ISO/IEC 14543-3-1

Edition 1.0 2006-09

INTERNATIONAL STANDARD

Information technology – Home electronic system (HES) architecture – Part 3-1: Communication layers – Application layer for network based control of HES Class 1

ISO/IEC 14543-3-1:2006-09(en)

ISO/IEC 14543-3-1

Edition 1.0 2006-09

INTERNATIONAL STANDARD

Information technology – Home electronic system (HES) architecture – Part 3-1: Communication layers – Application layer for network based control of HES Class 1

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 35.240.67

ISBN 2-8318-8796-8

CONTENTS

INFORMATION TECHNOLOGY – HOME ELECTRONIC SYSTEM (HES) ARCHITECTURE –

Part 3-1: Communication layers – Application layer for network based control of HES Class 1

FOREWORD

1) ISO (International Organization for Standardization) and IEC (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. Their preparation is entrusted to technical committees; any ISO and IEC member body interested in the subject dealt with may participate in this preparatory work. International governmental and non-governmental organizations liaising with ISO and IEC also participate in this preparation.

2) In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

3) 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 and ISO member bodies.

4) IEC, ISO and ISO/IEC Publications have the form of recommendations for international use and are accepted by IEC 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.

5) In order to promote international uniformity, IEC 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 Publication and the corresponding national or regional publication should be clearly indicated in the latter.

6) ISO and IEC provide no marking procedure to indicate their approval and cannot be rendered responsible for any equipment declared to be in conformity with an ISO/IEC Publication.

7) All users should ensure that they have the latest edition of this publication.

8) 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 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.

9) 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.

10) Attention is drawn to the possibility that some of the elements of this International Standard 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 14543-3-1 was prepared by subcommittee 25: Interconnection of information technology equipment, of ISO/IEC joint technical committee 1: Information technology.

This International Standard together with ISO/IEC 14543-3-2 cancels and replaces ISO/IEC TR 14543-3, published in 2000. It constitutes a complete revision of the principles outlined in ISO/IEC TR 14543-3 and provides the specifications essential for an international standard.

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

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 title page.

INTRODUCTION

This part of ISO/IEC 14543 specifies the services and protocol of the application layer for usage in Home Electronic Systems. Some services are targeted to field level communication between devices. Other services are exclusively reserved for management purposes. Some services can be used for both management and run-time communication.

Currently, ISO/IEC 14543, *Information technology – Home Electronic System (HES) architecture*, consists of the following parts:

Part 2-1: *Introduction and device modularity*
Part 3-1: *Communication layers – Application layer for network based control of HES Class 1*
Part 3-2: *Communication layers – Transport, network and general parts of data link layer for network based control of HES Class 1*
Part 3-3: *User process for network based control of HES Class 1 (under consideration)*
Part 3-4: *System management – Management procedures for network based control of HES Class 1 (under consideration)*
Part 3-5: *Media and media dependent layers – Power line for network based control of HES Class 1 (under consideration)*
Part 3-6: *Media and media dependent layers – Twisted pair for network based control of HES Class 1 (under consideration)*
Part 3-7: *Media and media dependent layers – Radio frequency for network based control of HES Class 1 (under consideration)*
Part 4: *Home and building automation in a mixed-use building (technical report)*
Additional parts may be added later.

INFORMATION TECHNOLOGY – HOME ELECTRONIC SYSTEM (HES) ARCHITECTURE –

Part 3-1: Communication layers – Application layer for network based control of HES Class 1

1 Scope

This part of the ISO/IEC 14543 specifies the services and protocol of the application layer for usage in Home Electronic Systems. It provides the services and the interface to the user process as defined in ISO/IEC 14543-3-3 (EN 50090-3-2). This procedure is based on the services and the protocol as provided by the transport layer, the network layer and the data link layer as specified in ISO/IEC 14543-3-2.

2 Normative references

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

ISO/IEC 11801, *Information technology – Generic cabling for customer premises*

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

ISO/IEC 14543-3-2, *Information technology – Home electronic system (HES) – Part 3-2: Communication layers – Transport, network and general parts of data link layer for network based control of HES class 1*

EN 50090-3-2:2003, *Home and Building Electronic Systems (HBES) – Part 3-2: Aspects of application – User process for HBES Class 1*

NOTE 1 Reference to this standard will be replaced by reference to International Standard ISO/IEC 14543-3-3 which is currently under consideration. Please refer to bibliography.

EN 50090-7-1:2003, *Home and Building Electronic Systems (HBES) – Part 7-1: System Management – Management procedures*

NOTE 2 Reference to this standard will be replaced by reference to International Standard ISO/IEC 14543-3-4 which is currently under consideration. Please refer to bibliography.

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
application (in the sense of network application)
system, including its associated transmission method, which is supported by telecommunications cabling

[ISO/IEC 11801:2002, definition 3.1.2]

3.1.2
user application
software functionality, the control algorithm that runs in one single device

3.2 Abbreviations

AL	Application Layer
AD-converter	Analog-to-Digital-converter
APDU	Application layer Protocol Data Unit
APCI	Application layer Protocol Control Information
ASAP	Application layer Service Access Point
Acon	Application layer confirmation
con	confirmation
CPU	Central Processing Unit
HES Class 1	refers to simple control and command
HES Class 2	refers to Class 1 plus simple voice and stable picture transmission
HES Class 3	refers to Class 2 plus complex video transfers
ind	indication
Lcon	Local confirmation
PDU	Protocol Data Unit
Rcon	Remote confirmation
req	request
res	response
TL	Transport Layer
TPDU	Transport layer Protocol Data Unit
TSAP	Transport layer Service Access Point
USERMSG	User Message

4 Conformance

An entity of operational exchange conforming to this International Standard shall meet the requirements of 7.1, 7.2.1, 7.2.2, 7.3.5, 7.4.3, 7.4.4, 7.4.7 and clause 8.

All services shall be implemented according to the provisions of clauses 5 and 6.

5 Services of the application layer

5.1 Communication modes

The application layer shall provide a large variety of application services to the application process. Application processes in different devices interoperate by using services of application layer over communication modes. According to the transport layer, the following different types of communication modes shall exist:

a) point-to-multipoint, connectionless (multicast);

b) point-to-domain, connectionless (broadcast);

c) point-to-all-points, connectionless (system broadcast);

d) point-to-point, connectionless;

e) point-to-point, connection-oriented.

The application layer services that are offered shall depend on the communication mode. An application layer service shall not be applied on a communication mode for which it is not specified.

Some services may be used on the point-to-point connection-oriented, as well as the point-to-point connectionless communication mode, although application layer services shall always be mapped to transport layer services depending on the type of the communication mode.

5.2 Service primitives of the application layer

Each specified application layer service shall be invoked by the transport layer primitives request (req), indication (ind) and confirmation (con). For a remote confirmed service, the remote device shall use the same transport layer primitives to respond to the service.

The transport layer confirmation primitive shall only be a confirmation from the transport layer instance and shall include all data from the request plus the state which indicates whether the service was sent successfully or not. The application layer shall map the transport layer confirmation primitive to a local application layer confirmation (Lcon). See Figure 1 and Figure 2 for the interaction of the application layer.

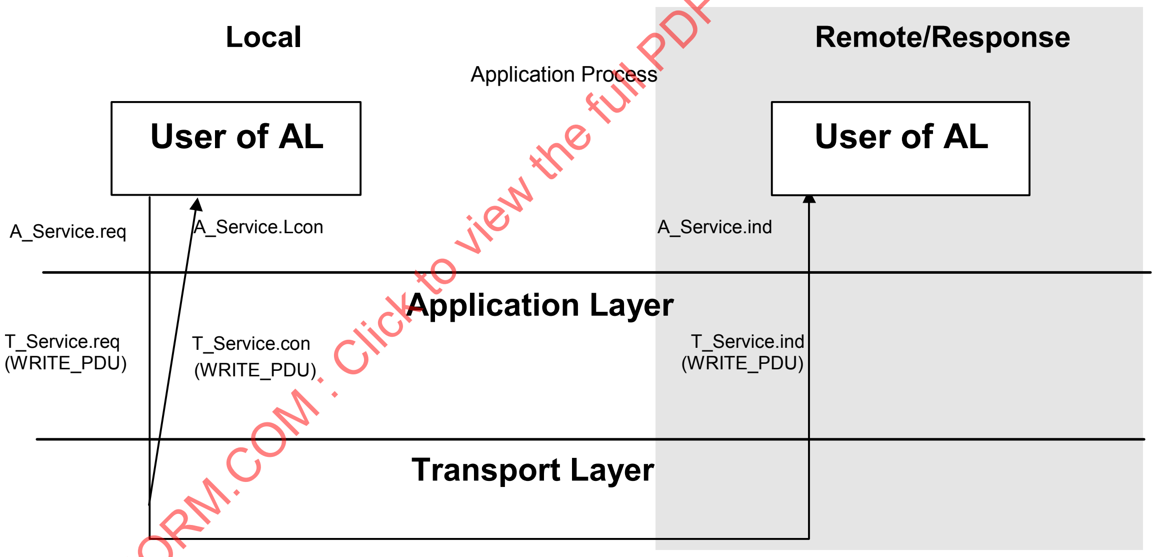


Figure 1 – Interaction of the application layer for services that are not remote confirmed

In case of a remote confirmed service the remote device shall initiate the response (res) primitive and the application layer shall map this service primitive to a transport layer request primitive. The local application layer shall receive the transport layer indication primitive and shall map it to an application layer confirmation (Acon). The transport layer confirmation in the remote device shall be mapped by the remote application layer to a remote confirmation (Rcon).

NOTE In the following service specifications the local application layer confirmation and the remote confirmation (Rcon) are not always described.

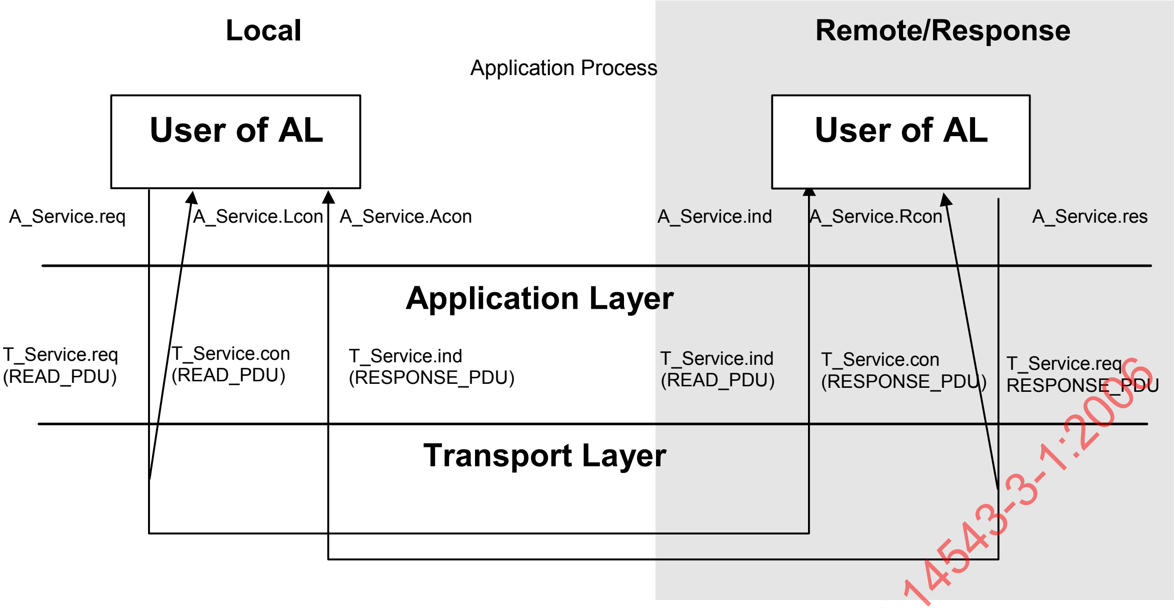


Figure 2 – Interaction of the application layer for services that are remote confirmed

6 Application layer protocol data unit (APDU)

An example of an APDU is shown in Figure 3.

