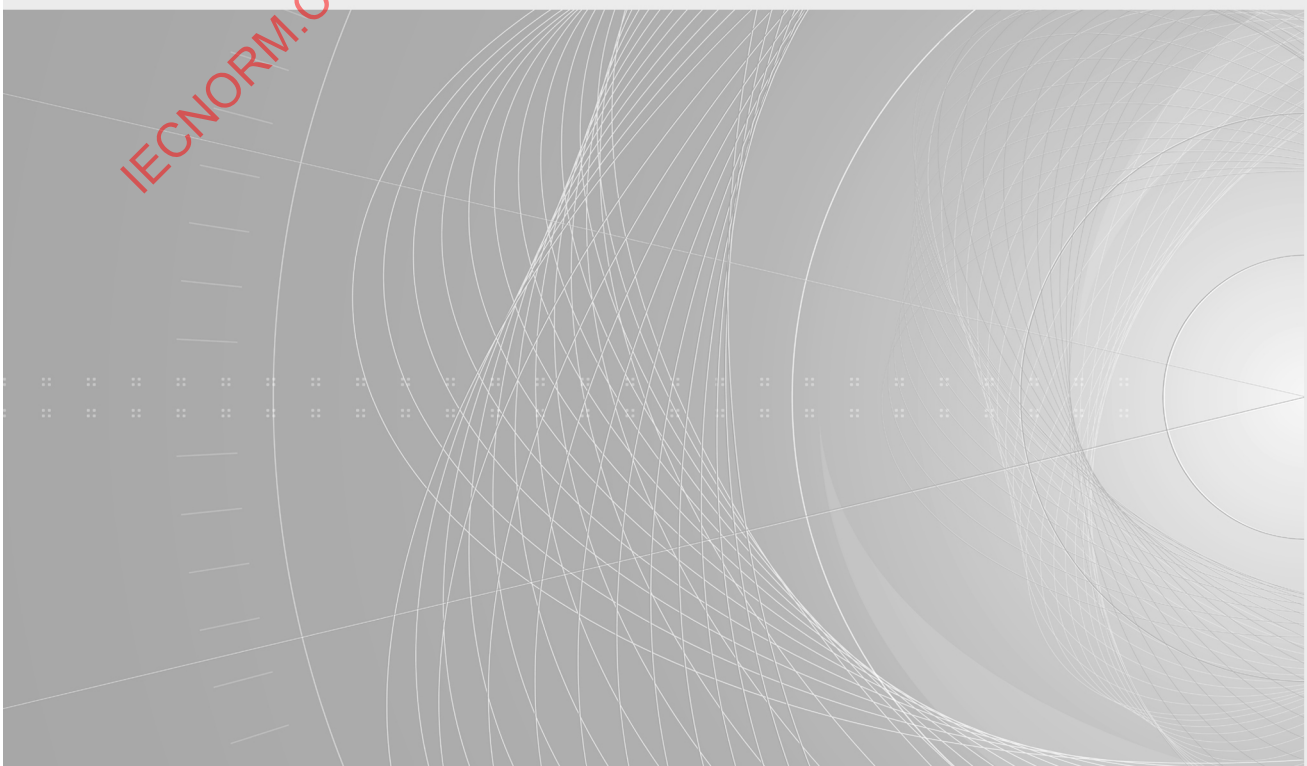
ISO/IEC 14543-4-3

Edition 1.0 2015-09

INTERNATIONAL STANDARD

colour inside

Information technology – Home electronic system (HES) architecture – Part 4-3: Application layer interface to lower communications layers for network enhanced control devices of HES Class 1



ISO/IEC 14543-4-3:2015-09(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 corrigendum or an amendment might have been published.

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 It also gives information on projects, replaced and withdrawn publications.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 once a month by email.

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.

Electropedia - www.electropedia.org
The world's leading online dictionary on electrotechnology, containing more than 22 000 terminological entries 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 between 2002 and 2015. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

ISO/IEC 14543-4-3

Edition 1.0 2015-09

INTERNATIONAL STANDARD

colour inside

Information technology – Home electronic system (HES) architecture – Part 4-3: Application layer interface to lower communications layers for network enhanced control devices of HES Class 1

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 35.240.67

ISBN 978-2-8322-2868-5

CONTENTS

INFORMATION TECHNOLOGY – HOME ELECTRONIC SYSTEM (HES) ARCHITECTURE –

Part 4-3: Application layer interface to lower communications layers for network enhanced control devices of HES Class 1

FOREWORD

1) ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

2) The formal decisions or agreements of IEC and ISO 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 and ISO member bodies.

3) IEC, ISO and ISO/IEC publications have the form of recommendations for international use and are accepted by IEC National Committees and ISO member bodies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC, ISO and ISO/IEC publications is accurate, IEC or ISO 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 and ISO member bodies undertake to apply IEC, ISO and ISO/IEC publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any ISO, IEC or ISO/IEC publication and the corresponding national or regional publication should be clearly indicated in the latter.

5) ISO and IEC do not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. ISO or IEC are 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 ISO or its directors, employees, servants or agents including individual experts and members of their technical committees and IEC National Committees or ISO member bodies 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 of, use of, or reliance upon, this ISO/IEC publication or any other IEC, ISO or ISO/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 ISO/IEC publication may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 14543-4-3 was prepared by subcommittee 25: Interconnection of information technology equipment, of ISO/IEC joint technical committee 1: Information technology.

The list of all currently available parts of the ISO/IEC 14543 series, under the general title *Information technology – Home electronic system (HES) architecture*, can be found on the IEC web site and ISO web site.

This International Standard has been approved by vote of the member bodies, and the voting results may be obtained from the address given on the second title page.

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

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

This part of ISO/IEC 14543 specifies the message structure, sequences and protocol of the application layer for use in the Home Electronic System. Some services are targeted for communications between devices. Other services are exclusively reserved for management purposes. Some services can be used for both management and run-time communications. This standard is applicable for energy management services, mobile access, remote appliance maintenance services, home healthcare services, home security services and comfort control. This standard focuses on the application layers (5th layer to 7th layer of the OSI reference model). This standard specifies a message structure that differs from the 12 message structures specified in ISO/IEC 14543-4-1. This standard allows the use of IP addressing or MAC addressing, while ISO/IEC 14543-4-1 specifies a different non-IP address structure. This part depends on routing functions provided by an external IP layer. ISO/IEC 14543-4-1 uses the routing functions specified in ISO/IEC 14543-4-2. Therefore Part 4-3 is an alternative to Part 3-1 plus Part 3-2.

ISO/IEC 14543, *Information technology – Home Electronic System (HES) architecture, provides*

an introduction to specifications for Home Electronic System (HES):

Part 2-1: Introduction and device modularity

and specifications for three types of HES devices:

Parts 3-x Specifications for network based control of HES Class 1

Parts 4-x Specifications for network enhanced control of HES Class 1

Parts 5-x Specifications for intelligent grouping and resource sharing for HES Class 2 and Class 3

INFORMATION TECHNOLOGY – HOME ELECTRONIC SYSTEM (HES) ARCHITECTURE –

Part 4-3: Application layer interface to lower communications layers for network enhanced control devices of HES Class 1

1 Scope

This part of ISO/IEC 14543 specifies the message structure, sequences and protocol of the application layer for use in network enhanced control devices of the Home Electronic System (HES) Class 1. It provides the services and the interface for the user-level process. This application layer protocol is independent of lower communications layers, which support MAC addressing or IP addressing. The communications sequence is based on the application services.

2 Normative references

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

ISO/IEC 14543-2-1, *Information technology – Home electronic system (HES) architecture – Part 2-1: Introduction and device modularity*

ISO/IEC 14543-4-1, *Information technology – Home electronic system (HES) architecture – Part 4-1: Communication layers – Application layer for the network enhanced control devices of HES Class 1*

ISO/IEC 14543-4-2, *Information technology – Home electronic system (HES) architecture – Part 4-2: Communication layers – Transport, network and general parts of data link layer for network enhanced control devices of HES Class 1*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document the terms and definitions given in ISO/IEC 14543-2-1 and the following apply.

3.1.1
NECD communications middleware
middleware between the lower communications layers to the application layer that performs communications processing according to the protocol specified in this standard

3.1.2
NECD communications processing block
processing block for the communications middleware

Note 1 to entry: This block performs communications protocol processing to facilitate remote device control / monitoring processing for application software, stores information for the above and controls various data on the device as well as the status of other devices.

3.1.3
NECD data
NDATA
data region for message exchanged by NECD communications middleware

3.1.4
NECD header
NHD
data containing the protocol type and message format for the NDATA section

3.1.5
NECD object
NOJ
model of information to be disclosed to the network from information owned by the NECD communications processing block, or an access procedure model

Note 1 to entry: The information or control target owned by each device is specified as a property and the operating method (setting, browsing) for this is specified as a service.

3.1.6
NECD property code
NPC
code value related to the NECD property

3.1.7
NECD service
NSV
code value related to the NECD service

3.1.8
NECD frames
frame composed of NHD1, NHD2, TID and NDATA

3.1.9
property value data
data value related to the NECD property code (NPC)

EXAMPLE Status notification or specific setting.

Note 1 to entry: Property value data is controlled by the NECD service (NSV).

3.1.10
transaction ID
TID
parameter to link a sent request with a received response

3.1.11
property data counter
PDC
indication of the size of the NDT region

3.2 Abbreviations

DNOJ	Destination NECD ObJect
IP	Internet Protocol
NDATA	NECD DATA
NDT	NECD DaTa
NECD	Network Enhanced Control Device
NHD	NECD HeaDer
NPC	NECD Property Counters
NSV	NECD SerVice
OPC	Processing Object Property Counter
PDC	Property Data Counter
SNOJ	Source NECD ObJect
TID	Transaction ID
UDP	User Datagram Protocol

4 Conformance

Enhanced control devices of HES Class 1 that claim conformance to this International Standard shall:

- send, receive and process application layer protocol data units as specified in Clause 6;
- provide application services specified in 6.6 that may be needed by devices for which the application is intended.

5 Services of the application layer

5.1 Positioning in communications layers

5.1.1 General

The NECD communications processing block is positioned between application and lower communications layers. This standard provides the specifications of “NECD communications processing block”. In Figure 1, the shaded area shows the communications middleware block to be specified.

