connection if the receiver is a server.

If the receiver is a client and does not understand incoming data, it shall immediately close the connection.

7.6.3.3 Maximum outgoing rate

The data volume for each transmit client of binary file shall not exceed 2 MBytes per second.

NOTE This provision is included to guarantee spare network capacity for other transmissions in between the blocks of a large binary file. When the binary file is transmitted as multicast, it will flood the network and can inhibit transmissions of other data.

7.6.4 TCP port and IP addresses

The IP address shall be freely selectable outside the addresses assigned for other purposes in this document and the IP address is depending on the network configuration of the corresponding equipment manufacturer.

The IP address of each file source and the file receiver has to be coordinated and set manually beforehand to be in the same IP address range.

Equipment unable to perform an address look-up service should be configured to the same IP sub-net. A router may be used if the equipment is connected on different IP sub-nets.

The default TCP port between sender and receiver for the transfer shall be 7097. Sender and receiver shall support configuration of the port number and IP address.

7.6.5 Implementation guidance

7.6.5.1 Receiver as server and sender as client

This setup is used for example for VDRs where the VDR as the file receiver has to be configured as a passive listening device. The file sender is the active transmit client connecting and transferring the data.

Depending on the application, the file receiver may be set up to accept multiple transmit clients on the same input port. This is necessary if more than one transmit client is assumed to send its files to the receiver server.

7.6.5.2 Connection management from sender client

The transmit client shall establish a connection to the receiver server immediately after system initialisation. Once the connection is established, the transmit client is responsible for the connection and streaming of data packet to the receiver server.

If the connection attempt fails or connection is lost, the transmit client shall try to establish the connection again. The interval between attempts shall not exceed 30 s.

7.6.5.3 Connection management from receiver server

The receiver server shall make the listening port available for data transfers during initialisation.

The manufacturer shall specify the maximum number of transmit client connections for the receiver server. The receiver server shall receive data individually from connected transmit clients and detect any loss of connection from transmit clients.

The equipment test and performance standard may require alerts to be raised for loss of connection.

The receiver server may in some cases only detect a failed connection by timeout since data was received last time. The receiver server shall reinitialise the listening port and the receiver software module after timeout for the transmit client.

7.6.5.4 Error handling

The receiver shall ensure data integrity at reception by verification of the header including token, version, consistency of data fields and the header CRC. Erroneous data reception shall be processed and indicated according to individual equipment standard.

NOTE Consistency of data fields can depend on application. However, strings can be checked against containing illegal characters, message sequence numbers can be checked, etc.

7.6.5.5 Transmission of a file

The client transmission of a file may occur at any time when the connection is open. The message header information is sufficient for the server to decode the data stream and reassemble the file and its associated header information data.

7.6.5.6 Device identification

All clients shall be configured with a unique source SFI and device identification (1 to 255) to allow the server to unambiguously identify the source of the received packets.

8 Methods of test and required results

8.1 Test set-up and equipment

The following test methods require test equipment capable of transmitting and receiving UDP datagrams over the Ethernet interface and the use of a network protocol analyser. The test equipment shall be capable of supporting the Ethernet interface appropriate for the EUT. The equipment shall also be capable of generating invalid data.

The test equipment shall be configured to transmit UDP broadcast messages for the ports defined in 6.2.2.

Simulation equipment is required to be capable of:

- generation of test UDP datagrams containing unique and numbered content, syntactically correct and incorrect sentences with datagram intensity that can be varied to exceed IEC 61162-1 and IEC 61162-2 channel capacity;
- if the EUT implements support for PGN, generation of IEC 61162-3 PGN test sentences containing unique and numbered content, syntactically correct and incorrect with variable length and correct, incorrect and missing checksum;

- generation of IEC 61162-1 test sentences containing unique and numbered content, syntactically correct and incorrect with variable length and correct, incorrect and missing checksum;
- generation and reception of non re-transmittable and re-transmittable binary files.

8.2 Basic requirements

8.2.1 Equipment to be connected to the network

(see 4.2.1)

Verify through inspection of test documentation that the EUT has been tested against the relevant requirements contained in IEC 60945.

For the purposes of IEC 60945 the following definitions apply.