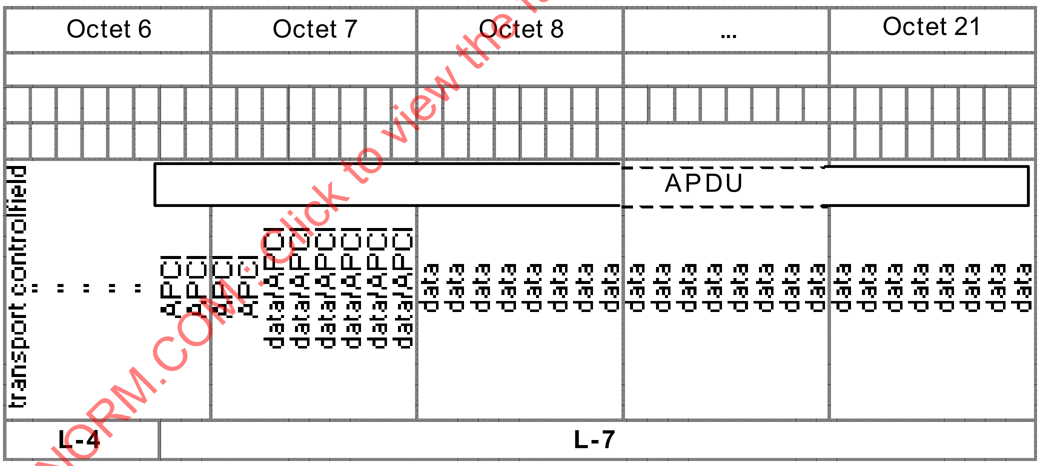


Figure 3 – APDU (Example)

Table 1 – APCI overview

APCI (bit position)																**Application layer Service**	**Allowed communication mode(s)**				
Octet n								Octet n+1									**Multicast**	**Broadcast**	**Point-to-all-points connectionless**	**Point-to-point connectionless**	**Point-to-point connection-oriented**
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1						
						APCI	APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI						
						0	0	0	0	0	0	0	0	0	0	A_GroupValue_Read	X				
						0	0	0	1							A_GroupValue_Response	X				
						0	0	1	0							A_GroupValue_Write	X				
						0	0	1	1	0	0	0	0	0	0	A_IndividualAddress_Write		X			
						0	1	0	0	0	0	0	0	0	0	A_IndividualAddress_Read		X			
						0	1	0	1	0	0	0	0	0	0	A_IndividualAddress_Response		X			
						1	1	1	1	0	1	1	1	0	0	A_IndividualAddressSerialNumber_Read		X			
						1	1	1	1	0	1	1	1	0	1	A_IndividualAddressSerialNumber_Response		X			
						1	1	1	1	0	1	1	1	1	0	A_IndividualAddressSerialNumber_Write		X			
						1	1	1	1	0	1	1	1	1	1	A_ServiceInformation_Indication_Write		X			
						1	1	1	1	0	1	1	0	1	0	A_NetworkParameter_Read		X			
						1	1	1	1	0	1	1	0	1	1	A_NetworkParameter_Response			X	X	
						1	1	1	1	1	0	0	1	0	0	A_NetworkParameter_Write		X		X	
						1	1	1	1	0	1	0	1	0	1	A_PropertyValue_Read				X	X
						1	1	1	1	0	1	0	1	1	0	A_PropertyValue_Response				X	X
						1	1	1	1	0	1	0	1	1	1	A_PropertyValue_Write				X	X
						1	1	1	1	0	1	1	0	0	0	A_PropertyDescription_Read				X	X
						1	1	1	1	0	1	1	0	0	1	A_PropertyDescription_Response				X	X
						1	1	1	1	1	0	0	1	0	1	A_Link_Read				X	X
						1	1	1	1	1	0	0	1	1	0	A_Link_Response				X	X
						1	1	1	1	1	0	0	1	1	1	A_Link_Write				X	X
						0	1	1	0							A_ADC_Read					X
						0	1	1	1							A_ADC_Response					X
						1	0	0	0	0	0					A_Memory_Read					X
						1	0	0	1	0	0					A_Memory_Response					X
						1	0	1	0	0	0					A_Memory_Write					X
						1	0	1	1	0	0	0	0	0	0	A_UserMemory_Read					X
						1	0	1	1	0	0	0	0	0	1	A_UserMemory_Response					X
						1	0	1	1	0	0	0	0	1	0	A_UserMemory_Write					X
						1	0	1	1	0	0	0	1	0	0	A_UserMemoryBit_Write (not for future use)					X
						1	0	1	1	0	0	0	1	0	1	A_UserManufacturerInfo_Read					X
						1	0	1	1	0	0	0	1	1	0	A_UserManufacturerInfo_Response					X
						1	0	1	1	0	0	0	1	1	1						X
														..	..	Reserved USERMSG					
						1	0	1	1	1	1	0	1	1	1						X
						1	0	1	1	1	1	1	0	0	0						X
														..	..	manufacturer specific area for USERMSG					
						1	0	1	1	1	1	1	1	1	0						X

Table 1 *(continued)*

APCI (bit position)																Application layer service	Allowed communication mode(s)				
Octet *n*								Octet *n*+1									**Multicast**	**Broadcast**	**Point-to-all-point connectionless**	**Point-to-point connectionless**	**Point-to-point connection-oriented**
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1						
						APCI	APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI	data/APCI						
						1	1	0	0	0	0	0	0	0	0	A_DeviceDescriptor_Read				X	
						1	1	0	1	0	0	0	0	0	0	A_DeviceDescriptor_Response				X	
						1	1	1	0	0	0	0	0	0	0	A_Restart				X	
Coupler specific services																					
						1	1	1	1	0	0	0	0	0	0	A_Open_Routing_Table_Req (not for future use)					X
						1	1	1	1	0	0	0	0	0	1	A_Read_Routing_Table_Req (not for future use)					X
						1	1	1	1	0	0	0	0	1	0	A_Read_Routing_Table__Res (not for future use)					X
						1	1	1	1	0	0	0	0	1	1	A_Write_Routing_Table_Req (not for future use)					X
						1	1	1	1	0	0	1	0	0	0	A_Read_Router_Memory_Res (not for future use)					X
						1	1	1	1	0	0	1	0	0	1	A_Read_Router_Memory_Res (not for future use)					X
						1	1	1	1	0	0	1	0	1	0	A_Write_Router_Memory_Req (not for future use)					X
						1	1	1	1	0	0	1	1	0	1	A_Read_Router_Status_Req (not for future use)					X
						1	1	1	1	0	0	1	1	1	0	A_Read_Router_Status_Res (not for future use)					X
						1	1	1	1	0	0	1	1	1	1	A_Write_Router_Status_Req (not for future use)					X
						1	1	1	1	0	1	0	0	0	0	A_MemoryBit_Write (not for future use)					X
						1	1	1	1	0	1	0	0	0	1	A_Authorize_Request					X
						1	1	1	1	0	1	0	0	1	0	A_Authorize_Response					X
						1	1	1	1	0	1	0	0	1	1	A_Key_Write					X
						1	1	1	1	0	1	0	1	0	0	A_Key_Response					X
Open Media Specific Services																					
						1	1	1	1	1	0	0	0	0	0	A_DomainAddress_Write		X			
						1	1	1	1	1	0	0	0	0	1	A_DomainAddress_Read		X			
						1	1	1	1	1	0	0	0	1	0	A_DomainAddress_Response		X			
						1	1	1	1	1	0	0	0	1	1	A_DomainAddressSelective_Read		X			

The APDU shall correspond to the transport layer protocol data unit (TPDU), but shall be reduced by the transport control field. The application control field shall be encoded and decoded by application layer and shall contain the application layer service codes (APCI). The application control field shall have a length of either 4 bit or 10 bit, as specified for each application layer service, in Clause 7.

The codes for the application control field are shown in Table 1. The complete protocol data unit (PDU) for each service primitive is shown in the description of every service.

Not defined and not supported application layer services shall be ignored by the application layer.

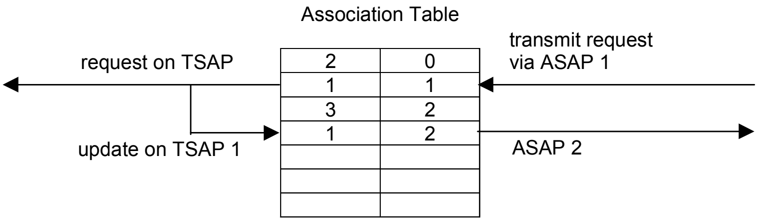
7 Application layer services

7.1 Application layer services on multicast communication mode

7.1.1 General

A multicast communication mode shall connect transport layer service access points (TSAP) to application layer service access points (ASAP). When one device sends an A_GroupValue-Service each device which is member of this group shall receive the A_GroupValue_Service.

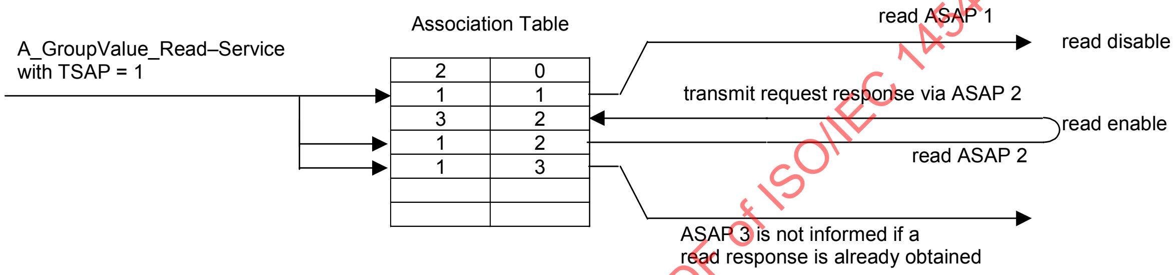
If the application layer of a device receives an A_GroupValue_Write-Service, it shall map the contained ASAP to exactly one TSAP; it shall search for other associations between ASAPs and the found TSAP informs all these associated ASAPs, as specified in 7.1.3, see Figure 4.



2	0
1	1
3	2
1	2

Figure 4 – Mapping the ASAP to the TSAP (example)

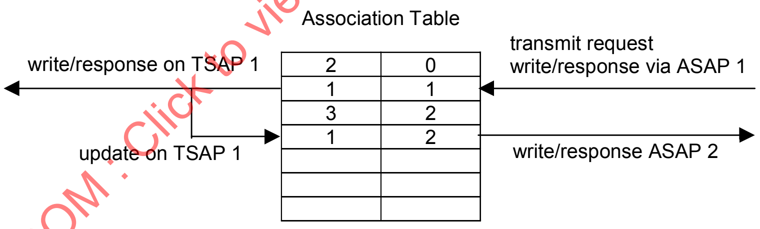
If the application layer of a device receives an A_GroupValue_Read-Service, it shall search for all ASAPs associated to this TSAP and shall inform all the associated ASAPs. Only one read response shall be generated by the user as specified in 7.1.2, see Figure 5.


Association Table
A_GroupValue_Read–Service
with TSAP = 1
read ASAP 1
read disable
transmit request response via ASAP 2
read enable
read ASAP 2
ASAP 3 is not informed if a
read response is already obtained

2	0
1	1
3	2
1	2
1	3

Figure 5 – Mapping a TSAP to an ASAP

If a transmission is requested (read response or write) via an ASAP, the application layer shall take the associated TSAP, update all the ASAPs with the same TSAP and generate an A_Group-Service-Request, see Figure 6.