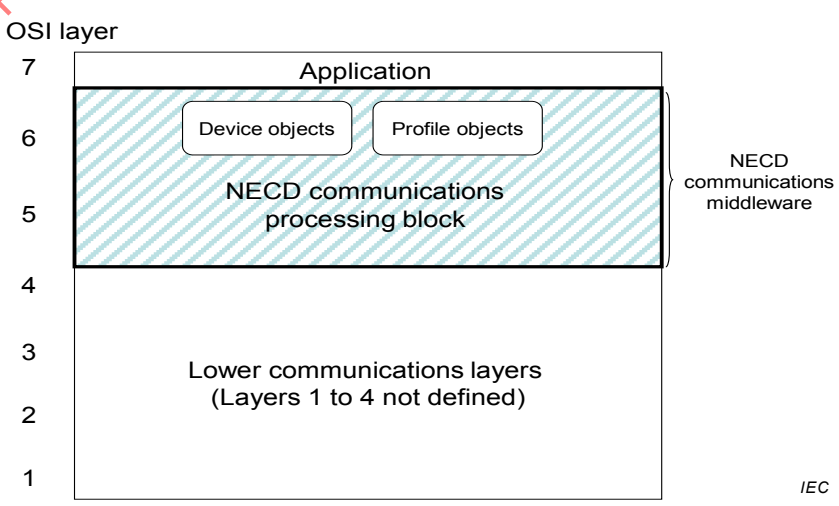


Figure 1 – Communications middleware

As Figure 1 shows, the NECD communications middleware block specified in this standard corresponds to the NECD communications processing block, which is specified as a function that is independent of layers 1 to 4. The NECD communications processing block sends and receives a NECD frame specified in Clause 6. There are two kinds of messages: unicast and broadcast.

Unicast transmission specifies a destination address that is in layer 4 or lower, and transmits the NECD frame to a specific NECD node. Broadcast transmission specifies a destination address that is in layer 4 or lower, and transmits the NECD frame to all the NECD nodes in a subnet. In case of UDP/IP, refer to 5.1.2.

When the transmission system of layer 4 or lower layer corresponds to neither multicasting nor broadcasting, it shall transmit to all the NECD nodes in a subnet using multiple unicast transmissions to achieve the equivalence of a broadcast transmission. The destination address and the method for setting it are not specified, but shall be defined for every lower communications layer.

Security is not specified in the NECD communications processing block. Security standard technologies in layer 4 or lower can be applied as necessary.

5.1.2 When using UDP in layer 4 and IP in layer 3

When using UDP/IP, the following addresses and ports shall be supported.

Each NECD node has an IP address. The IP address range and acquisition method are not specified. If NECD frames are transferred by UDP packets, the destination port number of UDP packets shall be 3 610. The source port number is not specified. For general broadcast (simultaneous transmission), NECD frames are mapped on IP multicast packets and transferred. For IPv4, the destination multicast address value shall be 224.0.23.0. For IPv6, ff02::1 (all-node multicast address) shall be used.

5.2 Service primitives of the application layer

5.2.1 General

The NECD objects are introduced with two objectives:

- compartmentalisation of the functions of devices connected to the NECD network;
- modularisation of communications between devices to enable application software developers to utilise NECD communications without having to consider detailed specifications.

The NECD objects are processed in the NECD communications processing block. Control content exchanged in communications can be classified into those relating to functions unique to each device and those relating to data profiling other than the functions unique to each device. In NECD, all of these are specified as NECD objects, and control and data exchange are achieved to enable their manipulation.

Each NECD object has some properties. The various unique functions possessed by an NECD node are represented as NECD properties. Devices are operated by reading or writing the NECD properties of the NECD object in the relevant NECD node.

NECD objects are defined by the following specifications: object type (codes are specified in 6.5 as NOJ); the properties possessed by each object (codes are specified in 6.8 as NPC); and the services for those properties (codes are specified in 6.6 as NSV).

NOTE It is assumed that each NECD node would have more than one NECD object of the same type (e.g., two human detection sensor objects in the same node), and that identification could be performed by stipulating a specific code.

5.2.2 NECD objects from the viewpoint of application software

Control from application software is described for the three main cases listed below, with a focus on how the NECD objects are perceived.

- Case 1: Obtaining the status of another node
- Case 2: Controlling the functions of other nodes
- Case 3: Notifying own node status to other nodes

5.2.3 Case 1: Obtaining the status of another node

This standard provides two methods: synchronous type and asynchronous type for obtaining the status of another node. Each device can select the synchronous type or asynchronous type. These methods are shown in Figure 2 (synchronous type) and Figure 3 (asynchronous type). In the method shown in Figure 2, when the NECD communications middleware receives a request from an application, the NECD communications middleware sends the request to obtain the status of another node to the target node (Node B). After that NECD communications middleware receives the results, NECD communications middleware notifies the application of the status. With this method, object data for the other node need not be stored in the NECD communications middleware for the node (Node A in Figure 2 and Figure 3), which sends the request. In the second method, shown in Figure 3, even when the NECD communications middleware does not receive any request from an application, it receives and holds the notified status of objects in other nodes in advance, and then returns them to an application when it receives a request. In this method, objects copied to NECD objects in other nodes actually exist within the NECD communications middleware.

In the former method (Figure 2), a virtual copy of the NECD objects in the other nodes exists in the NECD communications middleware because access is performed from an application. In the latter method (Figure 3), a copy of each property of the NECD objects in the other nodes exists in the NECD communications middleware. In both cases, in order to set the desired NECD object instance, not only the NECD object class code, but also an instance code and data that is specifying the node are necessary. From the viewpoint of the application, therefore, NECD objects are represented using the relationship shown in Figure 4 within the NECD communications middleware.

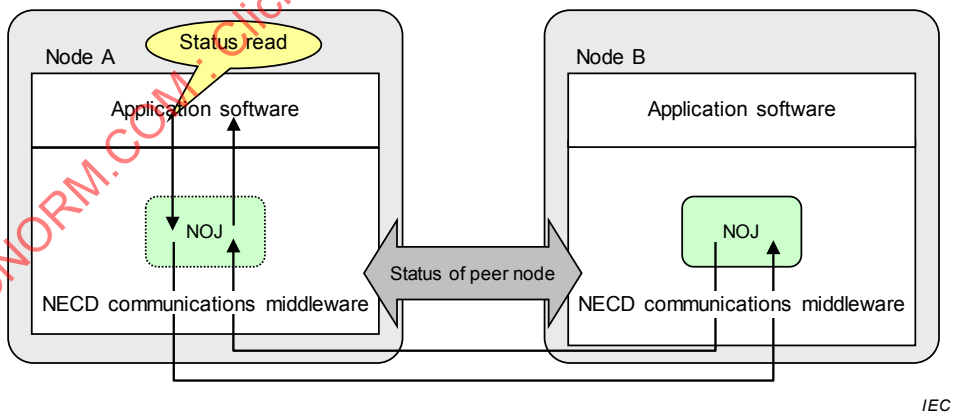


Figure 2 – Acquisition of status of another node (synchronous type)

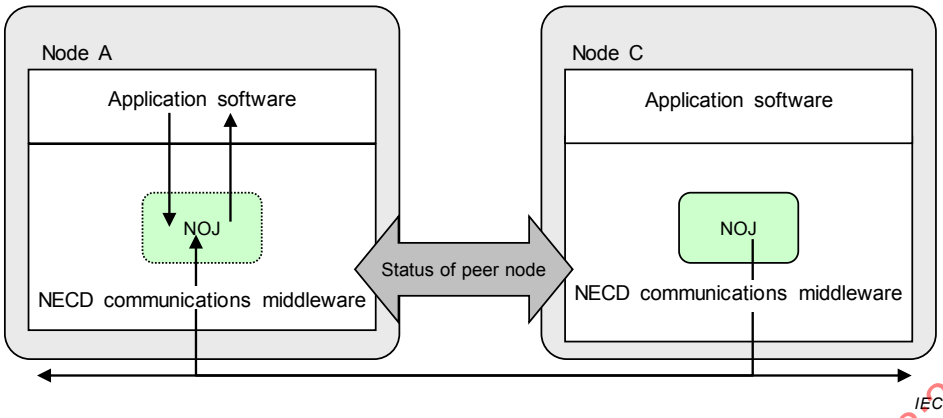


Figure 3 – Acquisition of status of another node (asynchronous type)

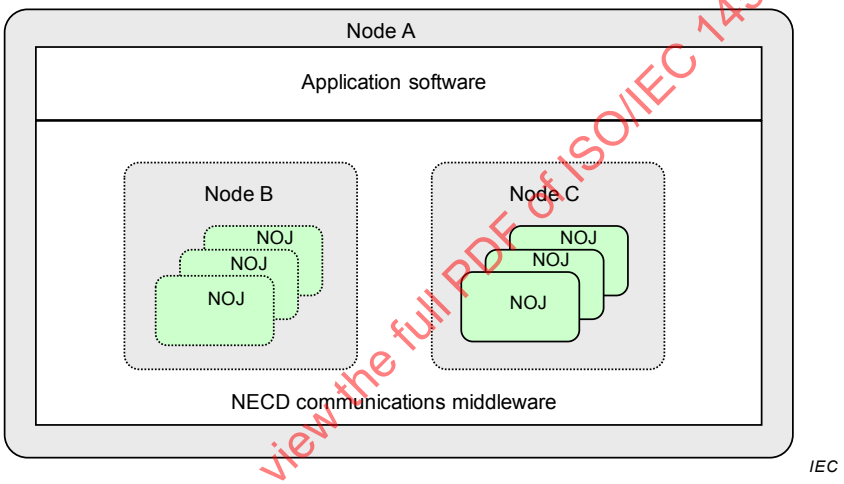


Figure 4 – Objects seen from application software

5.2.4 Case 2: Controlling the functions of other nodes

The NECD provides a method for controlling the functions of other nodes, as shown in Figure 5. Just as in Figure 2, however, a request for control (property value setting) is issued to objects in the target node (Node B), and the application is then notified of the results (although there are exceptions to this). Basically, therefore, property data for objects in the other node (Node B) need not be present in the NECD communications middleware for the node (Node A), which sends the request. From the viewpoint of the application, the NECD objects are seen in the relationship shown by Node B in Figure 6 within the NECD communications middleware.