- **Performance check**
  A performance check is the successful transmission and reception of data.
- **Performance test**
  A performance test consists of evaluating performance under different test scenarios.

8.2.2 Network infrastructure equipment

(see 4.2.2)

Confirm by inspection of manufacturer provided information that the EUT does not provide the functions of a repeater hub.

Confirm by inspection of documented evidence that the EUT supports IGMP protocol and that the version of IGMP support is documented.

If the EUT is a switch,

- confirm by inspection of documented evidence that it supports IGMP snooping, and
- confirm by inspection of documented evidence that the IGMP snooping based multicast traffic filtering is supported per each multicast address.

Use a simulation arrangement to generate multicast datagrams with address range of 224.0.0.1 to 224.0.0.255 and confirm by observation that the EUT does not filter out those datagrams.

8.2.3 Documentation

(see 4.4.1.7.1.1)

Confirm by inspection of manufacture's documentation that all of the implemented datagram types are specified.

8.3 Network function (NF)

8.3.1 Maximum data rate

(see 4.3.2)

Confirm by inspection that the manufacturer has specified the maximum datagram input rates as specified in items a) to c) in 4.3.2.

After activating all NF ports of the equipment under test with the specified maximum aggregate datagram rate as specified in 4.3.2, check that the performance of the equipment is not degraded in any way.

8.3.2 Error logging function

(see 4.3.3)

Confirm that the manufacturer has provided means to inspect a log of detected errors.

NOTE Tests for the errors to be logged are given in 8.5.2, 8.9.2, 8.10 and 8.11.4.

Confirm that, if external data logging capability is provided, the output of syslog messages conforms to the manufacturer's documentation and the requirements of 4.3.3.2.

If reception of syslog message capability is provided, confirm by analytic evaluation that the reception and logging of syslog messages conforms to the manufacturer's documentation and the requirements of 4.3.3.2.

8.4 System function block (SF)

8.4.1 General

(see 4.4.1)

For SFs that implement IEC 61162-1 interfaces, verify compliance in accordance with the test methods and required test results of IEC 61162-1.

For SFs that implement IEC 61162-2 interfaces, verify compliance in accordance with the test methods and required test results of IEC 61162-2.

8.4.2 Assignment of unique system function ID (SFI)

(see 4.4.2)

Check that means are provided to assign and configure the SFI, as described in 4.4.2.

Check that manufacturer's documentation include instructions how to select "cc" and "xxxx" part of the SFI so that the SFI is unique at least within the IEC 61162-450 network.

8.4.3 Implementing configurable transmission groups

(see 4.4.3)

Check that means are provided to assign and configure the transmission groups. Check that documentation has been provided describing the transmission groups supported by the device.

8.5 Serial-to-network gateway function (SNGF)

8.5.1 General

(see 4.5.1)

Check that it is possible to enter unique SFIs for all sources distinguished by different talker mnemonic per each serial port of the device and that the mapping of SFI to sources distinguished by different talker mnemonic per each serial port is correctly implemented by analyzing the UDP datagrams.

Check that TAG block source identification (s:) is correctly implemented to sources distinguished by different talker mnemonic per each serial port by analysing the UDP datagrams.

Check that TAG block destination identification (d:) is correctly implemented for routing from 450-Network to serial ports.

Check that documentation is available describing any filtering used in the device.

8.5.2 Serial line output buffer management

(see 4.5.2)

Verify the output routing by feeding the network under test with datagrams containing sentences for all available serial outputs and check that sentences are routed to the output ports having the set SFIs.

Verify output buffer overflow handling by increasing the datagram data rate until possible capacity of the serial lines are exceeded and check that

- prioritized sentences are correctly replaced, maintaining the FIFO order and not affecting sentence integrity, and
- in case buffer overflow sentences are discarded, the FIFO order is maintained, not affecting sentence integrity, and the buffer overflow events are logged as required.

Verify required functionality for prioritized messages by repeating the test with the unit set for prioritized messages and check that behaviour is correct.

Verify message buffer integrity by repeating the test also with grouped messages and check that overflow handling maintains group integrity, meaning that whole groups are discarded, regardless of the prioritized message setting.

8.5.3 Datagram output

(see 4.5.3)

Verify datagram conversion by feeding the input ports of the network under test with sentences and check that these are transmitted in UDP datagrams with correct syntax, SFI, source identification (s:) and, if required, destination identification (d:).