2	0
1	1
3	2
1	2

Figure 6 – Handling requests and responses

7.1.2 A_GroupValue_Read Service

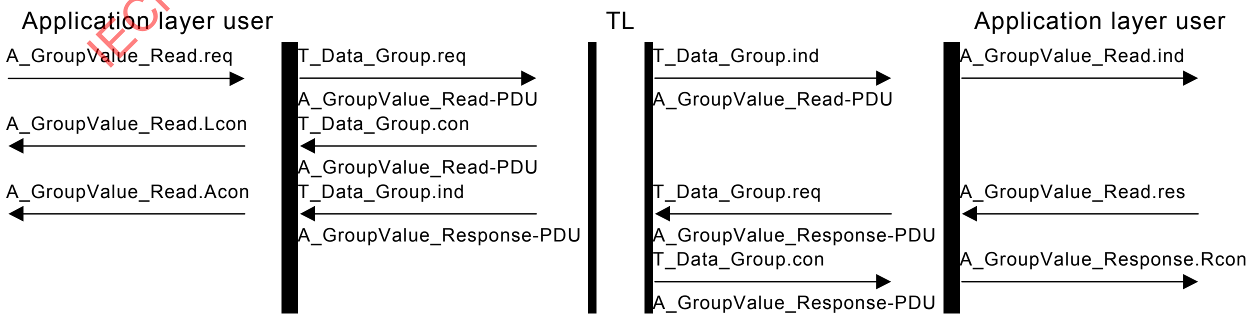


Figure 7 – Message flow for the A_Group_Value_Read service

The A_GroupValue_Read.req primitive shall be applied by the user of application layer, to receive an update of the value of its ASAP by making a communication partner respond with an

A_GroupValue_Read.res, i.e. the service shall be confirmed by the remote application process. The ASAP shall be associated to the TSAP, i.e. with a group address, as specified in ISO/IEC 14543-3-2. All other group members shall receive the A_GroupValue_Response-PDU as well, see Figure 7.

The local application layer shall accept the service request, map the ASAP to the TSAP and pass it with a T_Data_Group.req to the local transport layer. The parameters TSAP and priority shall be mapped to the corresponding parameters of the T_Data_Group.req primitive, the TSDU shall be an A_GroupValue_Read-PDU.

NOTE 1 During configuration the user of the HES system can decide about the mapping between ASAPs and TSAPs.

The remote application layer shall map a T_Data_Group.ind primitive with TSDU = A_GroupValue_Read-PDU to an A_GroupValue_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_GroupValue_Read.ind primitive. One A_GroupValue_Read.ind primitive shall be generated per ASAP that is assigned to the corresponding TSAP.

The remote application process shall evaluate the received A_GroupValue_Read-PDU and use the argument ASAP to obtain the response. It shall respond to the A_GroupValue_Read.ind primitive with an A_GroupValue_Read.res primitive containing the obtained response, see Figure 8.

NOTE 2 During configuration, the user of the HES system can decide whether or not the A_GroupValue_Read.res primitive is generated, although one ASAP should generate the A_GroupValue_Read.res primitive.

NOTE 3 It is left to the user application programmer to decide whether an A_GroupValue_Read.Acon time-out supervision is necessary.

Two different formats of the A_GroupValue_Response-PDU are used depending on the length of the value. The maximum length of the value shall be 14 octets. Unused data bits shall be set to zero, see Figure 9.

Octet 6								Octet 7							
						APCI									
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI						
						0	0	0	0	0	0	0	0	0	0

Figure 8 – A_GroupValue_Read-PDU (Example)

Octet 6								Octet 7								Octet 8..Octet 21							
						APCI										Value (up to 14 octets)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI														
						0	0	0	1	0	0	0	0	0	0	D	D	D	D	D	D	D	D

Figure 9 – A_GroupValue_Response-PDU (Example), length of ASAP data is more than 6 bit

Values that only consist of 6 bits or less shall have the following optimized A_GroupValue_Response-PDU format, see Figure 10.

Octet 6								Octet 7								
						APCI										
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
						APCI	APCI	APCI	APCI							
						0	0	0	1	D	D	D	D	D	D	Data

Figure 10 – A_GroupValue_Response-PDU (Example)
length of ASAP data is 6 bit or less

The remote application layer shall accept the service response, map the ASAP to the TSAP and pass it with a T_Data_Group.req to the local transport layer. The parameters ack_request, TSAP, hop_count_type and priority shall be mapped to the corresponding parameters of the T_Data_Group.req primitive, the TSDU shall be a A_GroupValue_Response-PDU.

The local application layer shall map a T_Data_Group.ind primitive with TSDU = A_GroupValue_Response-PDU to an A_GroupValue_Read.Acon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_GroupValue_Read. Acon primitive. More than one A_GroupValue_Read.Acon primitive may occur depending on the number of group members that have been configured to respond.

A_GroupValue_Read.req(ack_request, ASAP, priority, hop_count_type)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

ASAP: this parameter shall be used to contain the service access point

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

A_GroupValue_Read.Lcon(ack_request, ASAP, priority, hop_count_type, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

ASAP: this parameter shall be used to contain the service access point

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

priority: this parameter shall be used to indicate the priority that has been used to the transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

a_status: ok: this value of this parameter shall be used to indicate that the transmission of the A_GroupValue_Read.req has been successful

not_ok: this value of this parameter shall be used to indicate that the transmission of the A_GroupValue_Read.req did not succeed

A_GroupValue_Read.ind(ASAP, priority, hop_count_type)

ASAP: this parameter shall be used to contain the service access point

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

A_GroupValue_Read.res(ack_request, ASAP, priority, hop_count_type, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

ASAP: this parameter shall be used to contain the service access point

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

data: the parameter shall be used to contain the value of the associated service access point

A_GroupValue_Read.Rcon(ack_request, ASAP, priority, hop_count_type, data, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

ASAP: this parameter shall be used to contain the service access point

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

data: this parameter shall be used to contain the value of the associated service access point

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_GroupValue_Read.res has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_GroupValue_Read.res did not succeed

A_GroupValue_Read.Acon(ASAP, priority, hop_count_type, data)

ASAP: this parameter shall be used to contain the service access point

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

data: this parameter shall be used to contain the value of the associated service access point

7.1.3 A_GroupValue_Write Service

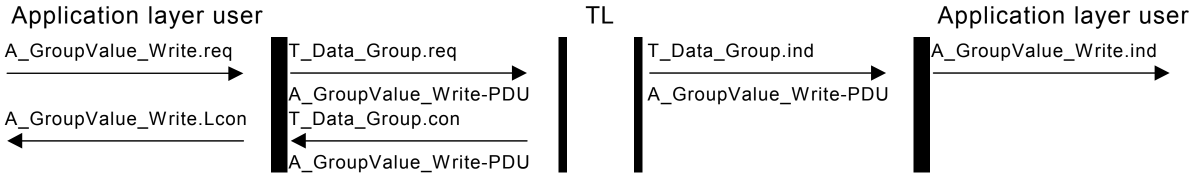


Figure 11 – Message flow for the A_Group_Value_Write service

The A_GroupValue_Write.req primitive shall be applied by the user of application layer, to send an update of its ASAP to all connected ASAPs, see Figure 11. The service shall not be confirmed by the remote application process, the confirmation shall be caused by the local T_Data_Group.con. The ASAP shall be associated to the TSAP i.e. with a group address, as specified in ISO/IEC 14543-3-2. All group members shall receive the A_GroupValue_Write-PDU.

The local application layer shall accept the service request, map the ASAP to the TSAP and pass it with a T_Data_Group.req to the local transport layer.

NOTE During configuration the user of the HES system can decide about this mapping between ASAPs and TSAPs.

The parameters TSAP and priority shall be mapped to the corresponding parameters of the T_Data_Group.req primitive, the TSDU shall be an A_GroupValue_Write-PDU.

The remote application layer shall map a T_Data_Group.ind primitive with TSDU = A_GroupValue_Write-PDU to an A_GroupValue_Write.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_GroupValue_Write.ind primitive. One A_GroupValue_Write.ind primitive shall be generated per ASAP that is assigned to the corresponding TSAP (i.e. group_address).

Two different formats of the A_GroupValue_Write-PDU are used depending on the length of the value. The maximum length of the value shall be 14 octets. Unused data bits shall be set to zero see Figure 12.

Octet 6								Octet 7								Octet 8.to.Octet 21							
						APCI										Value (up to 14 octets)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI														
						0	0	1	0	0	0	0	0	0	0	D	D	D	D	D	D	D	D

Figure 12 – A_GroupValue_Write-PDU (Example), length of ASAP data is more than 6 bit

Values that only consist of 6 bits or less shall have the following optimized A_GroupValue_Write-PDU format, see Figure 13.

Octet 6								Octet 7							
						APCI									
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI						
						0	0	1	0	d	d	d	d	d	d

Figure 13 – A_GroupValue_Write-PDU (Example), length of ASAP data is 6 bit or less

If the local application layer receives a T_Data_Group.con from the local transport layer, it shall pass an A_GroupValue_Write.Lcon primitive to the local application process. If the confirmation is positive (t_status = ok), the local application layer shall pass a positive A_GroupValue_Write.Lcon (a_status = ok) to the local application process. If the confirmation is negative (t_status = not_ok), the local application layer shall pass an A_GroupValue_Write.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated T_Data_Group.req did not succeed.

A_GroupValue_Write.req(ack_request, ASAP, priority, hop_count_type, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

ASAP: this parameter shall be used to contain the service access point

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

data: this parameter shall be used to contain the data of the associated application layer service access point

A_GroupValue_Write.Lcon(ack_request, ASAP, priority, hop_count_type, data, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

ASAP: this parameter shall be used to contain the service access point

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

data: this parameter shall be used to contain the data of the associated application layer service access point

a_status: ok: the value of this parameters shall be used to indicate that the transmission of the A_GroupValue_Write.req has been successful

not_ok: the value of this parameters shall be used to indicate that the transmission of the A_GroupValue_Write.req did not succeed

A_GroupValue_Write.ind(ASAP, priority, hop_count_type, data)

ASAP: this parameter shall be used to contain the service access point

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

data: this parameter shall be used to contain the data of the associated application layer service access point

7.2 Application layer services on broadcast communication mode

7.2.1 A_IndividualAddress_Write Service

The A_IndividualAddress_Write.req primitive shall be applied by the user of application layer to modify the individual address in a communication partner. The communication partner shall not be identified in the service, i.e. the destination shall be defined by selecting a destination manually. This may be done by pressing a button on exactly one device that brings this device into a ´programming mode´, i.e. only the device where the button is pressed shall accept the A_IndividualAddress_Write.ind, others shall ignore it. The way that a product is set to ´programming mode´ may be manufacturer specific.

The local application layer shall accept the service request and pass it with a T_Data_Broadcast.req to the local transport layer. The parameter priority, implicitly with value 'system', shall be mapped to the corresponding parameter of the T_Data_Broadcast.req primitive, the TSDU shall be an A_IndividualAddress_Write-PDU, see Figure 14.

Octet 6								Octet 7								Octet 8								Octet 9							
						APCI										New address (high)								New address (low)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI																						
						0	0	1	1	0	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	X

Figure 14 – A_IndividualAddress_Write-PDU (Example)

The remote application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_IndividualAddress_Write-PDU to an A_IndividualAddress_Write.ind primitive. The argument priority, implicitly with value 'system', shall be mapped to the corresponding argument priority of the A_IndividualAddress_Write.ind primitive.

The application process shall ignore the A_IndividualAddress_Write.ind primitive if the device is not in ´programming mode´. Otherwise the local individual address shall be set to the new address.

If the local application layer receives a T_Data_Broadcast.con from the local transport layer, it shall pass an A_IndividualAddress_Write.Lcon primitive to the local application process. If the confirmation is positive (t_status = ok), the local application layer shall pass a positive A_IndividualAddress_Write.Lcon(a_status = ok) to the local application process. If the confirmation is negative (t_status = not_ok), the local application layer shall pass an A_IndividualAddress_Write.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated T_Data_Broadcast.req did not succeed.