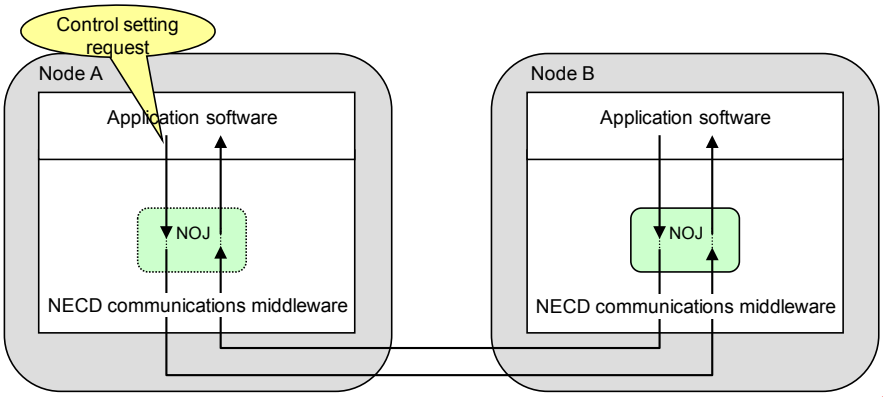


Figure 5 – Method of controlling other nodes

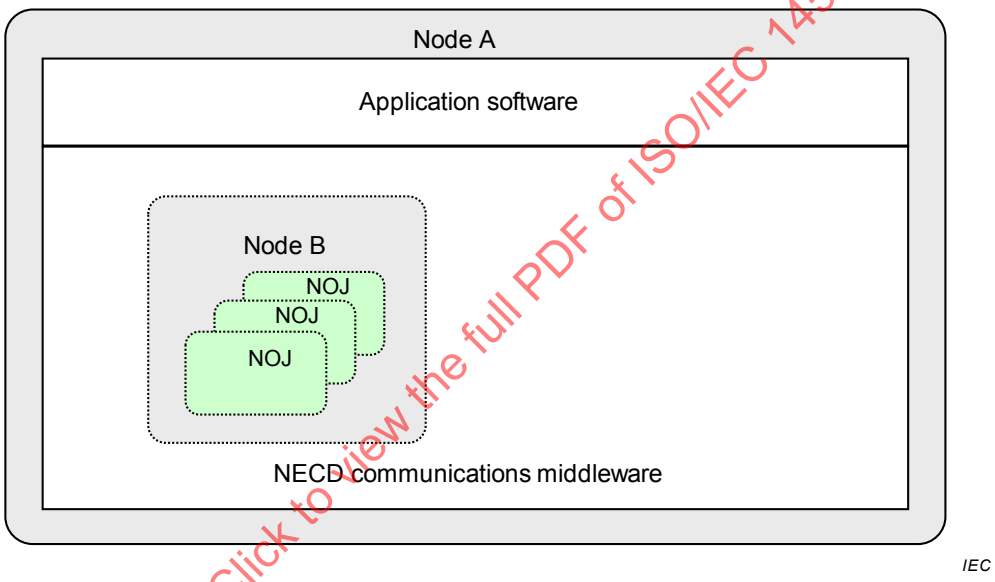


Figure 6 – Objects seen from application software

5.2.5 Case 3: Notifying own node status to other nodes

The NECD provides two methods: “synchronous type” and “asynchronous type” for notifying the status of the own node to the application software on another node. Each device can select the synchronous type or asynchronous type. These methods are shown in Figure 7 (synchronous type) and Figure 8 (asynchronous type). In the method shown in Figure 7, when the NECD communications middleware receives a request from an application, it sends the notification to the other specified node (Node B) immediately. In this case, the device status need not be stored as an object in the NECD communications middleware for the node (Node A) that is notifying the status.

In the second method, shown in Figure 8, when the NECD communications middleware receives a request from an application, it periodically notifies the property value to the other node. In this case, the NECD object data actually exists in the NECD communications middleware. In either case, from the viewpoint of the application, the NECD objects of the own node are seen as existing within the NECD communications middleware (Figure 9).

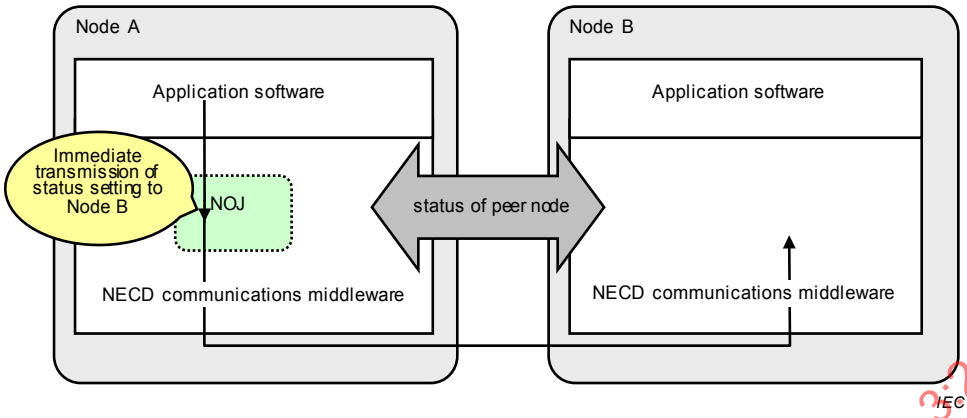


Figure 7 – Method of notification to other nodes (synchronous type)

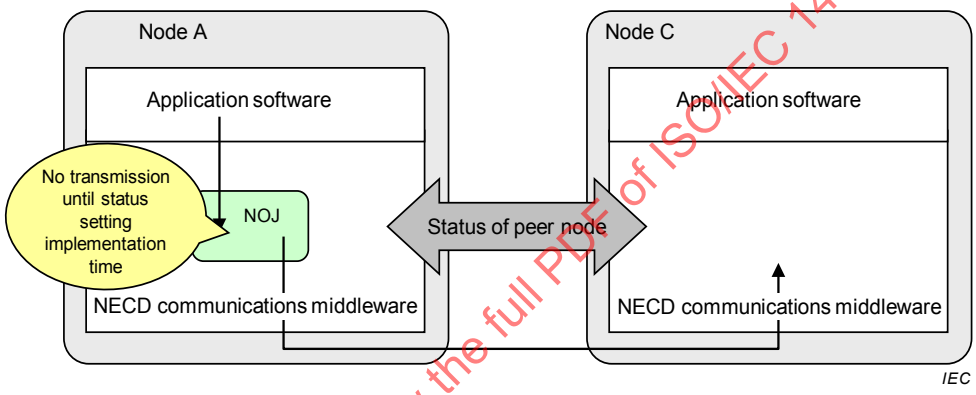


Figure 8 – Method of notification to other nodes (asynchronous type)

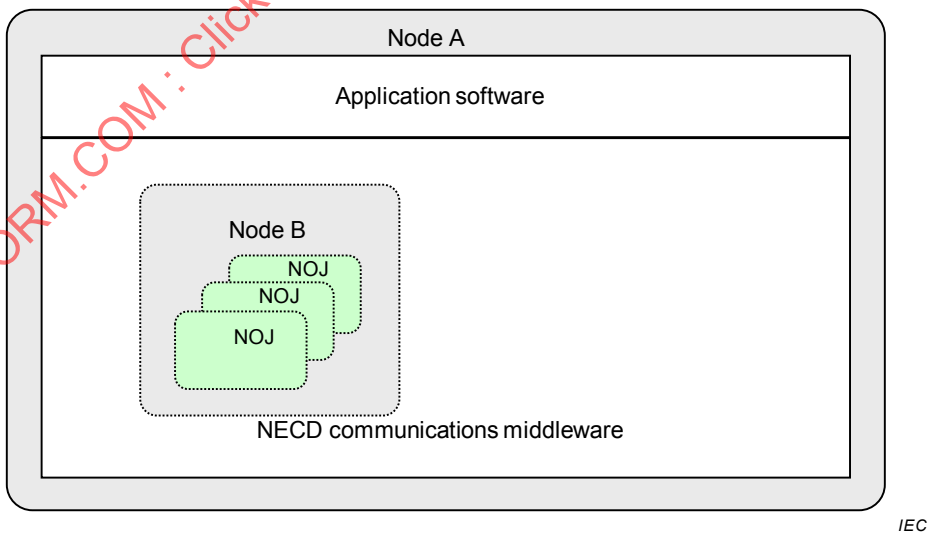


Figure 9 – Objects seen from application software

As is clear from the three cases shown above, the NECD communications middleware is viewed by the application software as containing (and in some cases actually does contain)

a) a collection of NECD objects of the own node whose role is to disclose the functions of the own node to other nodes and to be controlled by other nodes; and

b) NECD objects at the node level whose role is to control and obtain the status of the functions of other nodes.

Own node properties exist in the NECD communications middleware of only one device. The middleware also may contain properties of other related nodes.

Based on the above, Figure 10 shows an example of the NECD communications middleware object configuration for a system in which an air conditioner, ventilation fan and motion detection sensor are connected as separate nodes via a network, seen from the perspective of the application software in the air conditioner.