The test sentences should include TAG blocks and grouped messages.

Test configuration should include single source per serial port and multiple sources distinguished by different talker mnemonics per shared serial port.

8.5.4 Datagram output multi SF serial port

(see 4.5.4)

Verify datagram conversion by feeding the input ports of the network under test with sentences and check that these are transmitted in UDP datagrams with correct syntax, SFI, source identification (s:) and, if required, destination identification (d:).

Test configuration should be configured for multiple sources distinguished by different talker identifiers and manufacturer mnemonic codes (for proprietary sentences) per shared serial input port. The output should be configured for single destination by talker identifier.

Check the test cases below:

1) received sentences with configured talker identifiers and manufacturer mnemonic codes will transmit datagrams with the configured SFI;
2) received sentences without a configured talker identifier or manufacturer mnemonic code will transmit datagrams for each configured SFI;
3) received datagrams with configured destination SFIs will be transmitted on configured serial port;

4) received datagrams with a valid destination that is an unknown SFI will not be transmitted on any serial port;

5) received datagrams with no destination specified will be transmitted to all serial ports.

In the test cases below, the SNGF serial port default SFI is "SI0001", the configured SFIs are TI0001 (for Talker Identifier "TI") and VD0001 (for Talker Identifier "VD"). A proprietary sentence “PMANMSG” is configured for SFI VD0001. The typical received sentences will then include rate-of-turn ("$TIROT") and speed ("$VDVBW").

- Test case 1: An example of simple SFI conversion:

  "$TIROT,123.45*67<CR><LF>$VDVBW,10.00,,A,,,V,,V,,V*hh<CR><LF>”

  “$PMANMSG,proprietary_contents*hh<CR><LF>"

  will generate three datagrams, one for each SFI:

  "\s:TI0001,n:333*hh\$TIROT,123.45*67<CR><LF>"

  "\s:VD0001,n:111*hh\$VDVBW,10.00,,A,,,V,,V,,V*hh<CR><LF>"

  “\s:VD0001,n:111*hh\>$PMANMSG,proprietary_contents*hh<CR><LF>”

  with the IEC 61162-450 Header("UdPbC'0'").

- Test case 2: An example of un-configured talker identifier:

  "$SDDPT,123.4,,400*hh<CR><LF>"

  will generate one datagram for each configured SFI:

  "\s:TI0001,n:222*hh\$SDDPT,123.4,,400*hh<CR><LF>"

  "\s:VD0001,n:222*hh\$SDDPT,123.4,,400*hh<CR><LF>"

  with the IEC 61162-450 Header("UdPbC'0'").

- Test case 3: An example of simple SFI conversion, no TAG block support:

  Datagram "\s:IN0001,d:TI0001,n:333*hh\$INTIQ,ROT*hh<CR><LF>"

  will generate transmission of the sentence on the serial port configured for the destination SFI TI0001:

  "$INTIQ,ROT*hh<CR><LF>"

- Test case 4: An example of a specified destination but un-configured SFI conversion:

  Datagram "\s:IN0001,d:GN0001,n:333*hh\$INGNQ,ZDA*hh<CR><LF>"

  will not generate transmission on any serial ports.

- Test case 5: An example of no specified destination:

  Datagram “\s:IN0001,n:333*hh\$INGNQ,ZDA*hh<CR><LF>”

  will generate transmission of the sentence on all serial ports.

  “$INGNQ,ZDA*hh<CR><LF>”

8.5.5 Handling malformed data received on serial line

(see 4.5.5)

Verify datagram conversion by feeding the SNGF input ports under test with valid sentences interleaved with malformed data according to 4.5.5.

Confirm that the valid sentences are correctly converted into datagrams.

Each test shall include test cases for all of the start characters. Check that the test cases below will generate a datagram transmission:

1) when data has been received before a start character;

2) when data has been received after a valid start character and the maximum sentence and TAG block length has been exceeded;

3) when data has been received after a valid start character and end of line (<CR><LF>) has not been received within 1 s;
4) when a reserved character has been received and not having been appropriately escaped;
5) when random binary data is sent on serial line.

In the test cases below, the SNGF serial port is configured as SFI TI0001 and default SI0001 is the SFI of the SNGF.