A_IndividualAddress_Write.req(ack_request, priority, hop_count_type, newaddress)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system" hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

newaddress: this parameter shall be used to contain the new value of the individual address

A_IndividualAddress_Write.Lcon(ack_request, priority, hop_count_type, newaddress, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to the transmit the requested frame; it shall be “system”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

newaddress: this parameter shall be used to contain the new value of the individual address

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddress_Write.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddress_Write.req did not succeed

A_IndividualAddress_Write.ind(priority, hop_count_type, newaddress)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

newaddress: this value shall be used to contain the new value of the individual address

7.2.2 A_IndividualAddress_Read-Service

The A_IndividualAddress_Read.req primitive shall be applied by the user of application layer to read the individual address in a communication partner. The communication partner shall not be identified in the service, i.e. the destination shall be defined by selecting a destination manually. This can be done by pressing a button on one or more devices that brings these devices into a ´programming mode´, i.e. only a device where the button is pressed shall accept the A_IndividualAddress_Read.ind, others shall ignore it. The way that a product is set to ´programming mode´ may be manufacturer specific.

The local application layer shall accept the service request and pass it with a T_Data_Broadcast.req to the local transport layer. The parameter priority, implicitly with value ‘system’, shall be mapped to the corresponding parameter of the T_Data_Broadcast.req primitive; the TSDU shall be an A_IndividualAddress_Read-PDU, see Figure 15.

The remote application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_IndividualAddress_Read-PDU to an A_IndividualAddress_Read.ind primitive. The argument priority, implicitly with value ‘system’, shall be mapped to the corresponding argument priority of the A_IndividualAddress_Read.ind primitive.

The remote application process shall respond to the A_IndividualAddress_Read.ind primitive with an A_IndividualAddress_Read.res primitive only if the device is in ´programming mode'.

Octet 6								Octet 7							
						APCI									
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI						
						0	1	0	0	0	0	0	0	0	0

Figure 15 – A_IndividualAddress_Read-PDU (Example)

Octet 6								Octet 7							
						APCI									
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI						
						0	1	0	1	0	0	0	0	0	0

Figure 16 – A_IndividualAddress_Response-PDU (Example)

The remote application layer shall accept the service response and pass it with a T_Data_Broadcast.req to the transport layer; the TSDU shall be an A_IndividualAddress_Response-PDU, see Figure 16. The local application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_IndividualAddress_Response-PDU to an A_IndividualAddress_Read.Acon primitive. The argument priority, implicitly with value 'system', shall be mapped to the corresponding argument priority of the A_IndividualAddress_Read.Acon primitive. The argument source_address shall be mapped to the corresponding argument individual_address of the A_IndividualAddress_Read.Acon primitive.

A_IndividualAddress_Read.req(ack_request, priority, hop_count_type)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

A_IndividualAddress_Read.Lcon(ack_request, priority, hop_count_type, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddress_Read.req has been successful

not_ok: this value of this parameter shall be used to indicate that the transmission of the A_IndividualAddress_Read.req did not succeed

A_IndividualAddress_Read.ind(priority, hop_count_type)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

A_IndividualAddress_Read.res(ack_request, priority, hop_count_type, individual address)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

individual_address: this parameter shall be used to contain the individual address of the device

A_IndividualAddress_Read.Rcon(ack_request, priority, hop_count_type, individual_address, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to the transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

individual_address: this parameter shall be used to contain the individual address of this device

a_status: ok: this value of this parameter shall be used to indicate that the transmission of the A_IndividualAddress_Read.res has been successful

not_ok: this value of this parameter shall be used to indicate that the transmission of the A_IndividualAddress_Read.res did not succeed

A_IndividualAddress_Read.Acon(priority, hop_count_type, individual address)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

individual_address: this parameter shall be used to contain the individual address of the device that has sent the received frame

7.2.3 A_IndividualAddressSerialNumber_Read-Service

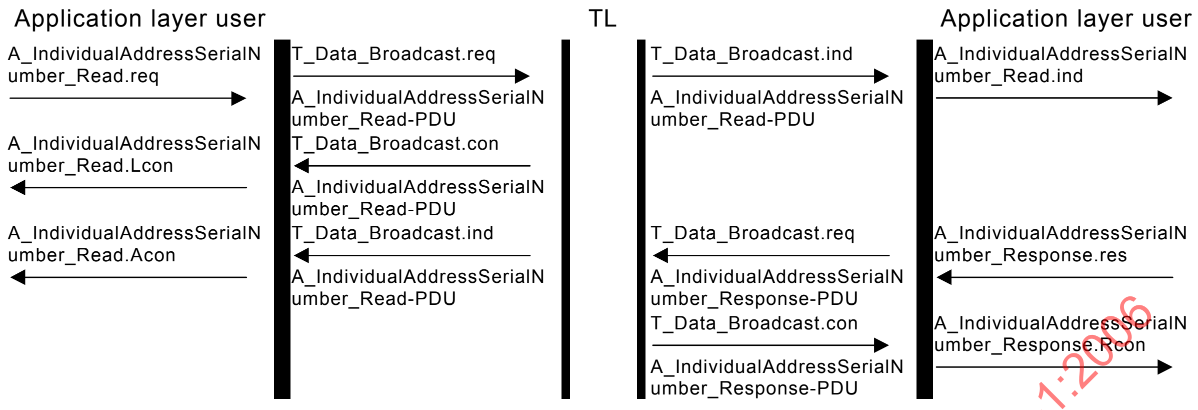

Application layer user
TL
Application layer user
A_IndividualAddressSerialNumber_Read.req
T_Data_Broadcast.req
A_IndividualAddressSerialNumber_Read-PDU
T_Data_Broadcast.ind
A_IndividualAddressSerialNumber_Read-PDU
A_IndividualAddressSerialNumber_Read.ind
A_IndividualAddressSerialNumber_Read.Lcon
T_Data_Broadcast.con
A_IndividualAddressSerialNumber_Read-PDU
A_IndividualAddressSerialNumber_Read.Acon
T_Data_Broadcast.ind
A_IndividualAddressSerialNumber_Read-PDU
T_Data_Broadcast.req
A_IndividualAddressSerialNumber_Response-PDU
A_IndividualAddressSerialNumber_Response.res
T_Data_Broadcast.con
A_IndividualAddressSerialNumber_Response-PDU
A_IndividualAddressSerialNumber_Response.Rcon

Figure 17 – Message flow for the A_IndividualAddressSerialNumber_Read service120

The A_IndividualAddressSerialNumber_Read.req primitive shall be applied by the user of application layer to read the individual address in a communication partner. The communication partner shall be identified using the unique serial number (6 octets) of the device, see Figure 17.

The local application layer shall accept the service request and pass it with a T_Data_Broadcast.req to the local transport layer. The parameter priority, implicitly with value 'system', shall be mapped to the corresponding parameter of the T_Data_Broadcast.req primitive; the TSDU shall be an A_IndividualAddressSerialNumber_Read-PDU, see Figure 18.

If the local application layer receives a T_Data_Broadcast.con from the local transport layer it shall pass an A_IndividualAddressSerialNumber_Read.Lcon to the local user. If the confirmation is positive (t_status = ok), the local application layer shall pass a positive A_IndividualAddressSerialNumber_Read.Lcon (a_status = ok) to the local user. If the confirmation is negative (a_status = not_ok), the local application layer shall pass an A_IndividualAddressSerialNumber_Read.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated A_IndividualAddressSerialNumber_Read.req did not succeed.

The remote application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_IndividualAddressSerialNumber_Read-PDU to an A_IndividualAddressSerial-Number_Read.ind primitive. The argument priority, implicitly with value 'system', shall be mapped to the corresponding argument priority of the A_IndividualAddress-SerialNumber_Read.ind primitive.

Octet 6								Octet 7								Octet 8 to octet 13							
																Serial number (6 octets)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI								
						1	1	1	1	0	1	1	1	0	0								

Figure 18 – A_IndividualAddressSerialNumber_Read-PDU (example)

The remote application process shall respond to the A_IndividualAddress-SerialNumber_Read.ind primitive with an A_IndividualAddressSerialNumber_Read.res primitive, if the serial number received is equal to the serial number of the device.

The remote application layer shall accept the service response and pass it with a T_Data_Broadcast.req to the remote transport layer; the TSDU shall be an A_IndividualAddressSerialNumber_Response-PDU, seeFigure 19.

Octet 6								Octet 7								Octet 8 to Octet 13								Octet 14								Octet 15								Octet 16								Octet 17							
																Serial number (6 octets)								Domain Address																Reserved															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																								
						1	1	1	1	0	1	1	1	0	1																																								

Figure 19 – A_IndividualAddressSerialNumber_Response-PDU (example)

If the remote application layer receives a T_Data_Broadcast.con from the remote transport layer, it shall pass an A_IndividualAddressSerialNumber_Read.Rcon to the remote user. If the confirmation is positive (t_status = ok), the remote application layer shall pass a positive A_IndividualAddressSerialNumber_Read.Rcon (a_status = ok) to the remote user. If the confirmation is negative (a_status = not_ok), the remote application layer shall pass an A_IndividualAddressSerialNumber_Read.Rcon (a_status = not_ok) to the remote user indicating that the transmission of the associated A_IndividualAddressSerialNumber_Read.res did not succeed.

The local application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_IndividualAddressSerialNumber_Response-PDU to an A_IndividualAddress-SerialNumber_Read.Acon primitive. The argument priority, implicitly with value 'system', shall be mapped to the corresponding argument priority of the A_IndividualAddressSerialNumber_Read.Acon primitive. The argument source_address shall be mapped to the argument individual_address.

A_IndividualAddressSerialNumber_Read.req (ack_request, priority, hop_count_type, serial_number)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

serial_number: this parameter shall contain the serial number

A_IndividualAddressSerialNumber_Read.Lcon (ack_request, priority, hop_count_type, serial_number, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

serial_number: this parameter shall contain the serial number a_status ok: this value of this parameter shall be used to indicate that the transmission of the A_IndividualAddressSerialNumber_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddressSerialNumber_Read.req did not succeed

A_IndividualAddressSerialNumber_Read.ind (priority, hop_count_type, serial_number)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

serial-number: this parameter shall contain the serial number

A_IndividualAddressSerialNumber_Read.res (ack_request, priority, hop_count_type, serial_number, domain_address)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

serial_number: this parameter shall contain the serial number

domain_address: this parameter shall contain the domain address of the remote device

A_IndividualAddressSerialNumber_Read.Rcon (ack_request, priority, hop_count_type, serial_number, domain_address, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to the transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

serial_number: this parameter shall contain the serial number

domain_address: this parameter shall contain the domain address of the remote device

a_status ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddressSerialNumber_Read.res has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddressSerialNumber_Read.res did not succeed

A_IndividualAddressSerialNumber_Read.Acon (priority, hop_count_type, serial_number, individual_address, domain_address)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

serial_number: this parameter shall contain the serial number

individual_address: this parameter shall contain the individual address of the remote device

domain_address: this parameter shall contain the domain address of the remote device

7.2.4 A_IndividualAddressSerialNumber_Write Service

The A_IndividualAddressSerialNumber_Write.req primitive shall be applied by the user of application layer, to modify the individual address in a communication partner. The communication partner shall be identified using the unique serial number (6 octets) of the device.

The local application layer shall accept the service request and pass it with a T_Data_Broadcast.req to the local transport layer. The parameter priority, implicitly with value 'system', shall be mapped to the corresponding parameter of the T_Data_Broadcast.req primitive; the TSDU shall be an A_IndividualAddressSerialNumber_Write-PDU, see Figure 20.