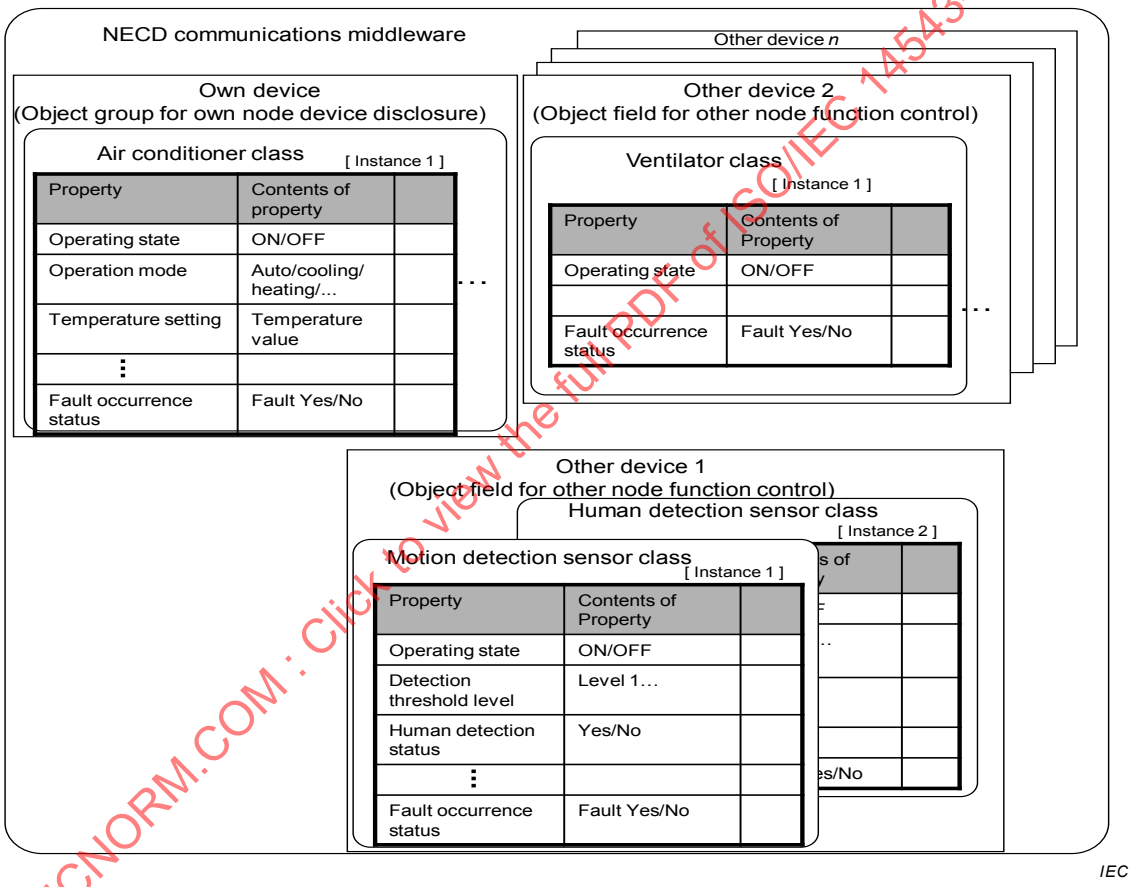


Figure 10 – Example of object configuration

6 Application layer protocol data unit (APDU)

6.1 Overview

To reduce the load of simple devices, NECD specifies the frame format for the NECD communications middleware block to minimise the message size while fulfilling the requirements of the communications layer structure.

Figure 11 shows the format of NECD frames processed by the NECD communications middleware. Detailed specifications for each message component are described in 6.2 to 6.10.

In these specifications, messages exchanged between the NECD communications processing blocks are called NECD frames. NECD frames are divided into two types depending on the specified NHD (see 6.2): message format specified by the NECD (Format 1) and message format unique to manufacturers (Format 2). Format 2 is specified so that each manufacturer can provide additional functions. The NECD frame length depends on the lower-layer communications media.

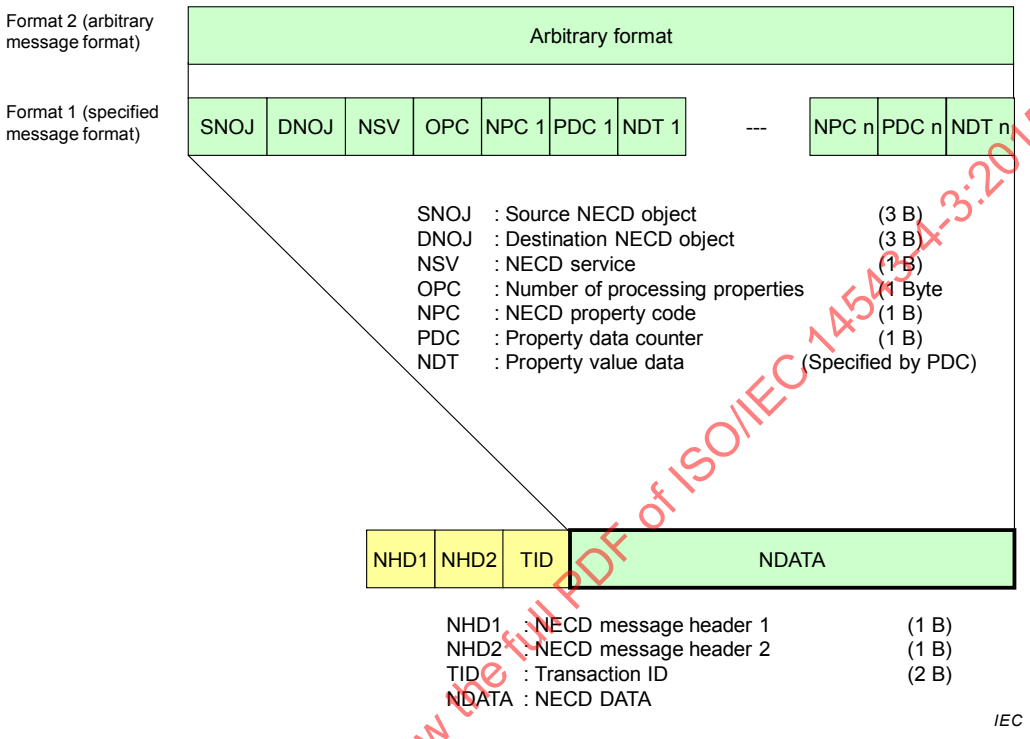


Figure 11 – NECD frame format

6.2 NECD header (NHD)

6.2.1 Overview

NHD consists of an NECD header 1 and an NECD header 2.

6.2.2 NECD header 1 (NHD1)

Figure 12 shows the detailed specifications of the NECD header 1 (NHD1) byte. The contents of this byte are shown in Figure 12.

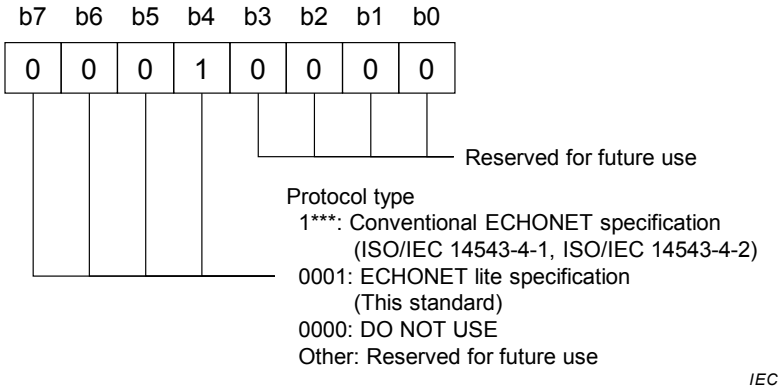


Figure 12 – Bit specifications of NHD 1

The combination of b7 to b4 specifies an NECD protocol type. "b7:b6:b5:b4 = 0:0:0:1" indicates the NECD protocol defined in this standard "b7:b6:b5:b4 = 0:0:0:0" shall not be used because it would prevent compatibility with ISO/IEC 14543-4-1 and ISO/IEC 14543-4-2.

6.2.3 NECD header 2 (NHD2)

Figure 13 shows the detailed specifications of NECD header 2 (NHD2) shown in Figure 11.

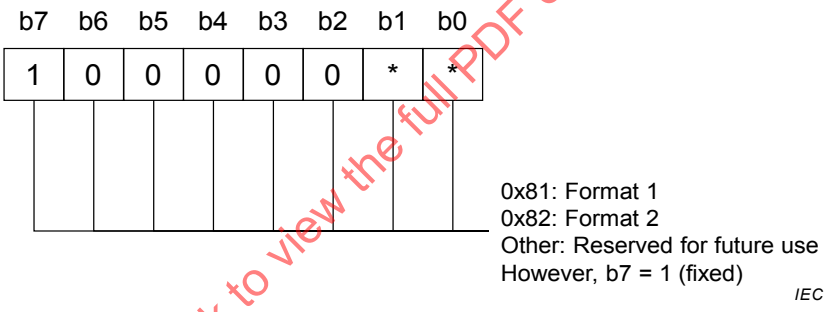


Figure 13 – Detailed specifications of NHD 2

NHD2 defines the NDATA frame format. When the value of NHD2 is 0x81, the NDATA frame format is Format 1 (specified message format) defined in this standard. When the value of NHD2 is 0x82, the NDATA frame format is Format 2 (arbitrary message format). For coexistence with the ISO/IEC 14543-4-1 and ISO/IEC 14543-4-2 Protocol, b7 shall be constantly set to 1.

6.3 Transaction ID (TID)

TID is a parameter to associate a received response with a sent request when the request sender receives the response in NECD communications. The response sender shall store the same value as that contained in the request message.

6.4 NECD data (NDATA)

NDATA refers to the data area of a message exchanged by the NECD communications middleware.

6.5 NECD object (NOJ)

Figure 14 shows the detailed specifications of NECD objects in Figure 11.

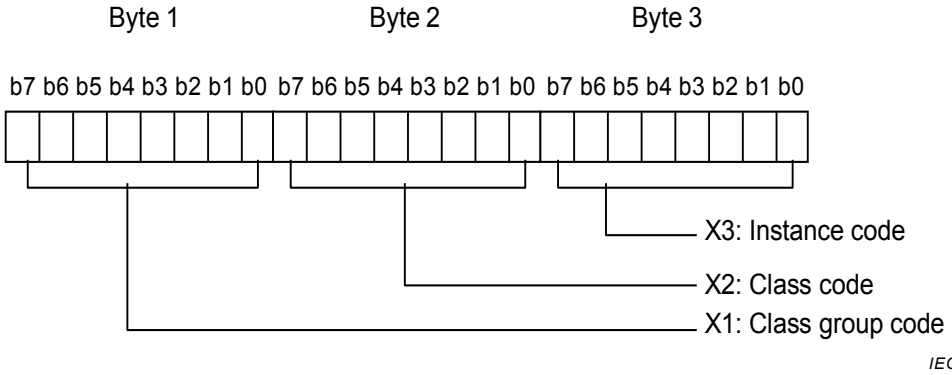


Figure 14 – Bit specifications of the NOJ code

NECD objects are described using the formats [X1.X2] and [X3], to be specified as shown below. (However, "." is used only for descriptive purposes and does not mean a specific code.) The object class is designated by the combination of X1 and X2, while X3 shows the class instance. A single NECD node may contain more than one instance of the same class, in which case X3 is used to identify each one.

The instance code 0x00 is regarded as a special code (code for specifying all instances). When a DNOJ having this specified code is received, it is handled as a code specifying general broadcast to all instances of a specified class.

- X1: Class group code

  0x00–0xFF.
- X2: Class code

  0x00–0xFF.
- X3: Instance code

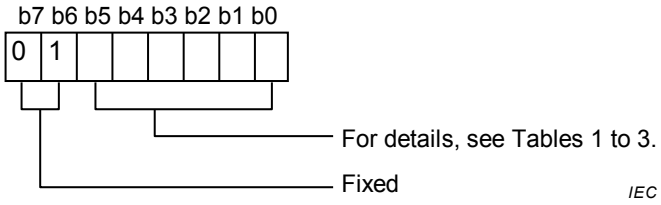
  0x00 – 0x7F. This is an identification code when more than one of the same class as that of the attributes specified by [X1.X2] exists in the same node.