- Test case 1: An example of data before start character:

  `Serial data "127,333*6B<CR><LF>$TIROT,123.45*67<CR><LF>"`

  will generate two datagrams:

  `"\s:SI0001,n:444*hh\127,333*6B<CR><LF>"`

  `"\s:TI0001,n:445*hh\$TIROT,123.45*hh<CR><LF>"`

  and with the IEC 61162-450 Header("UdPbC'0'").

- Test case 2: An example of too long line:

  `Serial data "$TIALR,123456,906,A,V,Sensor fault with a too long description to violate serial data maximum line length limitation*hh<CR><LF>"`

  will generate one datagram, i.e. no change to content:

  `"\s:TI0001,n:446*hh\$TIALR,123456,906,A,V,Sensor fault with a too long description to violate serial data maximum line length limitation*hh<CR><LF>"`

  and with the IEC 61162-450 Header("UdPbC'0'").

- Test case 3: An example of timeout:

  `Serial data "$TIALR,123456,906,A,V,"`

  <1.1 s delay>

  `Serial data "Sensor fault*hh<CR><LF>"`

  will generate two datagrams:

  `"\s:TI0001,n:447*nn\$TIALR,123456,906,A,V," and`

  `"\s:SI0001,n:448*nn\Sensor fault*hh<CR><LF>"`

  and with the IEC 61162-450 Header("UdPbC'0'").

- Test case 4: An example of incorrect escape:

  `"$TITXT,01,01,01,Incorrect * escape*hh<CR><LF>"`

  will generate a datagram (i.e. no change to content):

  `"\s:TI0001,n:449*nn\$TITXT,01,01,01,Incorrect * escape*hh<CR><LF>"`

  and with the IEC 61162-450 Header("UdPbC'0'").

- Test case 5: An example of random serial data including start characters "$" that will initiate a new buffer:

  This will generate a datagram:

  `"\s:SI0001,n:449*nn\kfajds...3efbnajfu93hn" followed by`

  `"\s:SI0001,n:450*nn\$lkfdajkf98873tq87784((/kfajd.."`

  and with the IEC 61162-450 Header("UdPbC'0'").

8.6 Other network function (ONF)

(see 4.7)

Verify by inspection of the manufacturer's documentation that information for the use of ONF is provided as described in 4.7.

Verify using the test equipment described in 8.1 that the ONF does not use any of the multicast IP addresses reserved in 5.4.

NOTE The test equipment to confirm the source and destination of general ONF traffic could be a network analyser.

8.7 Low level network

8.7.1 Electrical and mechanical requirements

(see 5.1)

Verify by observation that one of the connectors specified in Table 3 is available on the equipment.

Verify by inspection of manufacturer documentation that one or more of these interfaces meets the requirements of Table 3.

Verify by inspection of manufacturer documentation that the laser safety requirements for Class 1 devices are met.

8.7.2 Network protocol

(see 5.2)

Confirm by inspection of documented evidence that the relevant IEEE 802.3 data link protocol is used.

Verify using the network protocol analyser that IP (Version 4) protocol is used and that no IP option is used.

Confirm using ping program that each device supports the network protocols specified.

8.7.3 IP address assignment for equipment

(see 5.3)

Confirm by observation that means are provided to configure an IP address for the device.

Confirm that an IP address for the device is configured within the ranges reserved for private networks as described in ISOC RFC 1918.

Confirm that any excluded IP ranges reserved for internal sub-nets (internal to the equipment) are documented and those are not in the range given by 5.3.

Using the test equipment described in 8.1 and documentation provided by the manufacturer, verify by transmitting and receiving data that the equipment does not change its IP address and IP port settings after an OFF/ON power cycle.

8.7.4 Multicast address range

(see 5.4)

Verify, using the network protocol analyser, that each datagram is transmitted and received with the multicast address 239.192.0.1 to 239.192.0.64.

8.8 Transport layer

(see Clause 6)

Verify that UDP messages are transmitted and received at each of the appropriate port numbers as defined in Tables 4 and 5.

Verify that UDP are discarded if the received UDP checksum is invalid.

Verify that each datagram contains no more than 1 472 bytes.