Octet 6								Octet 7								Octet 8 to Octet 13								Octet 14								Octet 15								Octet 16 to Octet 19							
						APCI										Serial number (6 octets)								New address (high)								New address (low)								Reserved (4 octets)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																
						1	1	1	1	0	1	1	1	1	0																																

Figure 20 – A_IndividualAddressSerialNumber_Write-PDU (Example)

The remote application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_IndividualAddressSerialNumber_Write-PDU to an A_IndividualAddressSerialNumber_Write.ind primitive. The argument priority shall be mapped to the corresponding argument priority of the A_IndividualAddressSerialNumber_Write.ind primitive.

If the local application layer receives a T_Data_Broadcast.con from the local transport layer, it shall pass an A_IndividualAddressSerialNumber_Write.Lcon primitive to the local application process. If the confirmation is positive (t_status = ok), the local application layer shall pass a positive A_IndividualAddressSerialNumber_Write.Lcon(a_status = ok) to the local application process. If the confirmation is negative (t_status = not_ok), the local application layer shall pass an A_IndividualAddressSerialNumber_Write.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated T_Data_Broadcast.req did not succeed.

A_IndividualAddressSerialNumber_Write.req(ack_request, priority, hop_count_type, serial_number, newaddress)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

serial_number: this parameter shall contain the serial number

newaddress: this parameter shall contain the new value of the individual address for the remote device

A_IndividualAddressSerialNumber_Write.Lcon(ack_request, priority, hop_count_type, new_address, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

newaddress: this parameter shall contain the new value of the individual address for the remote device

a_status: ok: this value of this parameter shall be used to indicate that the transmission of the A_IndividualAddressSerialNumber_Write.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_IndividualAddressSerialNumber_Write.req did not succeed

A_IndividualAddressSerialNumber_Write.ind(priority, hop_count_type, serial_number, new-address)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

serial_number: this parameter shall contain the serial number

newaddress: this parameter shall contain the new value of the individual address

7.2.5 A_ServiceInformation_Indication_Write Service

The A_ServiceInformation_Indication_Write.req primitive shall be applied by the user of application layer, to inform communication partners about the status of the user application (running/stopped), duplicate individual address and verify mode. These elements are specified in ISO/IEC 14543-3-3 (EN 50090-3-2).

The local application layer shall accept the service request and pass it with a T_Data_Broadcast.req to the local transport layer. The parameter priority, implicitly with value ‘system’, shall be mapped to the corresponding parameter of the T_Data_Broadcast.req primitive, the TSDU shall be an A_ServiceInformation_Indication_Write-PDU, see Figure 21.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10							
						APCI										Info																							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	reserved	reserved	reserved	reserved	reserved	verify mode	dupl. phys. Addr	appl. stopped	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved
						1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	x	x	x	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 21 – A_ServiceInformation_Indication_Write-PDU (example)

The remote application layer shall map a T_Data_Broadcast.ind primitive with TSDU = A_ServiceInformation_Indication_Write-PDU to an A_ServiceInformation_Indication_Write.ind primitive. The argument priority shall be mapped to the corresponding argument priority of the A_ServiceInformation_Indication_Write.ind primitive.

If the local application layer receives a T_Data_Broadcast.con from the local transport layer, it shall pass an A_ServiceInformation_Indication_Write.Lcon primitive to the local application process. If the confirmation is positive (t_status = ok), the local application layer shall pass a positive A_ServiceInformation_Indication_Write.Lcon(a_status = ok) to the local application process. If the confirmation is negative (t_status = not_ok), the local application layer shall pass an A_ServiceInformation_Indication_Write.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated T_Data_Broadcast.req did not succeed.

A_ServiceInformation_Indication_Write.req(ack_request, priority, hop_count_type, info)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

info: this parameter shall contain the service information

A_ServiceInformation_Indication_Write.Lcon(ack_request, priority, hop_count_type, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to the transmit the requested frame; it shall be “system”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

info: this parameter shall contain the service information

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_ServiceInformation_Indication_Write.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_ServiceInformation_Indication_Write.req did not succeed

A_ServiceInformation_Indication_Write.ind(priority, hop_count_type, info)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

info: this parameter shall contain the service information

7.2.6 A_DomainAddress_Write service

The A_DomainAddress_Write.req primitive shall be applied by the user of application layer, to modify the Domain Address in a communication partner. The communication partner shall not be identified in the service, i.e. the destination shall be identified by selecting a destination manually. This can be done by pressing a button on exactly one device that shall bring this device into a ´programming mode´, i.e. only the device where the button is pressed shall accept the A_DomainAddress_Write.ind, others shall ignore it. The way that a product is set to ´programming mode´ may be manufacturer specific.

The local application layer shall accept the service request and pass it with a T_Data_SystemBroadcast.req to the local transport layer. The parameter priority shall be mapped to the corresponding parameter of the T_Data_SystemBroadcast.req primitive, the TSDU shall be an A_DomainAddress_Write-PDU, see Figure 22.

Octet 6								Octet 7								Octet 8								Octet 9							
						APCI										Domain address (high)								Domain address (low)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																
						1	1	1	1	1	0	0	0	0	0																

Figure 22 – A_DomainAddress_Write-PDU

The remote application layer shall map a T_Data_SystemBroadcast.ind primitive with TSDU = A_DomainAddress_Write-PDU to an A_DomainAddress_Write.ind primitive. The arguments priority and domain_address_new shall be mapped to the corresponding arguments of the A_DomainAddress_Write.ind primitive.

The application process shall ignore the A_DomainAddress_Write.ind primitive if the device is not in ´programming mode´.

If the local application layer receives a T_Data_SystemBroadcast.con from the local transport layer it shall pass an A_DomainAddress_Write.Lcon primitive to the local application process. If the confirmation is positive (t_status = ok), the local application layer shall pass a positive A_DomainAddress_Write.Lcon(a_status = ok) to the local application process. If the confirmation is negative (t_status = not_ok), the local application layer shall pass an A_DomainAddress_Write.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated T_Data_SystemBroadcast.req did not succeed.

A_DomainAddress_Write.req(ack_request, priority, hop_count_type, domain_address_new)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

domain_address_new:this parameter shall contain the new value of the Domain Address

A_DomainAddress_Write.Lcon(ack_request, priority, hop_count_type, domain_address_new, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

domain_address_new:this parameter shall contain the new value of the Domain Address a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_DomainAddress_Write.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_DomainAddress_Write.req did not succeed

A_DomainAddress_Write.ind(priority, hop_count_type, domain_address_new)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

domain_address_new:this parameter shall contain the new value of the Domain Address

7.2.7 A_DomainAddress_Read Service

The A_DomainAddress_Read.req primitive shall be applied by the user of application layer, to read the Domain Address from a communication partner. The communication partner shall not be identified in the service, i.e. the destination shall be defined by selecting a destination manually. This can be done by pressing a button on one or more devices that shall bring these devices into a ´programming mode´, i.e. only a device where the button is pressed shall accept the A_DomainAddress_Read.ind, others shall ignore it. The way that a product is set to ´programming mode´ may be manufacturer specific.

The local application layer shall accept the service request and pass it with a T_Data_SystemBroadcast.req to the local transport layer. The parameter priority shall be mapped to the corresponding parameter of the T_Data_SystemBroadcast.req primitive, the TSDU shall be an A_DomainAddress_Read-PDU, see Figure 23.

The remote application layer shall map a T_Data_SystemBroadcast.ind primitive with TSDU = A_DomainAddress_Read-PDU to an A_DomainAddress_Read.ind primitive. The argument priority shall be mapped to the corresponding argument priority of the A_DomainAddress_Read.ind primitive.

The remote application process shall respond to the A_DomainAddress_Read.ind primitive with an A_DomainAddress_Read.res primitive only if the device is in ´programming mode´.

Octet 6								Octet 7							
						APCI									
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI
						1	1	1	1	1	0	0	0	0	1

Figure 23 – A_DomainAddress_Read-PDU (example)

Octet 6								Octet 7								Octet 8								Octet 9							
						APCI										Domain address (high)								Domain address (low)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																
						1	1	1	1	1	0	0	0	1	0																

Figure 24 – A_DomainAddress_Response-PDU (example)

The remote application layer shall accept the service response and pass it with a T_Data_SystemBroadcast.req to the remote transport layer, the TSDU shall be an A_DomainAddress_Response-PDU, see Figure 24.

The local application layer shall map a T_Data_SystemBroadcast.ind primitive with TSDU = A_DomainAddress_Response-PDU to an A_DomainAddress_Read.Acon primitive. The argument priority shall be mapped to the corresponding argument priority of the A_DomainAddress_Read.Acon primitive.

A_DomainAddress_Read.req(ack_request, priority, hop_count_type)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

A_DomainAddress_Read.Lcon(ack_request, priority, hop_count_type, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_DomainAddress_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_DomainAddress_Read.req did not succeed

A_DomainAddress_Read.ind(priority, hop_count_type)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

A_DomainAddress_Read.res(ack_request, priority, hop_count_type, domain_address)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

domain_address: this parameter shall contain the value of the domain address

A_DomainAddress_Read.Acon(priority, hop_count_type, domain_address)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

domain_address: this parameter shall contain the value of the domain address

7.2.8 A_DomainAddressSelective_Read Service

The A_DomainAddressSelective_Read.req primitive shall be applied by the user of application layer, to read the domain address from the communication partner that is identified within the service. This service is particularly used to check the existence of any open media devices with the specified domain address in possibly neighbouring installations.

The local application layer shall accept the service request and pass it with a T_Data_SystemBroadcast.req to the local transport layer. The parameter priority shall be mapped to the corresponding parameter of the T_Data_SystemBroadcast.req primitive, the TSDU shall be an A_DomainAddressSelective_Read-PDU, see Figure 25.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11								Octet 12							
						APCI										Domain address (high)								Domain address (low)								Start address (high)								Start address (low)								Range							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																								
						1	1	1	1	1	0	0	0	1	1																																								

Figure 25 – A_DomainAddressSelective_Read-PDU (example)

The remote application layer shall map a T_Data_SystemBroadcast.ind primitive with TSDU = A_DomainAddressSelective_Read-PDU to an A_DomainAddressSelective_Read.ind primitive. The arguments priority, domain_address, start_address and range shall be mapped to the corresponding arguments of A_DomainAddressSelective_Read.ind primitive.

The remote application process shall ignore the A_DomainAddressSelective_Read.ind primitive, if the domain address of the remote device does not match the argument domain_address, or the individual address of the remote device is lower than the argument start_address or the individual address of the remote device is higher than the (start_address + range).

If the remote application process accepts the A_DomainAddressSelective_Read.ind primitive it shall respond with an A_DomainAddress_Read.res primitive after a wait time: (individual_address - start_address) Tmedia. If the received argument range was lower than 0xFF and application process receives during the waiting time an A_DomainAddress_Read.res, the transmission of the response shall be terminated.

A_DomainAddressSelective_Read.req(ack_request, priority, hop_count_type, domain_address, start_address, range)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

domain_address: this parameter shall contain the domain address to be scanned start_address: this parameter shall contain the start individual address

range: this parameter shall be used to indicate the range of individual addresses involved in the scan process (scan from start_address to start_address+range)

A_DomainAddressSelective_Read.Lcon(ack_request, priority, hop_count_type, domain_address, start_address, range, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

domain_address: this parameter shall contain the domain address to be scanned

start_address: this parameter shall contain the start individual address

range this parameter shall be used to indicate the range of individual addresses involved in the scan process (scan from start_address to start_address+range)

a_status: ok: this value of this parameter shall be used to indicate that the transmission of the A_DomainAddressSelective_Read.req has been successful

not_ok: this value of this parameter shall be used to indicate that the transmission of the A_DomainAddressSelective_Read.req has been successful did not succeed

A_DomainAddressSelective_Read.ind(priority, hop_count_type, domain_address, start_address, range)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

domain_address: this parameter shall contain the domain address to be scanned

start_address: this parameter shall contain the start individual address

range: this parameter shall be used to indicate the range of individual addresses involved in the scan process (scan from start_address to start_address+range)

7.2.9 A_NetworkParameter_Read service

The A_NetworkParameter_Read.req primitive shall be applied by the user of Application Layer of the management client to check about the configuration of a network parameter. The service shall be broadcast to all devices in the network. A device shall respond to the service if it complies to the conditions specified in the service parameters.