  However, 0x00 is used for general broadcast to instances of the same class.

6.6 NECD Service (NSV)

6.6.1 Overview

This subclause provides detailed specifications for the NECD service (NSV) code shown in Figure 15.



NOTE Except when b7:b6=0:1, b0 to b5 have different meanings.

Figure 15 – Bit specifications of the NSV code

It specifies a simultaneous action for one or more properties stipulated by the NPC. However, it does not stipulate the order of operations. The order of property operations is an implementation issue.

The following three types of operations are provided. The response is subdivided into two types: “response” and “response not possible”. The “response” is used when the service request in relation to all the NPC-stipulated properties is accepted. The “response not possible” is used when one or more specified properties do not exist or when the specified service cannot be processed for one or more properties.

“Request,” “Response” (response/response not possible) and “Notification”.

The “response” is a response to a “request” that requires a response. It shall be returned when an NOJ-stipulated object exists. When the service-processing request related to all the NPC-stipulated properties is accepted, the “response” shall be returned. If the processing request related to one or more specified properties cannot be accepted or if the object exists but one or more properties do not exist, “response not possible” shall be returned. When the “request” does not require any response or when the specified object does not exist, no “response” will be returned.

There are two types of “notification”: one for transmitting own property information autonomously and the other for sending a response to a notification request. However, these two types have the same code.

Four specific operations are provided:

- write (response required / no response required),
- read,
- read & write and
- notification (notification / notification with response required).

The six operations shown below are set:

- property value write (response required),
- property value write (no response required),
- property value read,
- property value write and read,
- property value notification,
- property value notification (response required).

The NSV and message configuration and their relationship to NPC and NSV are described here. The NPC of an NECD message is such that the NSV value determines whether the target object is stipulated by the SNOJ or DNOJ. When the NSV is a “response” or “notification”, the NPC forms a SNOJ-stipulated object and the “response” or “notification” is addressed to a DNOJ-stipulated object. On the other hand, when the NSV is a “request”, the NPC forms a DNOJ and the “request” is issued from an SNOJ-stipulated object.

If there is no NOJ to be set as SNOJ or DNOJ, a node profile class shall be specified.

Table 1, Table 2 and Table 3 show specific NSV code assignments based on the content described above. (The related number is indicated in the remarks column of the table.)

In the diagrams in 6.6.2 to 6.6.7, the NOJ values used in relation to “requests” are individually specified codes. However, although a service request is made to two or more non-specific object instances using a single message when the NOJ value indicates general broadcast to all instances of the specified class (i.e., X3 = 0x00), the processing in such a case shall assume that a request message was sent individually to each instance. That is, when it is necessary to send response messages, they shall be generated in such a manner that the number of instances equals the number of response messages, and messages with contents that match the individual instances shall be sent after storing such contents.

Figure 16 shows a sequence diagram of the relationship between individual NSVs.

Table 1 – List of NSV Codes for Requests

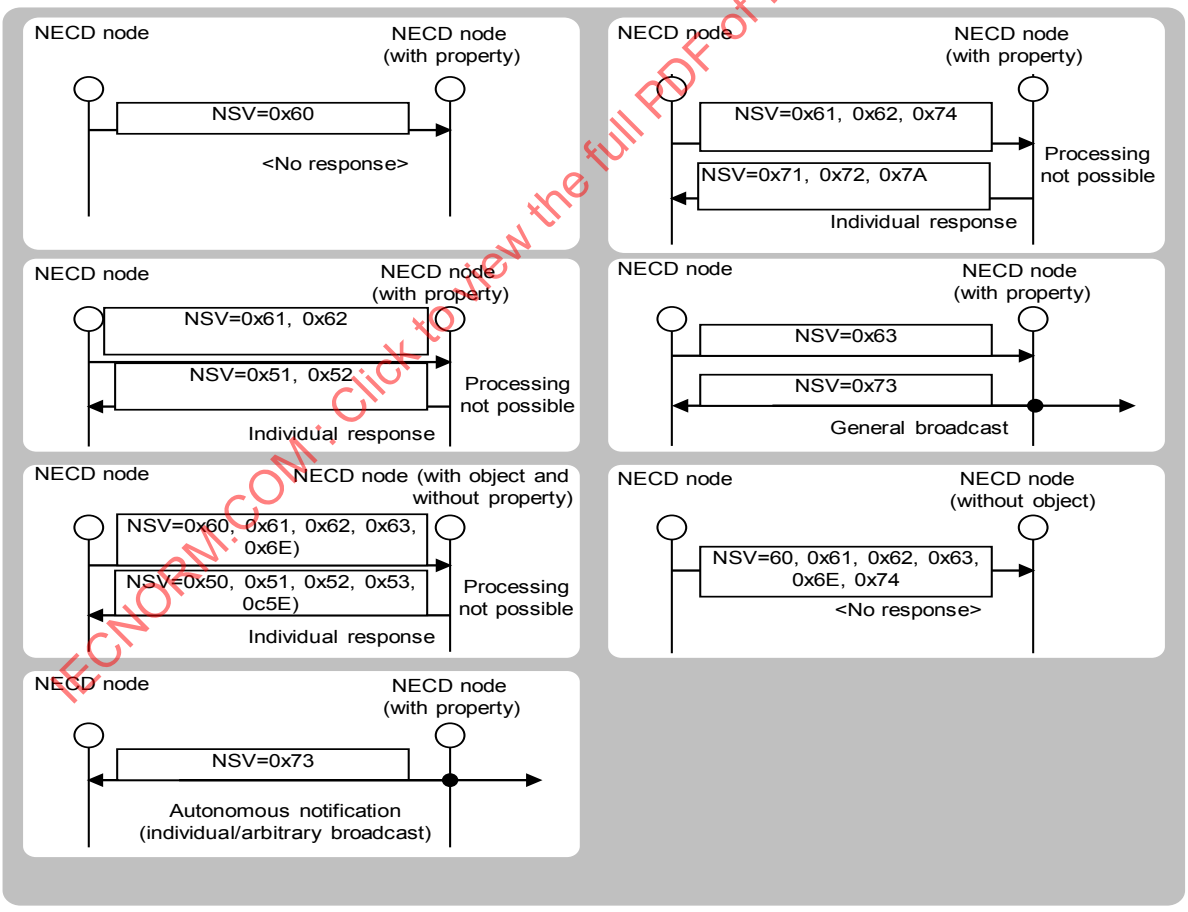
Service code NSV	NECD service content	Symbol	Remarks
0x60	Property value write request (no response required)	Set	Broadcast possible
0x61	Property value write request (response required)	SetC	
0x62	Property value read request	Get	Broadcast possible
0x63	Property value notification request	INF_REQ	Broadcast possible
0x64–0x6D	Reserved for future use		
0x6E	Property value write and read request	SetGet	Broadcast possible
0x6F	Reserved for future use		

Table 2 – List of NSV codes for response/notification

Service code NSV	NECD service content	Symbol	Remarks
0x71	Property value write response	Set_Res	NSV = 0x61 response; Individual response
0x72	Property value read response	Get_Res	NSV = 0x62 response; Individual response
0x73	Property value notification	INF	NSV = 0x63 response; Broadcast response Both individual notification and broadcast notification
0x74	Property value notification (response required)	INFC	Individual notification
0x75–0x79	Reserved for future use		
0x7A	Property value notification response	INFC_Res	NSV = 0x74 response; Individual response
0x7B–0x7D	Reserved for future use		
0x7E	Property value write and read response	SetGet_Res	NSV = 0x6E response; individual response
0x7F	Reserved for future use		

Table 3 – List of NSV codes for "Response not possible"

Service code NSV	NECD service content	Symbol	Remarks
0x50	Property value write request "response not possible"	SetI_SNA	NSV = 0x60 response not possible; Individual response
0x51	Property value write request "response not possible"	SetC_SNA	NSV = 0x61 response not possible; Individual response
0x52	Property value read "response not possible"	Get_SNA	NSV = 0x62 response not possible; Individual response
0x53	Property value notification "response not possible"	INF_SNA	NSV = 0x63 response not possible; Individual response
0x54–0x5D	Reserved for future use		
0x5E	Property value write and read "response not possible"	SetGetI_SNA	NSV = 0x6E response not possible; Individual response
0x5F	Reserved for future use		



IEC

Figure 16 – Sequence diagram for NSV transmission and reception

6.6.2 Property value write service (no response required) [0x60, 0x50]

A “request” (0x60) indicates a request to write the content shown in the NDT to the property stipulated in the NPC of the DNOJ-stipulated object. If more than one property is set in one message, this standard does not specify the order of the property controlled.

When the request is not accepted, or when the stipulated DNOJ exists but the stipulated NPC does not exist, “response not possible” (0x50) is returned. When the specified DNOJ exists but there are too many target properties in the control request to process all, the number of properties processed from the beginning is stored in OPC and “response not possible” (0x50) is returned as a response. Then the destination address of the lower communications layer shall be the source of “request” (the source address of the “request” message in the lower communications layer).

When the relevant object itself does not exist, neither “response” nor “response not possible” is returned. (See Figure 17 for the exchange procedure.)