8.9 Application layer

8.9.1 Application

(see 7.2.1)

Using the test equipment described in 8.1 and documentation provided by the manufacturer, verify by transmitting and receiving data that each SF and SNGF port of the equipment under test can send and receive IEC 61162-1 sentences and allows several sentences to be merged into one datagram if applicable.

8.9.2 Datagram header

(see 7.1)

Check that all UDP multicast datagrams are headed by

- "UdPbC" for transmission of IEC 61162-1 formatted sentences,
- "RaUdP" for transmission of binary files,
- "RrUdP" for transmission of re-transmittable binary files, and
- "NkPgN" for transmission of IEC 61162-3 PGN messages,

followed by a null character (all bits set to zero) as the first six bytes of the datagram.

Check that all TCP/IP datagrams are headed by "RrTcP" for transmission of binary files as described in 7.6 followed by a null character (all bits set to zero) as the first six bytes of the datagram.

Check that incoming datagrams with an unknown header are discarded without processing the content beyond the header.

Verify that, as part of error logging, the count of received datagrams without valid datagram header (see 7.1) is increased if datagram header is unrecognized or invalid.

8.9.3 Types of messages

(see 7.2.2)

Using the test equipment described in 8.1, and documentation provided by the manufacturer, verify by transmitting and receiving data that each SF and SNGF port of the equipment under test can send and receive each of the message types specified by the manufacturer; one or more of SBM, MSM and CRP. For CRP messages, verify that the requirements of Clause C.4 are met by inspection of recorded datagrams and, in the case of timeout handling, the equipment's error log data.

8.9.4 TAG block parameters

(see 7.2.3)

8.9.4.1 Test of the transmitter

Verify using a receiving protocol analyzer that

- all members of group have same group code value,
- next group code value after 99 is 1,
- the EUT transmits the source identifier (two separate test cases – default and configured),
- if used, the EUT transmits valid destination code,

- line count value increments for each line and resets after 999 to 1,
- if provided, the heartbeat sentence (HBT) is transmitted at least once every 60 s, and
- the EUT only feeds sentences preceded by a valid TAG block (for example "\s:II0001,n:23*31\$LCGLL,5420.123,N,01030.987,E,,A,A*58<CR><LF>") into the network.

8.9.4.2 Test of the receiver

Verify, using a transmitting protocol analyser, that

- lines without a TAG block are not used as defined in 7.2.3.1,
- adding a TAG block containing syntactically correct parameter codes (for example "\z:Y23G81*56\") not defined in this document is transparent to normal operation,
- only complete sentence groups are used, and
- TAG block lines with the EUT as destination are processed.

NOTE Processing can also mean that data is dropped.

8.9.4.3 Test for bidirectional communication

If the network under test supports CRP, then, using a bidirectional protocol analyzer, verify that source and destination are correct in the CRP communication.

8.9.4.4 Configuration

Verify by inspection of documentation that it is not possible to dynamically configure any identities after installation.

8.9.5 General authentication

(see 7.2.3.8)

These tests apply to a EUT that includes transmission of authentication.

Confirm by inspection of manufacturer's documentation which signature methods the EUT provides.

Confirm by analytic evaluation that the EUT transmits sentence or message with correct authentication code as described in 7.2.3.8. Repeat the test for all signature methods supported by the EUT.

Use simulation arrangement to create valid examples of authenticated sentences or messages and confirm by observation that, if the EUT is not set to require authentication, the EUT processes all sentences or messages.

Use simulation arrangement to create same valid examples of authenticated sentences or messages as in previous test, and confirm by observation that, if the EUT is set to require authentication, the EUT processes all sentences or messages. Repeat the test for all signature methods supported by the EUT.

Use simulation arrangement to create same valid examples of sentences or messages as in previous test, but without including authentication parameter code, and confirm by observation that, if the EUT is set to require authentication, the EUT discards all sentences or messages.

Use simulation arrangement to create same valid examples of sentences or messages as in previous test, but with intentionally incorrect value in the authentication parameter code, and confirm by observation that, if the EUT is set to require authentication, the EUT discards all sentences or messages. Repeat the test for all signature methods supported by EUT.