The local Application Layer shall accept the service request and pass it with a T_Data_Broadcast.req to the local Transport Layer. The parameter priority, implicitly with value 'system', shall be mapped to the corresponding parameter of the T_Data_Broadcast.req primitive, the TSDU shall be an A_NetworkParameter_Read-PDU, see Figure 26.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11 to N							
						APCI										Parameter_type																								Test_info							
																Object_type																PID															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																
						1	1	1	1	0	1	1	0	1	0																																

Figure 26 – A_NetworkParameter_Read-PDU (example)

The remote Application Layer shall map a T_Data_Broadcast.ind primitive with TSDU= A_NetworkParameter_Read-PDU to an A_NetworkParameter_Read.ind primitive. The argument priority, implicitly with value 'system', shall be mapped to the corresponding argument priority of the A_NetworkParameter_Read.ind primitive.

The remote Application Layer shall accept the service response. If the parameter comm_mode equals "point_to_point, connectionless", it shall pass the service with a T_Data_Individual.req to the local Transport Layer; the parameter ASAP shall be mapped to the corresponding parameter TSAP of the T_Data_Individual.req primitive. If the parameter comm_mode equals "point-to-all-points connectionless", it shall pass the service with a T_Data_Broadcast.req to the local Transport Layer. The TSDU shall be always an A_NetworkParameter_Response-PDU, see Figure 27.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11 to N								Octet N+1 to 21							
						APCI										Parameter_type																								Test_info								Test_result							
																Object_type																PID																							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																								
						1	1	1	1	0	1	1	0	1	1																																								

Figure 27 – A_NetworkParameter_Response-PDU (example)

The local application layer shall map a T_Data_Broadcast.ind primitive or a T_Data_Individual.ind, with TSDU = A_NetworkParameter_Response-PDU to an A_NetworkParameter_Read.Acon primitive. The parameter priority, implicitly with value 'system', shall be mapped to the corresponding parameter priority of the A_NetworkParameter_Read.Acon primitive. The parameter TSAP shall be mapped to the corresponding parameter ASAP of the A_NetworkParameter_Read.Acon primitive.

A_NetworkParameter_Read.req(hop_count_type, parameter_type, priority, test_info)

hop_count_type: hop count 7 or standard

parameter_type: network parameter type that is verified, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

test_info: value against which the resource indicated by parameter_type is tested

A_NetworkParameter_Read.Lcon(hop_count_type, parameter_type, priority, test_info, a_status)

hop_count_type: hop count 7 or standard

parameter_type: network parameter type that is verified, structured as Interface Object Type and Property Identifier priority: system, urgent, normal or low priority

test_info value against which the resource indicated by parameter_type is tested

a_status: ok: A_NetworkParameter_Read.req sent successfully with T_Data_Broadcast service

not_ok: transmission of the associated T_Data_Broadcast request frame did not succeed

A_NetworkParameter_Read.ind(ASAP, hop_count_type, parameter_type, priority, test_info)

ASAP: local reference of the service access point or individual address

hop_count_type: hop count 7 or standard

parameter_type: network parameter type that is verified, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

test_info: value against which the resource indicated by parameter_type is tested

A_NetworkParameter_Read.res(ASAP, comm_mode, hop_count_type, parameter_type, priority, test_info, test_result)

ASAP: local reference of the service access point or individual address

comm_mode: point-to-all-points connectionless communication mode or point-to-point connectionless communication mode

hop_count_type: hop count 7 or standard

individual_address: the destination address for the service

parameter_type: network parameter type that is verified, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

test_info: value against which the resource indicated by parameter_type is tested

test_result: parameter_type dependent response

A_NetworkParameter_Read.Rcon(ASAP, comm_mode, hop_count_type, parameter_type, priority, test_info, test_result, a_status)

ASAP: local reference of the service access point or individual address

comm_mode: point-to-all-points connectionless communication mode or point-to-point connectionless communication mode

hop_count_type: hop count 7 or standard

individual_address: the destination address for the service

parameter_type: network parameter type that is verified, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

test_info: value against which the resource indicated by parameter_type is tested

test_result: parameter_type dependent response

a_status: ok: A_NetworkParameter_Read.res sent successfully with T_Data_Broadcast or T_Data_Individual service

not_ok: transmission of the associated T_Data_Broadcast or T_Data_Individual request frame did not succeed

A_NetworkParameter_Read.Acon(ASAP, hop_count_type, individual_address, parameter_type, priority, test_info, test_result)

ASAP:	local reference of the service access point or individual address
hop_count_type:	hop count 7 or standard
individual_address:	the individual address of the device that has sent the response
parameter_type:	network parameter type that is verified, structured as Interface Object Type and Property Identifier
priority:	system, urgent, normal or low priority
test_info:	value against which the resource indicated by parameter_type is tested
test_result:	parameter_type dependent response

7.2.10 A_NetworkParameter_Write service

The A_NetworkParameter_Write.req primitive shall be applied by the user of Application Layer of the management client to set network configuration information in one or multiple management servers. The service can be transmitted on either point-to-point connectionless or on point-to-all points (broadcast) communication mode. This shall be specified in the parameter comm_mode of the A_NetworkParameter_Write.req primitive.

The local Application Layer shall accept the service request. If the parameter comm_mode equals "point-to-point, connectionless", it shall pass the service with a T_Data_Individual.req to the local Transport Layer; the parameter ASAP shall be mapped to the corresponding parameter TSAP of the T_Data_Individual.req primitive. If the parameter comm_mode equals "point-to-all-points connectionless", it shall pass the service with a T_Data_Broadcast.req to the local Transport Layer. The TSDU shall in both cases always be an A_NetworkParameter_Write-PDU, see Figure 28.

If the remote Application Layer receives a T_Data_Individual.ind or a T_Data_Broadcast.ind with TSDU = A_NetworkParameter_Write-PDU, it shall map the service primitive to an A_NetworkParameter_Write.ind primitive. The parameter priority, implicitly with value 'system', shall be mapped to the corresponding parameter priority of the A_NetworkParameter_Write.ind primitive.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11…N							
						APCI										Parameter type																								Value							
																Interface object type																Property-ID															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																
						1	1	1	1	1	0	0	1	0	0																																

Figure 28 – A_NetworkParameter_Write-PDU (example)

Prior to passing an A_NetworkParameter_Write.Lcon primitive to the local application process, the local Application Layer needs a T_Data_Individual.con or a T_Data_Broadcast.con, depending on the parameter comm_mode of the service request, from the local Transport Layer. If the confirmation is positive (t_status = ok), the local Application Layer shall pass a positive A_NetworkParameter_Write.Lcon(a_status = ok) to the local application process. If the confirmation is negative (t_status = not_ok), the local Application Layer shall pass an A_NetworkParameter_Write.Lcon (a_status = not_ok) to the local user indicating that the transmission of the associated Transport Layer specification did not succeed.

NOTE This service is as such not confirmed by the remote partner. The A_NetworkParameter_Write.Lcon is only a local confirmation caused by the local Transport Layer confirmation, basically caused by the Data Link Layer confirmation (ok, not_ok).

A_NetworkParameter_Write.req(ASAP, comm_mode, hop_count_type, parameter_type, priority, value)

ASAP: local reference of the service access point or individual address

comm_mode: point-to-all-points connectionless communication mode or point-to-point connectionless communication mode

hop_count_type: hop count 7 or standard

parameter_type: the network parameter that shall be set, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

value: value to which the network parameter indicated by parameter_type shall be set

A_NetworkParameter_Write.Lcon(ASAP, comm_mode, hop_count_type, parameter_type, priority, value, a_status)

ASAP: local reference of the service access point or individual address

comm_mode: point-to-all-points connectionless communication mode or point-to-point connectionless communication mode

hop_count_type: hop count 7 or standard

parameter_type: the network parameter that shall be set, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

value: value to which the network parameter indicated by parameter_type shall be set

a_status: ok: A_NetworkParameter_Write.req sent successfully with the requested Transport Layer service

not_ok: transmission of the requested Transport Layer service did not succeed

A_NetworkParameter_Write.ind(ASAP, parameter_type, priority, value)

ASAP: local reference of the service access point or individual address

parameter_type: the network parameter that shall be set, structured as Interface Object Type and Property Identifier

priority: system, urgent, normal or low priority

value: value to which the network parameter indicated by parameter_type shall be set

7.3 Application layer services on point-to-point connectionless communication mode

7.3.1 General

A point-to-point connectionless communication mode shall be used to connect one device with another device. The following services can be applied on the point-to-point connectionless communication mode as well as on the point-to-point connection-oriented communication mode. The following subclauses specify the mapping of the services on the point-to-point connectionless communication mode. For using these services on a connection oriented communication mode, the T_Data_Connected service of transport layer shall be applied instead of the T_Data_Individual service.

For the connectionless communication mode the ASAP parameter shall be the individual address of the communication partner.

For the connection-oriented communication mode the ASAP parameter shall be the identifier of the communication relationship

The services on the point-to-point connectionless communication mode shall allow accessing properties of interface objects in the communication partner. Interface objects and their structure are specified in ISO/IEC 14543-3-3 (EN 50090-3-2).

7.3.2 A_PropertyValue_Read Service

The A_PropertyValue_Read.req primitive shall be applied by the user of application layer to read the value of a property of an interface object. The communication partner shall be addressed with a local ASAP that shall be mapped to a individual address by transport layer. The object of the partner shall be addressed with an object_index and the property of the object shall be addressed with a property_id. The nr_of_elem and start_index shall indicate the number of array elements starting with the given start_index in the property value that the user wants to read. The user of application layer in the partner device shall respond with an A_PropertyValue_Read.res, i.e. the service shall be confirmed by the remote application process.

The local application layer shall accept the service request and pass it with a T_Data_Individual.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters ASAP and priority of the T_Data_Individual.req primitive, the TSDU shall be an A_PropertyValue_Read-PDU, see Figure 29.

The remote application layer shall map a T_Data_Individual.ind primitive with TSDU = A_PropertyValue_Read-PDU to an A_PropertyValue_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_PropertyValue_Read.ind primitive.

The remote application process shall respond to the A_PropertyValue_Read.ind primitive with an A_PropertyValue_Read.res primitive containing the requested number of elements of the property value of the property of the associated interface object. If the remote application process has a problem, for example, object or property doesn't exist or the data does not fit in a PDU or the requester has not the required access rights, then the nr_of_elem of the A_PropertyValue_Response-PDU shall be zero and shall contain no data.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11							
						APCI										Object_index								Property_id								nr_of_elem				Start_index											
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																
						1	1	1	1	0	1	0	1	0	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 29 – A_PropertyValue_Read-PDU (example)

<table>
<tr><td colspan="8">Octet 6</td><td colspan="8">Octet 7</td><td colspan="8">Octet 8</td><td colspan="8">Octet 9</td><td colspan="8">Octet 10</td><td colspan="8">Octet 11</td><td colspan="8">Octet 12 to Octet N</td></tr>
<tr><td colspan="6"></td><td colspan="10">APCI</td><td colspan="8">Object_index</td><td colspan="8">Property_id</td><td colspan="4">nr_of_elem</td><td colspan="12">Start_index</td><td colspan="8">Data</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="6"></td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td colspan="8"></td><td colspan="8"></td><td colspan="4"></td><td colspan="12"></td><td colspan="8"></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

Figure 30 – A_PropertyValue_Response-PDU (example)

The remote application layer shall accept the service response and pass it with a T_Data_Individual.req to the remote transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters TSAP and priority of the T_Data_Individual.req primitive, the TSDU shall be an A_PropertyValue_Response-PDU, see Figure 30.