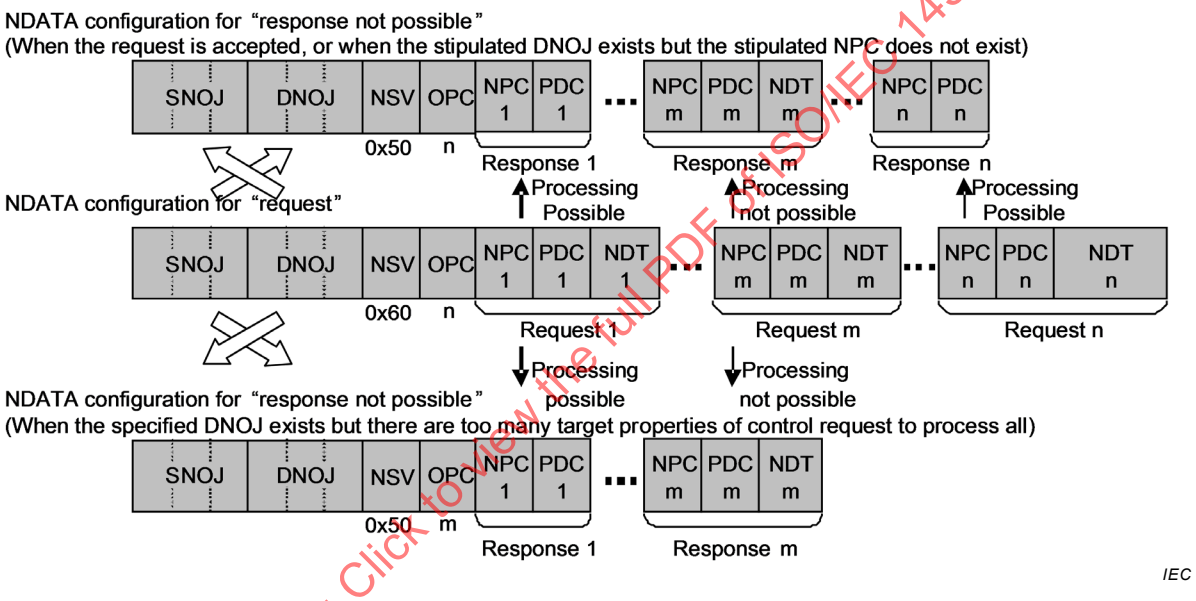


Figure 17 – NDATA configuration for property value write service (no response required)

6.6.3 Property value write service (response required) [0x61, 0x71, 0x51]

A “request” (0x61) indicates a request to write the content shown in the NDT to the property stipulated in the NPC of the DNOJ-stipulated object. If more than one property is set in one message, this standard does not specify the order of the property controlled.

In response to this “request,” when the request is to be (or has already been) accepted, a “response” (0x71) is returned. However, this “response” is not a processing implementation response. In the frame format for response, the value of the object stipulated by the request is set in the SNOJ, and the same value as for the request is set in the OPC. In the NPC, the same property code for the request is set. To indicate that the request was accepted, the PDC is set to 0 and no NDT is attached.

When the request is not accepted, or when the stipulated DNOJ exists but the stipulated NPC does not exist, “response not possible” (0x51) is returned. As with a message of “response,” the request-stipulated object value is set in the SNOJ, the request-source object value is set in the DNOJ, the same value as for the request is set in the OPC, and the same property code for the request in the NPC for a message of “response not possible”. For the NPC that accepted the request, 0 is set in the succeeding PDC and no NDT is attached. For the NPC

that did not accept the request, the same value as for the request is set in the succeeding PDC and the requested NDT is attached to indicate that the request could not be accepted.

When the specified DNOJ exists but there are too many target properties in a control request to process all, the number of properties processed from the beginning is stored in the OPC, the same property code for the request is set in the NPC and 0 is set in the PDC. Then "response not possible" (0x51) is returned as a response. In this case, the responding side can determine the number of property values to be returned. However, the sequence of such properties shall be the same as in the request message.

When the relevant object itself does not exist, neither "response" nor "response not possible" is returned (see Figure 18 for the sequence). Whether a response is possible or not, the destination address of the lower communications layer shall be the source of "request" (the source address of the "request" message in the lower communications layer).

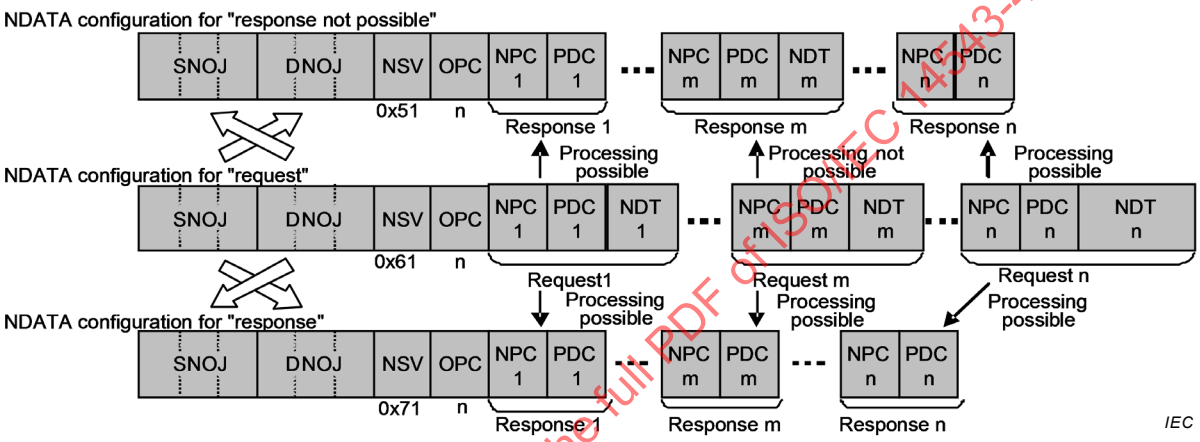


Figure 18 – NDATA configuration for property value write service (response required)

6.6.4 Property value read service [0x62, 0x72, 0x52]

A "read" (0x62) indicates a request to read the NPC-stipulated properties from the DNOJ-stipulated object. If more than one property is set in one message, this standard does not specify the order of the property monitored. In the case of request, the value of PDC shall be 0.

When the request is to be (or has already been) accepted for all properties, a "response" (0x72) is returned. In the frame format for response, the value of the object stipulated by the request is set in the SNOJ, and the value of the request-source object is set in the DNOJ. In the OPC, the same value as for the request is set. To indicate that the request was accepted, the length of the read property is set in the PDC and the read property value in the NDT.

When the request is not accepted, or when the stipulated DNOJ exists, but the stipulated NPC does not exist, "response not possible" (0x52) is returned. As with a message of "response," the request-stipulated object value is set in the SNOJ, the request-source object value is set in the DNOJ, the same value as for the request is set in the OPC, and the same property code for the request is set in the NPC for a message of "response not possible". For the NPC that accepted the request, the length of the read property is set in the succeeding PDC and the read property value in the NDT. For the NPC that did not accept the request, 0 is set in the succeeding PDC and no NDT is attached to indicate that the request was not accepted.

When the specified DNOJ exists but there are too many target properties of control request to process, or all the property values requested for read cannot be returned because the allowable message length is not enough, the number of properties processed from the beginning is stored in the OPC, the same property code for the request is set in the NPC, the

length of the read property is set in the PDC, and the read property value is set in the NDT. Then “response not possible” (0x52) is returned as a response. In this case, the responding side can determine the number of property values to be returned. However, the sequence of such properties shall be the same as in the request message.

When the relevant object itself does not exist, neither “response” nor “response not possible” is returned (see Figure 19 for the sequence). Whether a response is possible or not, the destination address of the lower communications layer shall be the source of “request” (the source address of the “request” message in the lower communications layer).

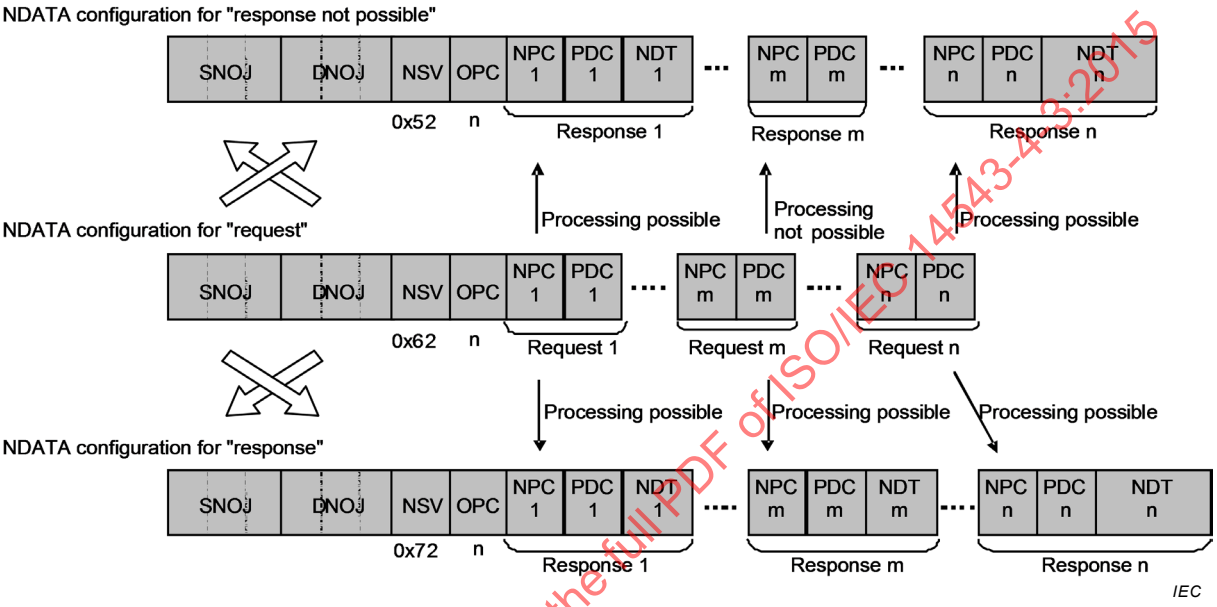


Figure 19 – NDATA configuration for property value read service

6.6.5 Property value write and read service [0x6E, 0x7E, 0x5E]

“Write and read” (0x6E) indicates a service to process two requests by a single message: a request for writing NDT-stipulated contents into NPC-stipulated properties of a DNOJ-stipulated object and a request for the contents of NPC-stipulated properties from a DNOJ-stipulated object. The number of write-requested properties is stored in the OPCSet and that of read-requested properties is set in the OPCGet. The value of PDC for “Get request” shall be 0. The value of PDC for “Set request” shall be the length of relevant NDT.

The order of processing "write requests" and "read requests" is not specified. Moreover, if more than one property is set in one message, this standard does not specify the order of the property controlled.

When the request is to be (or has already been) accepted, a “response” (0x71) is returned. In the frame format for response, the value of the object stipulated by the request is set in the SNOJ and the request-source object value in the DNOJ. The same value as the request is set in the OPCSet the same property code for the request is set in the NPC, 0 is set in the PDC, and no NDT is attached. The same value as for the request is set in the OPCGet, the same property code for the request is set in the NPC, the length of the read property is set in the PDC, and the read property value is set in the NDT.

When the request is not accepted, or when the stipulated DNOJ exists but the stipulated NPC does not exist, “response not possible” (0x5E) is returned. When the specified DNOJ exists but there are too many target properties of control request to process, or all the property values requested for write or read cannot be returned because the allowable message length is not enough, the number of properties processed from the beginning is stored in the OPCSet

and OPCGet. Then “response not possible” (0x5E) is returned as a response. In this case, the responding side can determine the number of property values to be returned. However, the sequence of such properties shall be the same as in the request message.