8.10 Error logging

(see 7.2.5)

By feeding test sentences with variable contents into the network, verify that the network under test processes only sentences preceded by a valid TAG block as defined in 7.2.3.1 and verify that

- lines with TAG checksum errors increase the corresponding error log count as defined in 4.3.3,
- lines with TAG syntax errors increase the corresponding error log count as defined in 4.3.3, and
- lines with TAG framing errors (i.e. missing "\" character at start, stop and between adjacent TAG blocks) increase the corresponding error log count as defined in 4.3.3.

Check handling of incorrect messages by feeding the network under test with sentences having

- incorrect syntax,
- incorrect checksum, and
- incorrect message length.

Verify that these sentences are discarded and that the network's error logs are updated.

8.11 Binary file transfer using UDP multicast – Single transmitter, multiple receiver

(see 7.3.)

8.11.1 Sender process test

8.11.1.1 Non re-transmittable binary file transfer

Using a test set-up with non re-transmittable binary files, verify that

- header token is set correctly,
- header version is set according the Table 9,
- SrcID is set according to Table 9,
- DestID is correctly set according to Table 9,
- unique BlockID is correctly set,
- BlockID, SequenceNum and MaxSequence are correctly set;,
- Device is correctly set,
- Channel is correctly set,
- the IP address and port numbers are assigned by one of the addresses for non-re-transmittable binary file transfer,
- the SequenceNum of first datagram is set to 1, and
- there is no response when a receiver sends any ACK messages.

8.11.1.2 Re-transmittable binary file transfer

Using a test set-up with re-transmittable binary files, verify that

- header token is set correctly,
- header version is according to Table 9,
- SrcID and DestID are correctly set by "ccxxxx",
- Unique BlockID is correctly set,

- BlockID, SequenceNum and MaxSequence are correctly set,
- Device is correctly set,
- Channel is correctly set,
- the IP address, port number and AckDestPort are assigned by one of the addresses for binary file transfer,
- the maximal re-transmission count is calculated correctly according 7.3.8.7,
- the SequenceNum of the first datagram is set to 1,
- ACK messages are received from multicast group, specified from AckDestPort,
- ACK messages are only processed if the DestID of ACK message is equal to own SFI and if the SourceID of ACK message is equal to actual DestID, otherwise the ACK message is ignored,
- the binary transfer is finished and marked as successful after an ACK message is received, whose SequenceNum is equal to the MaxSequence, after all data is transmitted,
- a QUERY message is sent when there is no ACK message received, after all data are transmitted,
- not more than three QUERY messages at all are sent when there is no ACK message received after a QUERY message is transmitted,
- binary file data from SequenceNum one higher as SequenceNum of ACK is re-transmitted when an ACK message whose SequenceNum is less than the MaxSequence is received,
- the same SequenceNum is retransmitted not more than three times, otherwise the re-transmittable sender continues with normal transfer and ignores ACK message,
- the number of all re-transmissions is not more than the maximal re-transmission count, otherwise the re-transmittable sender continues with normal transfer and ignores ACK messages,
- the binary transfer is finished and marked as not successful after all data is transmitted, if three QUERY messages are sent and no ACK message whose SequenceNum is equal to the MaxSequence is received, and
- log messages are correct.

8.11.2 Receiver process test

8.11.2.1 Non re-transmittable binary file transfer

Using a test set-up with non re-transmittable binary files, verify that

- messages are received correctly on given IP and port address,
- message is only processed if the header token is equal to "RaUdP" or "RrUdP",
- message is only processed with valid header and valid binary image descriptor,
- each separate binary file transfer is identified by the combination of SrcID, BlockID, Device and Channel,
- a new receiving process starts when a message with new BlockID is received for the combination of SrcID, Device and Channel,
- the received messages are the same as that of the transmitted data when there is no loss,
- any log information is provided if there is any loss, and
- log messages are correct.