The local application layer shall map a T_Data_Individual.ind primitive with TSDU = A_PropertyValue_Response-PDU to an A_PropertyValue_Read.Acon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_PropertyValue_Read.Acon primitive.

A_PropertyValue_Read.req(ack_request, priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the object addressed

property_id: this parameter shall contain the property_id of the property of the object

nr_of_elem: this parameter shall contain the number of array elements that shall be read in the property value

start_index: this parameter shall contain the array index of the first array element that shall be read

A_PropertyValue_Read.Lcon(ack_request, priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the object addressed property_id: this parameter shall contain the property_id of the property of the object

nr_of_elem: this parameter shall contain the number of array elements that shall be read in the property value

start_index: this parameter shall contain the array index of the first array element that shall be read

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_PropertyValue_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_PropertyValue_Read.req did not succeed

A_PropertyValue_Read.ind(priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the object addressed

property_id: this parameter shall contain the property_id of the property of the object addressed

nr_of_elem: this parameter shall contain the number of array elements that shall be read in the property value

start_index: this parameter shall contain the array index of the first array element that shall be read

A_PropertyValue_Read.res(ack_request, priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

nr_of_elem: this parameter shall contain the number of array elements that has been read in the property value, or zero if a problem occurred

start_index: this parameter shall contain the array index of the first array element read

data: this parameter shall contain the value of the array elements read, or no data, if a problem occurred

A_PropertyValue_Read.Acon(priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index, data)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the object addressed

property_id: this parameter shall contain the property_id of the property of the addressed object

nr_of_elem: this parameter shall contain the number of array elements that has been read in the property value or zero if a problem occurred

start_index: this parameter shall contain the array index of the first array element read

data: this parameter shall contain the value of the array elements read, or no data, if a problem occurred

7.3.3 A_PropertyValue_Write service

The A_PropertyValue_Write.req primitive shall be applied by the user of application layer to modify the value of a property of an interface object. The communication partner shall be addressed with a local ASAP that shall be mapped to an individual address by the transport layer. The object of the partner shall be addressed with the object_index and the property of the object shall be addressed with the property_id. The nr_of_elem and start_index shall indicate the number of array elements that shall be written in the property value, starting with the given start_index.

The local application layer shall accept the service request and pass it with a T_Data_Individual.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters TSAP and priority of the T_Data_Individual.req primitive, the TSDU shall be an A_PropertyValue_Write-PDU, see Figure 31.

The remote application layer shall map a T_Data_Individual.ind primitive with TSDU = A_PropertyValue_Write-PDU to an A_PropertyValue_Write.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_PropertyValue_Write.ind primitive.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11								Octet 12 to Octet N							
						APCI										Object_index								Property_id								nr_of_elem				Start_index												Data							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																								
						1	1	1	1	0	1	0	1	1	1																																								

Figure 31 – A_PropertyValue_Write-PDU (example)

The remote application process shall respond to the A_PropertyValue_Write.ind primitive with an A_PropertyValue_Read.res primitive containing the requested number of elements of the property value of the property of the associated object. The value of the property of the associated object shall be explicitly read back after writing to it.

If the remote application process has a problem, for example object or property doesn't exist or the requester does not have the required access rights, then the nr_of_elem of the A_PropertyValue_Response-PDU, as specified in Figure 30 shall be zero and shall contain no data.

The remote application layer shall accept the service response and pass it with a T_Data_Individual.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters TSAP and priority of the T_Data_Individual.req primitive, the TSDU shall be an A_PropertyValue_Response-PDU.

The local application layer shall map a T_Data_Individual.ind primitive with TSDU = A_PropertyValue_Response-PDU to an A_PropertyValue_Write.Acon primitive if an A_PropertyValue_Write-PDU has been sent before to this communication partner to this object and property. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_PropertyValue_Read.Acon primitive.

A_PropertyValue_Write.req(ack_request, priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

nr_of_elem: this parameter shall contain the number of array elements that shall be written in the property value

start_index: this parameter shall contain the array index of the first array element that shall be written

data: this parameter shall contain the data that shall be written to the array elements

A_PropertyValue_Write.Lcon(ack_request, priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index, data, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to the transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object nr_of_elem: this parameter shall contain the number of array elements that shall be written in the property value

start_index: this parameter shall contain the array index of the first array element that shall be written

data: this parameter shall contain the data that shall be written to the array elements

a_status: ok: this value of this parameter shall be used to indicate that the transmission of the A_PropertyValue_Write.req has been successful

not_ok: this value of this parameter shall be used to indicate that the transmission of the A_PropertyValue_Write.req did not succeed

A_PropertyValue_Write.ind(priority, hop_count_type, ASAP, object_index, property_id, nr_of_elem, start_index, data)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

nr_of_elem: this parameter shall contain the number of array elements that shall be written in the property value

start_index: this parameter shall contain the array index of the first array element that shall be written

data: this parameter shall contain the data that shall be written to the array elements

7.3.4 A_PropertyDescription_Read service

The A_PropertyDescription_Read.req primitive shall be applied by the user of application layer, to read the description of the property of an object. The communication partner shall be addressed with a local ASAP that shall be mapped to a individual address by transport layer. The object of the partner shall be addressed with an object_index and the property of the object shall be addressed with a property_id or with a property_index. The property_index shall be used only if the property_id is zero. The property_index, if evaluated, shall address the property of the object with a sequential number, for example, property_index = 0 shall mean first property of the associated object, property_index = 1 shall mean second property. The service shall be confirmed by the remote application process.

The local application layer shall accept the service request and pass it with a T_Data_Individual.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters TSAP and priority of the T_Data_Individual.req primitive, the TSDU shall be an A_PropertyDescription_Read-PDU, see Figure 32.

The remote application layer shall map a T_Data_Individual.ind primitive with TSDU = A_PropertyDescription_Read-PDU to an A_PropertyDescription_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_PropertyDescription_Read.ind primitive.

The remote application process shall respond to the A_PropertyDescription_Read.ind primitive with an A_PropertyDescription_Read.res primitive containing the description of the property of the associated property of the object addressed.

If the property_id in the A_PropertyDescription_Read-PDU is zero, the remote application process shall use the indicated property_index to access the property description, otherwise the property_id shall be used. If the remote application process has a problem, for example, object or property doesn't exist, then the max_nr_of_elem of the A_PropertyDescription_Response-PDU shall be zero.

The service shall not be confirmed negative for authorization reasons (see A_Authorize_Request Service in 7.4.8).

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10							
						APCI										Object_index								Property_id								Property_index							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																								
						1	1	1	1	0	1	1	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 32 – A_PropertyDescription_Read-PDU (example)

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10							
						APCI										Object_index								Property_id								Property_index							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																								
						1	1	1	1	0	1	1	0	0	1																								

Octet 11								Octet 12								Octet 13								Octet 14							
Type								max_nr_of_elem																Access							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
																								read_level				write_level			
						1	1	1	1	0	1	1	0	0	1																

Figure 33 – A_PropertyDescription_Response-PDU (example)

The remote application layer shall accept the service response and pass it with a T_Data_Individual.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters TSAP and priority of the T_Data_Individual.req primitive, the TSDU shall be a A_PropertyDescription_Response-PDU, see Figure 33.

The local application layer shall map a T_Data_Individual.ind primitive with TSDU = A_PropertyDescription_Response-PDU to an A_PropertyDescription_Read.Acon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_PropertyDescription_Read.Acon primitive.

A_PropertyDescription_Read.req(ack_request, priority, hop_count_type, ASAP, object_index, property_id, property_index)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

property_index: this parameter shall contain the sequential property number of the property of the addressed object

A_PropertyDescription_Read.Lcon(ack_request, priority, hop_count_type, ASAP, object_index, property_id, property_index, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

property_index: this parameter shall contain the sequential property number of the property of the addressed object

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_PropertyDescription_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_PropertyDescription_Read.req did not succeed

A_PropertyDescription_Read.ind(priority, hop_count_type, ASAP, object_index, property_id, property_index)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object property_id: this parameter shall contain the property_id of the property of the addressed object

property_index: this parameter shall contain the sequential property number of the property of the addressed object

A_PropertyDescription_Read.res(ack_request, priority, hop_count_type, ASAP, object_index, property_id, property_index, type, max_nr_of_elem, access)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

property_index: this parameter shall contain the sequential property number of the property of the addressed object

max_nr_of_elem: this parameter shall contain the maximum number of elements of the array of the property value of the addressed object; it shall contain zero to indicate a problem

access: this parameter shall contain the access level to read or write to the property value of the property of the addressed object

A_PropertyDescription_Read.Acon(priority, hop_count_type, ASAP, object_index, property_id, property_index, type, max_nr_of_elem, access)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point or individual address

object_index: this parameter shall contain the object_index of the addressed object

property_id: this parameter shall contain the property_id of the property of the addressed object

property_index: this parameter shall contain the sequential property number of the property of the addressed object

max_nr_of_elem: this parameter shall contain the maximum number of elements of the array of the property value of the addressed object; it shall contain zero to indicate a problem

access: this parameter shall contain the access level to read or write to the property value of the property of the addressed object

7.3.5 A_DeviceDescriptor_Read service

The A_DeviceDescriptor_Read.req primitive shall be applied by the user of application layer, to read the device descriptor of the communication controller in a communication partner. The service shall be confirmed by the remote application process containing the device descriptor information.

The local application layer shall accept the service request and pass it with a T_Data_Individual.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Individual.req primitive, the TSDU shall be an A_DeviceDescriptor_Read-PDU, see Figure 34.

The remote application layer shall map a T_Data_Individual.ind primitive with TSDU = A_DeviceDescriptor_Read-PDU to an A_DeviceDescriptor_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_DeviceDescriptor_Read.ind primitive.

Octet 6								Octet 7							
						APCI				Descriptor_type					
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI						
						1	1	0	0	x	x	x	x	x	x

Figure 34 – A_DeviceDescriptor_Read-PDU (example)

The remote application process shall respond to the A_DeviceDescriptor_Read.ind primitive with an A_DeviceDescriptor_Read.res primitive containing the device descriptor information.

The device descriptor information that shall be used is specified in ISO/IEC 14543-3-4 (EN 50090-7-1).

Octet 6								Octet 7								Octet 8 to Octet n															
						APCI				Descriptor_type						Device descriptor															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI																						
						1	1	0	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 35 – A_DeviceDescriptor_Response-PDU (example)

The remote application layer shall accept the service response and pass it with a T_Data_Individual.req to the local transport layer. The parameters TSAP and priority shall be mapped to the corresponding parameters of the T_Data_Individual.req primitive, the TSDU shall be a A_DeviceDescriptor_Response-PDU, see Figure 35.

The local application layer shall map a T_Data_Individual.ind primitive with TSDU = A_DeviceDescriptor_Response-PDU to an A_DeviceDescriptor_Read.Acon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_DeviceDescriptor_Read.Acon primitive.