When the relevant object itself does not exist, neither “response” nor “response not possible” is returned (see Figure 20 for the sequence). Whether a response is possible or not, the destination address of the lower communications layer shall be the source of “request” (the source address of the “request” message in the lower communications layer).

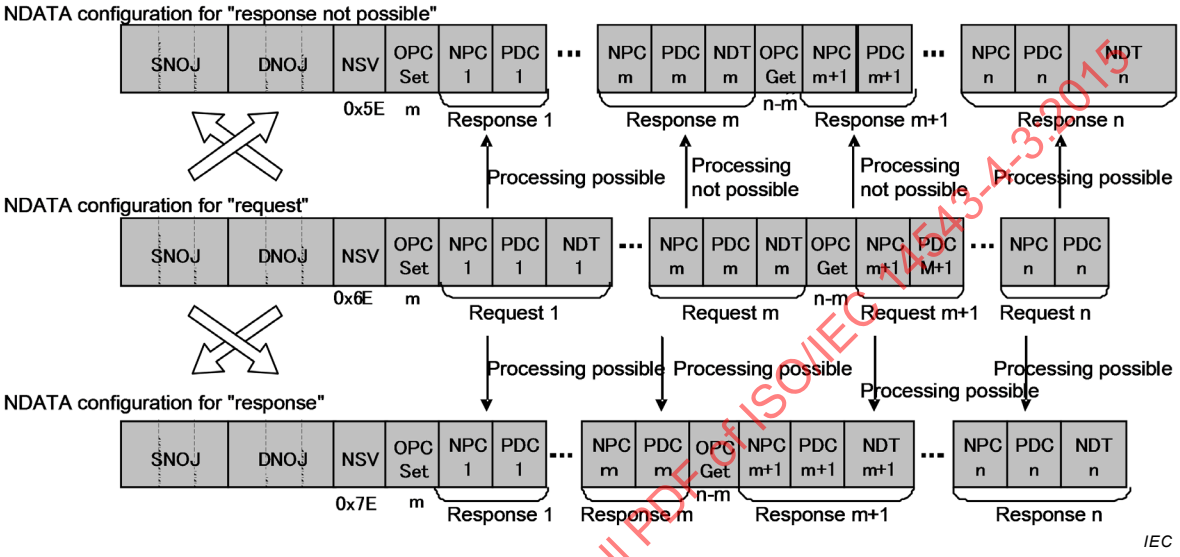


Figure 20 – NDATA configuration for property value write and read service

This service is an option. If a node not supporting this service receives a request for the service, 0 shall be set in the OPCSet and OPCGet and a “response not possible” (0x5E) shall be returned as a response.

6.6.6 Property value notification service [0x63, 0x73, 0x53]

There are two types of “notification”: the notification sent as a response to a “notify request” (0x63) and the autonomous notification, which is unrelated to notify requests. The codes for the two types are identical. (Here, notification in response to a “notify request” signifies an announcement that does not specify the property value [content], while an autonomous notification is a voluntary announcement that was not made in response to a request.) In the case of a “notify request” (0x63), this indicates a request to notify (by general broadcast; hereafter “announce” will signify a general broadcast) the content of the property stipulated in the NPC of the DNOJ-stipulated object. If more than one property is set in one message, this standard does not specify about the order of the property notified. In the case of request, the value of PDC shall be 0.

In response to this “notify request,” when the request is to be accepted, a “response” (0x73) value is notified. The request-stipulated object value is set in the SNOJ, the request-source object value is set in the DNOJ, and the same value as for the request is set in the OPC. The same property code as for the request is set in the NPC and the property length of notification is set in the PDC. In the NDT, the requested property value (contents of notification) is stored. For broadcast, destination addresses in lower communications layers are set.

When the request is not accepted, or when the stipulated DNOJ exists but the stipulated NPC does not exist, “response not possible” (0x53) is returned. In the same way as for a message of “response”, the request-stipulated object value is set in the SNOJ, the request-source object value is set in the DNOJ, the same value as for the request is set in the OPC, and the same property code for the request in the NPC for a message of “response not possible”. For

the NPC that accepted the request, the length of the read property is set in the succeeding PDC and the read property value in the NDT. For the NPC that did not accept the request, 0 is set in the succeeding PDC and no NDT is attached to indicate that the request was not accepted. When the specified DNOJ exists but there are too many target properties of control request to process, or the property value (contents of notification) requested for read cannot be returned because the allowable message length is not enough, the number of properties processed from the beginning is stored in the OPC, the same property code for the request is set in the NPC, the length of the read property is set in the PDC, and the read property value is set in the NDT. Then “response not possible” (0x53) is returned as a response. In this case, the responding side can determine the number of property values to be returned. Also, for a response not possible, the address of the lower communications layer of the request source shall be set as the destination address of the lower communications layer. When the relevant object itself does not exist, neither “response” nor “response not possible” is returned. (See Figure 21 for the sequence.) In the case of an autonomous “notification”, the DEA is set to a general broadcast for a required status change notification. In the other cases, however, the destination of the lower communications layer can be set arbitrarily for broadcast or individual transmission.

For an autonomous “notification”, a node profile class is stored because there is no NOJ to be set in the DNOJ in particular.

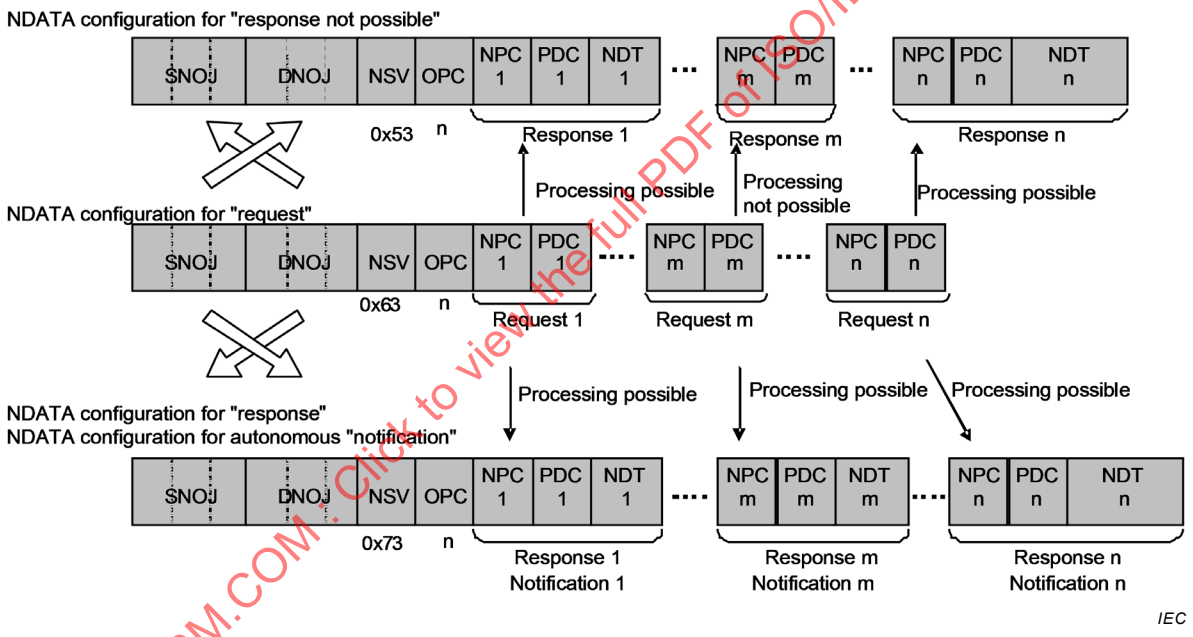


Figure 21 – NDATA configuration for property value notification service

6.6.7 Property value notification (response required) [0x74, 0x7A]

The “notification (response required)” (0x74) autonomously notifies a specific node of the property value stipulated by the NPC of the SNOJ-stipulated object and requests a response. The response process for this “notification (response required)” varies depending on whether or not the DNOJ is specified.

Processing varies depending on whether the specified DNOJ exists. When the specified DNOJ exists, a “response” (0x7A) for autonomous notification reception is returned (see Figure 22). In a response message, the requested object value is set in the SNOJ and the request-source object value in the DNOJ. The same value as for notification is set in the OPC and the same property code as for notification is set in the NPC. To indicate that the notification was received, the PDC is set to 0 and no NDT is attached.

When the specified DNOJ does not exist, the message shall be discarded. The nodes that received the notification (response required) by broadcast shall discard this message.

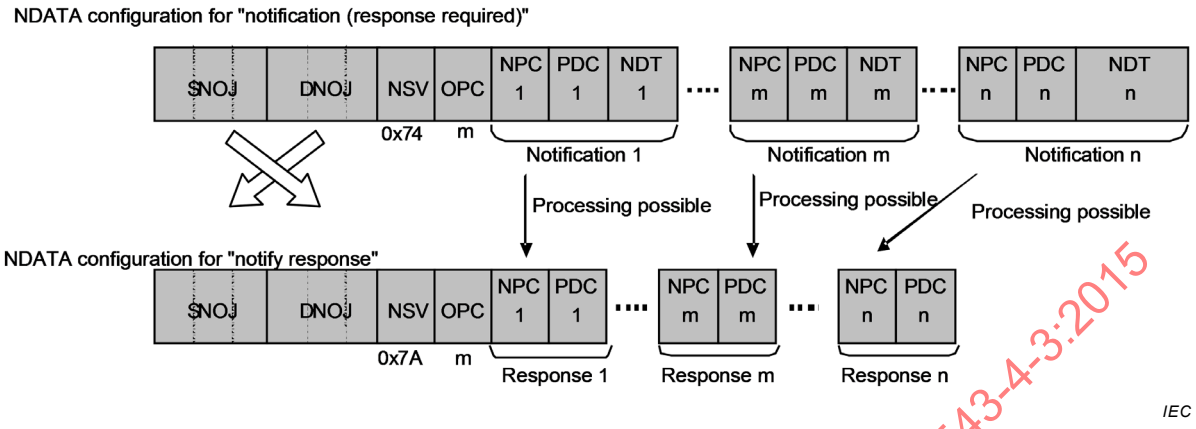


Figure 22 – NDATA configuration for property value notification (response required) service

6.7 Processing object property counters (OPC, OPCSet and OPCGet)

A target property counter consists of 1 B. For the write or read service by NSV, the number of properties to be written is contained in the OPCSet and the number of properties to be read is contained in the OPCGet.