8.11.2.2 Re-transmittable binary file transfer

Using a test set-up with re-transmittable binary files, verify that

- messages are received correctly on given IP and port address,

- message is only processed if the header token is equal to "RrUdP",
- message is only processed with valid header and valid binary image descriptor,
- message is only processed if DestID is equal to own SFI,
- each separate binary file transfer is identified by the combination of SrcID, BlockID, Device and Channel,
- the maximal re-transmission count is calculated correctly according 7.3.8.7,
- ACK messages are generated with the correct token = RrUdP, SrcID = own SFI, DestID = SrcID of received message, BlockID and without binary image descriptor and without data block,
- ACK messages are transmitted to the multicast group corresponding to the AckDestPort of the actual received binary file block,
- the receive process is marked as successful and an ACK message is transmitted when the received SequenceNum is equal to the MaxSequence with the same instance identifier and if all sequences of the actual binary file block are received,
- an ACK message for a successful received binary file block is not more than three times repeated if query messages are received,
- if no complete binary file block is received, the transfer is marked as not successful and no ACK message after the last received sequence with SequenceNum equal to MaxSequence is transmitted, either directly after received last sequence or after received query messages,
- an ACK message is transmitted with the last received SequenceNum before a gap when the re-transmittable receiver detects that there is a gap in the SequenceNum between two consecutive messages,
- an ACK message for the same requested SequenceNum is not more than three times repeated for the same binary file block,
- the number of all ACK messages for retransmission request is not more than the maximal re-transmission count for the same binary file block,
- a new receiving process starts when a message with new BlockID is received for the combination of SrcID, Device and Channel,
- the received messages are the same as that of the transmitted data,
- the re-transmittable receiver does not send any ACK message when a re-transmittable sender sends a binary file block with different DestID, and
- log messages are correct.

8.11.3 Binary file descriptor test

Using a test set-up with binary files, verify that

- the AckDestPort field is correctly set,
- the Length field, the TypeLength field and the StatusLength field are correctly set,
- binary file length in the descriptor is the same as the size of the received data, and
- the received data format is the same as that of the data type in the descriptor.

8.11.4 Binary file transfer error logging

Using a test set-up with binary files, verify that the following events can be logged:

- number of binary file blocks where errors occur;
- missing datagrams;
- unrecognized headers.

8.11.5 Maximum outgoing rate

Confirm by inspection of documented evidence that the EUT has an effective method to limit the outgoing rate to be within the given limit.

8.12 PGN to network gateway function (PNGF)

(see 7.4.)

8.12.1 General

Check that it is possible to enter unique SFIs for all sources distinguished by different devices and that the mapping of SFI to sources distinguished by different device identifier is correctly implemented by analysing the UDP datagrams.

Check that documentation is available describing any filtering used in the device.

8.12.2 Output buffer management

Verify the output routing by feeding the network under test with datagrams containing PGNs for all IEC 61162-3 networks and check that PGNs are routed to the network having the set device identifier.

Check that documentation is available describing the maximum buffer capacity.

Check that the means are provided to configure the maximum buffer capacity.

Check that the overflow is logged as required.

8.12.3 Datagram output

Verify datagram conversion by feeding the input ports of the network under test with PGNs and check if these are transmitted in UDP datagrams as described in 7.4.1.

Verify a single IEC 61162-3 PGN message transmission per each feeding IEC 61162-450 message.

8.12.4 PGN group

Verify PGN group filtering by transmitting four PGN groups and check that the device only receives the PGN group messages to which it belongs.

8.12.5 Address conflicts

Confirm by observation that the EUT assigns new IEC 61162-3 addresses at the address translation table within 1 min when IEC 61162-3 addresses at the EUT conflict with the addresses in IEC 61162-3 Network.

8.13 System function ID resolution

(see 7.5)

Confirm by observation that the EUT sends SRP sentences to address 239.192.0.56 port 60056 when boot up, 1 min after boot up, 5 min after boot up and after reconfiguration, including both reconfiguration of setup parameters and reconfiguration based on a change caused by redundancy arrangements.

8.14 Binary file transfer using TCP point-to-point

(see 7.6)

8.14.1 Test of transmit client

8.14.1.1 Description

The test set-up is a controllable receiver server and the equipment under test. The following tests shall be performed and passed.

8.14.1.2 Connection establishment test

Remove receiver server from the network and power up the transmit client. Confirm by observation that the transmit client performs reconnection attempts as specified.

Connect receiver server and confirm by observation that connection is established.

8.14.1.3 Lost connection test

Power down or physically remove receiver server from the network and confirm by observation that the transmit client detects connection failure. This will normally require the transmission of some data from the transmit client.