The minimum value of a processing target counter is 1 and the maximum value is limited by the message length by lower communications media in transmission and reception. In the case of SetGet_SNA, the value of the processing target property counter may be set to 0.

If, for instance, there are three requests as shown in Figure 23, the processing target property counter is 0x03.

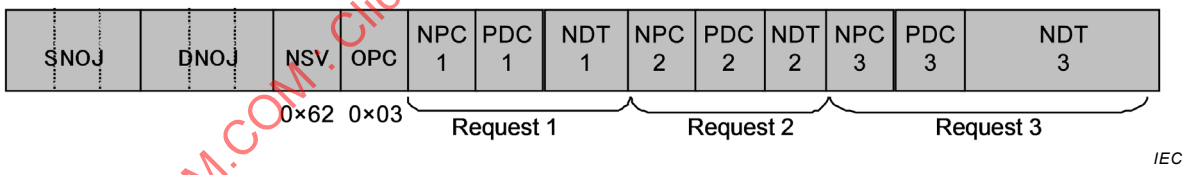
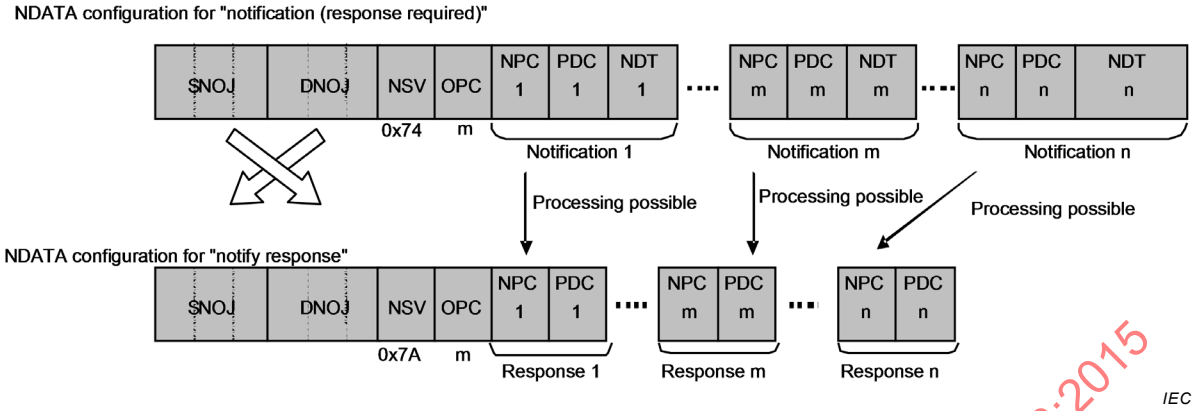


Figure 23 – Processing target property counter for three requests

6.8 NECD property (NPC)

This subclause provides detailed specifications for the NECD property (NPC) code shown in Figure 11. The NPC specifies a service target function and for each object stipulated by X1 (class group code) and X2 (class code) described in 6.5. (When a specified object changes, the target function also changes even when the code remains unchanged. However, the definition of NPC in this subclause is designed to ensure that, whenever possible, the same functions will have the same code.) These codes correspond to the object property identifiers in the object definitions as shown in Figure 24.



NOTE When b7 = 0, the other bits are defined differently.

Figure 24 – NPC detailed specifications

	8	9	A	B	C	D	E	F
0	**Region shared by all object classes**		**Region shared by each class group[b]**			**Region unique to each class[b]**		**User defined[a]**
1								
2								
3								
4								
5								
6								
7								
8								
9								
A								
B								
C								
D								
E								
F								

b7-b4 values (hex)

b3-b0 values (hex)

IEC

[a] Stipulated for each user. In the case of a user-defined object class, 0xA to 0xF in the four high-order bits (b7 to b4) are user-defined.

[b] As a rule these two regions are used, but in practice the boundary line will change for each class group. Individual regions will be specified in the object class detailed specifications in Clause 6 and "NECD Device Objects: Detailed specifications."

Figure 25 – NPC code allocation

6.9 Property data counter (PDC)

The property data counter retains the number of bytes in the NECD property value data (NDT). If, for instance, the NECD property value data sizes for requests 1, 2 and 3 are 2 B, 1 B and 5 B, respectively, the values placed in the first, second and third property data counters are 0x02, 0x01 and 0x05, respectively, as shown in Figure 25. In case of property value read request and property value notification request, the value of PDC shall be set to 0x00.

6.10 NECD property value data (NDT)

This subclause presents detailed code specifications for the NECD property value data (NDT) range shown in Figure 11. NDT consists of data for the relevant NECD property (NPC), such as status notification or specific setting and control by an NECD service (NSV). Detailed specifications are provided for the size, code value, etc. of the NDT for each NPC.

7 Basic sequences

7.1 General

Of the sequences exchanged between the NECD communications middleware for nodes connected to the NECD network, those that shall be implemented are called “basic sequences”. This chapter divides these basic sequences into two main categories for specification, as follows.

- Basic sequences for object control.
- Basic sequences for node start-up.

Depending on the type of device, some of the basic sequences specified in this clause, all of which are required, involve complex exchanges and thus entail much heavier communications processing than application processing. Therefore, the specifications were formulated to make the sequences as simple as possible.

7.2 Basic sequences for object control

7.2.1 Overview

NECD communications middleware exchanges are performed by stipulating the service (NSV: NECD service) with respect to the object property specified in Clause 6. Basic sequences for object control can be broadly divided into basic sequences for object control in general and basic sequences for service content. These two types are described below.

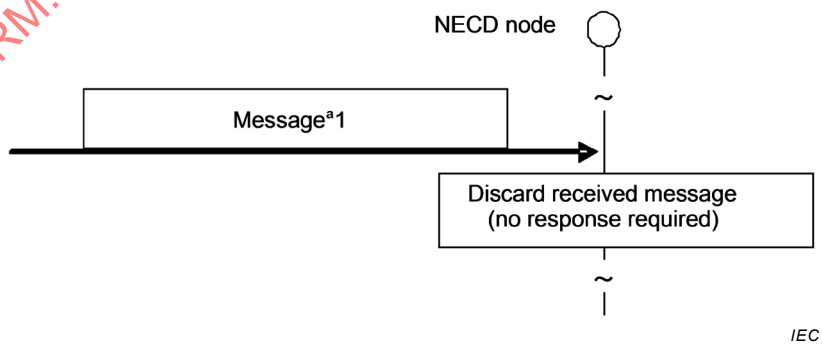
- Basic sequences for object control in general.
- Basic sequences for service content.

7.2.2 Basic sequences for object control in general

The NECD communications middleware performs the following four processes as basic processing when it receives a service (specified in Table 1 to Table 3) for an object property. The first three processes are described here. The fourth process “d)” is described in 7.2.3.

a) Processing when the controlled object does not exist

The NECD node receiving a message discards the received NECD message and returns no response, as shown in Figure 26.



[a] Message stipulating a DNOJ that does not exist in the destination NECD node.

Figure 26 – Basic sequence when controlled object does not exist

b) Processing when the controlled object exists but the controlled property does not exist, the control content cannot be interpreted, or only some properties of the controlled object can be processed.

The received NECD message is discarded, and the associated “response not possible” (NSV = 0x50–0x5F) is returned. Figure 27 shows the basic sequence when controlled objects exist but controlled property does not exist, and when control content cannot be interpreted, or only some properties of the controlled objects can be processed. This basic sequence is performed when a received request NSV = 0x6* (*: 0 to F).

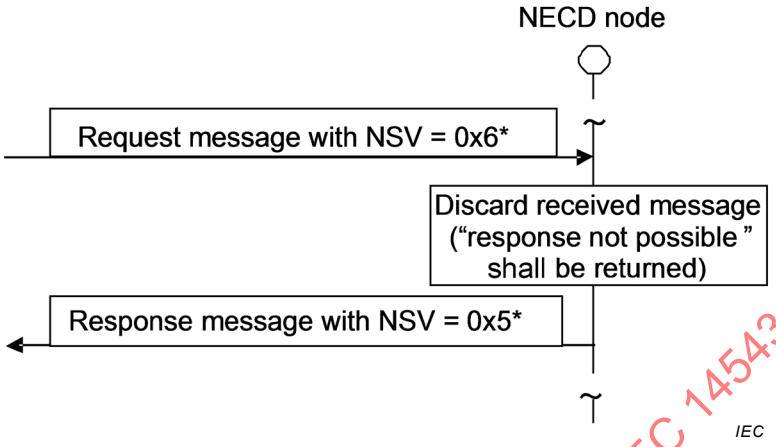


Figure 27 – Basic sequence when controlled objects exist

c) Processing when the controlled property exists but the stipulated service processing functions are not available

Processing is similar to that in b).

d) Processing when the controlled property exists and the stipulated service processing functions are available (described in 7.2.3).

7.2.3 Basic sequences for service content

The NECD communications middleware has three basic processing sequences for the reception of object property-related services (specified in Table 1, Table 2 and Table 3), assuming the stipulated property exists and has service functions.

a) Basic sequence for receiving a request (no response required)

There are some operations (NSV = 0x60–0x6E) that an NECD node performs in relation to properties. Figure 28 shows the basic sequence of the NECD node that is performed upon receipt of NSV = 0x60.

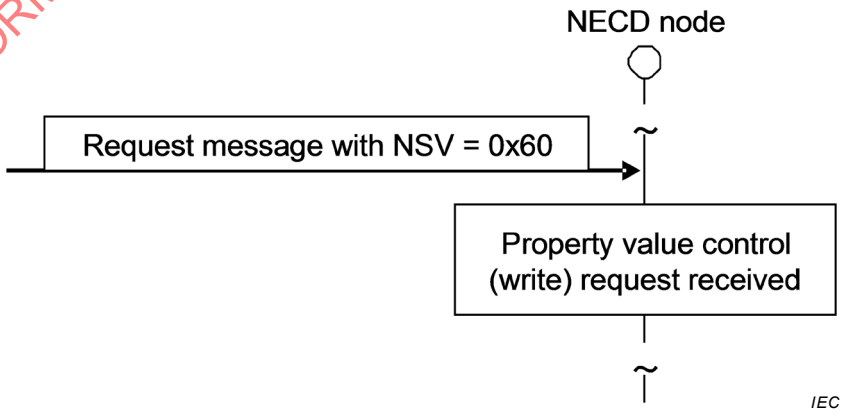


Figure 28 – Basic request receiving sequence for NSV = 0x60

b) Basic sequence for receiving a request (response required)