Reconnect receiver server and confirm by observation that the transmit client reconnects the receiver server. Confirm by observation that the transmit client sends data as specified. Confirm by observation that the headers are according to the data format specification. Confirm by observation that time stamp is increased.

Break connection in the middle of a transfer. Confirm by observation that the transmit client continues to operate and tries to reconnect.

8.14.2 Test of receiver server

8.14.2.1 Test set-up

The test set-up is a controllable transmit client and the equipment under test. The transmit client shall be able to generate the following, and tests shall be made that check the correct functioning of the receiver server in these cases.

8.14.2.2 Connection establishment test

Remove transmit client(s) from the network and power up receiver server. Confirm by observation that receiver server starts up as specified.

Connect transmit client(s) and confirm by observation that receiver server enters normal operation.

8.14.2.3 Lost connection test

Break connection in the middle of a transfer. Confirm by observation that the receiver server continues to operate.

8.14.2.4 Message transfer test

Transfer at least one file with a content type which is supported by the receiver server, streamed as a sequence of at least 5 datablocks. Confirm by observation that the receiver server correctly processes the file.

8.14.2.5 Multiple transmit client test

If the receiver server supports simultaneous connections from multiple transmit clients, establish the maximum number of connections according to the manufacturer. Send files

simultaneously over all connections. Confirm by observation that the receiver server correctly processes all received files.

8.14.2.6 Erroneous input test

Send a file with datalength in the header set to a value which is smaller than the actual size of transmitted data. Confirm by observation that the receiver server detects the error and indicates it according to the individual equipment standard.

Send a file with an invalid crcHeader value in the header. Confirm by observation that the receiver server detects the error and indicates it according to the individual equipment standard.

Send a file which is streamed as at least 5 packets. Discard the 3rd packet. Confirm by observation that the receiver server detects the error and indicates it according to the individual equipment standard.

8.14.2.7 Undefined header test

Send a file with the header version set to a value higher than defined in this document. Confirm by observation that the receiver server ignores the unknown part of the header based on the implemented header version and that the receiver server processes the file.

8.14.3 Maximum outgoing rate

Confirm by inspection of documented evidence that the EUT has an effective method to limit the outgoing rate to be within the given limit.

8.14.4 TCP port and IP addresses

Confirm by inspection of manufacturer's installation documentation that the default port is specified as 7097 and that there are instructions to set both sender and receiver in the same IP address range.

Annex A
(normative)

Classification of IEC 61162-1 talker identifier mnemonics and sentences

A.1 General

Table A.1 gives a mapping from talker identifier mnemonic to a default transmission group for an SF.

Table A.2 gives default classification of each of the IEC 61162-1 sentence formatters as belonging to one of the following three types of message:

- sensor broadcast message (SBM), see 3.22;
- multi-sentence message (MSM), see 3.14;
- command-response pair (CRP), see 3.4.

If provided by the equipment, the default transmission group and classification can be changed by the parameter setup system of the equipment to USR1 to USR8, RCOM, PROP in Table 4 or any in Table 5.

A.2 Talker identifier mnemonic to transmission group mapping

Table A.1 maps the two first characters of the SFI, which is normally the IEC 61162-1 talker identifier mnemonic, to the default transmission group the SF shall use for transmitting sentences. For the two character codes listed in Table A.1, the transmission group is identified in column three. For two character codes not in this table, the SF shall use the MISC transmission group as default. For alert communication purposes, an alert source may use transmission group BAM1, BAM2 or transmission group based on Table A.1.

Proprietary sentences that do not use a talker identifier mnemonic can be given a default transmission group by the manufacturer.

Table A.1 – Classification of IEC 61162-1 talker identifier mnemonics

Type of equipment	Talker identifier	Transmission group
Heading/track controller (autopilot) general	AG	NAVD
magnetic	AP	NAVD
Automatic identification system	AI	TGTD
Bilge system	BI	MISC
Bridge navigational watch alarm system	BN	VDRD
CAM of BAM	CA	CAM1 or CAM2
Communications: digital selective calling (DSC)	CD	RCOM
data receiver	CR	RCOM
satellite	CS	RCOM
radio-telephone (MF/HF)	CT	RCOM
radio-telephone (VHF)	CV	RCOM
scanning receiver	CX	RCOM
Direction finder	DF	NAVD