A_DeviceDescriptor_Read.req(ack_request, priority, hop_count_type, ASAP, descriptor_type)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

descriptor_type this parameter shall contain the requested device descriptor type

A_DeviceDescriptor_Read.Lcon(ack_request, priority, hop_count_type, ASAP, descriptor_type, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point

descriptor_type this parameter shall contain the requested device descriptor type

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_DeviceDescriptor_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_DeviceDescriptor_Read.req did not succeed

A_DeviceDescriptor_Read.ind(priority, hop_count_type, ASAP, descriptor_type)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

descriptor_type this parameter shall contain the requested device descriptor type

A_DeviceDescriptor_Read.res(ack_request, priority, hop_count_type, ASAP, descriptor_type, device_descriptor)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

descriptor_type this parameter shall contain the descriptor type of the device descriptor contained in the response

device_descriptor: this parameter shall contain the device descriptor of the remote device

A_DeviceDescriptor_Read.Acon(priority, hop_count_type, ASAP, descriptor_type, device_descriptor)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point descriptor_type this parameter shall contain the descriptor type of the device descriptor contained in the response r

device_descriptor: this parameter shall contain the device descriptor of the remote device

7.3.6 A_Link_Read service

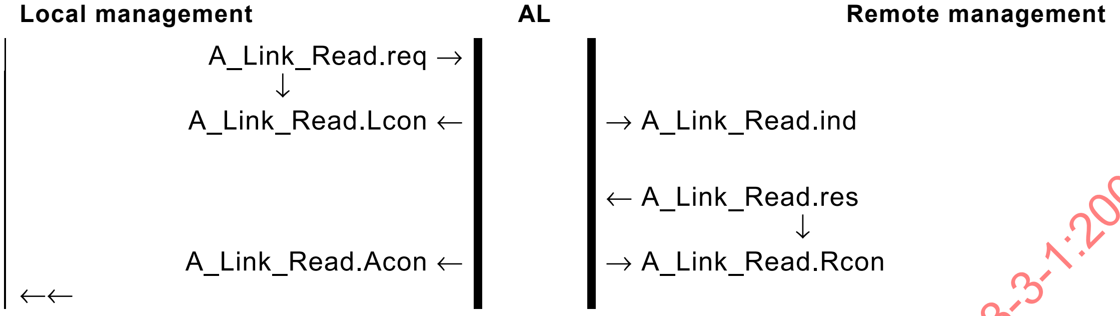


Figure 36 – Message flow for A_Link_Read service

The message flow of the A_Link_Read service is shown in Figure 36.

The A_Link_Read.req primitive shall be applied by the user of Application Layer (Local Management) to read the links to a given Group Object in a communication partner. It shall indicate the start_index from which Group Addresses shall be read in the list of links.The TSDU used by the local application layer shall be an A_Link_Read-PDU, see Figure 37.

Octet 6								Octet 7								Octet 8								Octet 9							
						APCI										Group_object_number								Start_index							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI													–	–	–	–
						1	1	1	1	1	0	0	1	0	1	x	x	x	x	x	x	x	x	0	0	0	0				

Figure 37 – A_Link_Read-PDU (example)

The remote management process shall respond to the A_Link_Read.ind primitive with an A_Link_Read.res primitive. The TSDU used by the remote application layer shall be an A_Link_Read-PDU, see Figure 38.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								...								Octet N ≤ 21							
						APCI										Group_object_number								Sending_-address				Start_-address				Group_address_list																							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI									S	S	S	S	–	–	–	–																								
						1	1	1	1	1	0	0	1	1	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 38 – A_Link_Response-PDU

This response shall indicate the updated list of links on the addressed Group Object, and the sending Group Address, if any.

A_Link_Read.req (ack_request, priority, hop_count_type, ASAP, group_object_number, start_index)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

group_object_number:local index of the intended Group Object

start_index index of first GA to send back in the list of GAs attached to the Group Object

A_Link_Read.res (ack_request, priority, hop_count_type, ASAP, group_object_number, sending_address, start_index, group_address_list)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

group_object_number:local index of the intended Group Object

sending_address: index of the sending address in the list of GAs attached to the Group Object

start_index: index of the first transmitted GA in the list of GAs attached to the Group Object

group_address_list: list of Group Addresses connected to the Group Object; 0 to 6 group addresses can be contained, each occupying.

7.3.7 A_Link_Write service

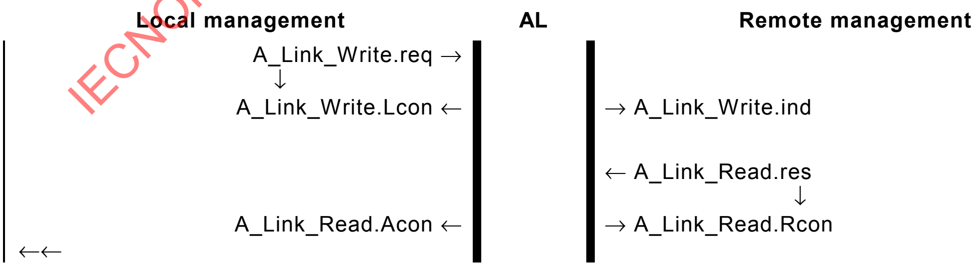


Figure 39 – Message flow for A_Link_Write service

The message flow of the A_Link_Write service is shown in Figure 39.

The A_Link_Write.req primitive shall be applied by the user of Application Layer (Local Management) to write a link to a given Group Object in a communication partner. The contained flags shall indicate that the action is “add a link”. It shall in addition indicate if the

Group Address is a sending address or not. The TSDU used by the local application layer shall be an A_Link_Write-PDU, see Figure 1.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11							
						APCI										Group_object_number								Flags								Group_address															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI															d flag	s flag																
						1	1	1	1	1	0	0	1	1	1	x	x	x	x	x	x	x	x	0	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 40 – A_Link_Write-PDU

The remote management process shall respond to the A_Link_Write.ind primitive with an A_Link_Read.res primitive.

The Link Management information that shall be used is specified in ISO/IEC 14543-3-4 (EN 50090-7-1).

A_Link_Write.req (ack_request, priority, hop_count_type, ASAP, group_object_number, flags, group_address)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point or individual address

group_object_number:local index of the intended Group Object

flags: action to be done (delete/add); sending at flag (yes/no)

group_address: Group Address corresponding to the link to add or suppress

7.4 Application layer services on point-to-point connection-oriented communication mode

7.4.1 General

A point-to-point connection-oriented communication mode shall connect one device with another device. The following services can be applied on point-to-point connection-oriented communication modes if the connection is established; the requirements for establishing a connection between devices are specified in ISO/IEC 14543-3-2. Due to the behaviour of the transport layer state machine, the user of the application layer shall take into account that the connection can be released by the remote communication partner or by an error detected in the communication protocol. Therefore, a T_Disconnect.ind primitive can occur at any time, i.e. also if the user of an application layer is waiting for a confirmation from the application layer. The transport layer services T_Connect.ind and T_Disconnect.ind shall be mapped transparently to A_Connect.ind and A_Disconnect.ind service and passed to the user of application layer.

The application layer shall also provide an access protection mechanism on the point-to-point connection-oriented communication mode by an authorization procedure. This procedure is described in 7.4.8.

7.4.2 A_ADC_Read service

The A_ADC_Read.req primitive shall be applied by the user of application layer to read the value of the Analog-to-Digital (AD)-converter. The service shall be confirmed by the remote application process containing the value of the converter.

The local application layer shall accept the service request and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_ADC_Read-PDU, see Figure 41.

The remote application layer shall map a T_Data_Connected.ind primitive with TSDU = A_ADC_Read-PDU to an A_ADC_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_ADC_Read.ind primitive.

The A_ADC_Read-PDU shall contain the channel number of the AD-converter and the number of consecutive read operations to the AD-converter.

Octet 6								Octet 7								Octet 8							
						APCI										Read_count							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	Channel_nr	channel_nr	channel_nr	channel_nr	channel_nr	channel_nr								
						0	1	1	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 41 – A_ADC_Read-PDU (example)

The remote application process shall respond to the A_ADC_Read.ind primitive with an A_ADC_Read.res primitive containing the value of the AD-converter computed by the summation of the consecutive Central Processing Unit (CPU) accesses. If the remote application process has a problem, for example overflow when computing the summation, or wrong channel number, then the read_count of the A_ADC_Response-PDU shall be zero.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10							
						APCI										Read_count								Sum of AD_converter_access															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
										Channel_nr														Value high								Value low							
						APCI	APCI	APCI	APCI																														
						0	1	1	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 42 – A_ADC_Response-PDU (example)

The remote application layer shall accept the service response and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be a A_ADC_Response-PDU, see Figure 42.

The local application layer shall map a T_Data_Connected.ind primitive with TSDU = A_ADC_Response-PDU to an A_ADC_Read.Acon primitive. The parameters TSAP and priority shall be mapped to the corresponding parameters ASAP and priority of the A_ADC_Read.Acon primitive.

A_ADC_Read.req(ack_request, priority, hop_count_type, ASAP, channel_nr, read_count)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

channel_nr: this parameter shall contain the number of the channel of the AD-converter that shall be read

read_count: this parameter shall contain the number of desired consecutive CPU accesses that shall be done to the requested AD-converter channel

A_ADC_Read.Lcon(ack_request, priority, hop_count_type, ASAP, channel_nr, read_count, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point

channel_nr: this parameter shall contain the number of the AD-converter channel that shall be read

read_count: this parameter shall contain the number of desired consecutive CPU accesses that shall be done to the requested AD-converter channel

a_status: ok: this value of this parameter shall be used to indicate that the transmission of the A_ADC_Read.req service has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_ADC_Read.req service did not succeed

A_ADC_Read.ind(priority, hop_count_type, ASAP, channel_nr, read_count)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

channel_nr: this parameter shall contain the number of the AD-converter channel that shall be read

read_count: this parameter shall contain the number of desired consecutive CPU accesses that shall be done to the requested AD-converter channel

A_ADC_Read.res(ack_request, priority, hop_count_type, ASAP, channel_nr, read_count, sum)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

channel_nr: this parameter shall contain the number of the AD-converter channel that has been read

read_count: this parameter shall contain the number of executed consecutive CPU accesses to the requested AD-converter channel; it shall contain zero to indicate a problem

sum: this parameter shall contain the sum of the consecutively read AD-converter values

A_ADC_Read.Acon(priority, hop_count_type, ASAP, channel_nr, read_count, sum)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be "system", "urgent", "normal" or "low"

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

channel_nr: this parameter shall contain the number of the AD-converter channel that has been read

read_count: this parameter shall contain the number of executed consecutive CPU accesses to the requested AD-converter channel; it shall contain zero to indicate a problem

sum: this parameter shall contain the sum of the consecutively read AD-converter values

7.4.3 A_Memory_Read service

The A_Memory_Read.req primitive shall be applied by the user of application layer, to read between 1 and 12 octets in the address space of the remote communication controller. The parameter memory_address shall specify the 16 bit start address and number shall contain the number of octets that shall be read beginning with the start address to increasing addresses. The service shall be confirmed by the remote application process with the contents of the address space.

The local application layer shall accept the service request and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_Memory_Read-PDU, see Figure 43.

The remote application layer shall map a T_Data_Connected.ind primitive with TSDU = A_Memory_Read-PDU to an A_Memory_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_Memory_Read.ind primitive.

Octet 6								Octet 7								Octet 8								Octet 9							
						APCI						Number				Address (high)								Address (low)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI																						
						1	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 43 – A_Memory_Read-PDU (example)

The remote application process shall respond to the A_Memory_Read.ind primitive with an A_Memory_Read.res primitive containing the number of octets read beginning with the start address to increasing addresses. If the remote application process has a problem, for example address space unreachable or protected or an illegal number of octets is requested, then the parameter number of the A_Memory_Response-PDU shall be zero and shall contain no data.

If the remote application process is requested to read data from a protected area or from any logical address that is not associated to physical memory then no data should be returned to indicate an error. The same should apply if only part of the memory that is requested to be read from is protected or physically existing.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10 to Octet N							
						APCI						Number				Address (high)								Address (low)								Data							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI																														
						1	0	0	1	0	0																												

Figure 44 – A_Memory_Response-PDU (example)

The remote application layer shall accept the service response and pass it with a T_Data_Connected.req to the local transport layer. The parameters TSAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_Memory_Response-PDU, see Figure 44.

The application layer shall map a T_Data_Connected.ind primitive with TSDU = A_Memory_Response-PDU to an A_Memory_Read.Acon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_Memory_Read.Acon primitive.

A_Memory_Read.req(ack_request, priority, hop_count_type, ASAP, number, memory_address)

ack_request:	this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional
priority:	this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”
hop_count_type:	this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used
ASAP:	this parameter shall be used to contain the service access point
number:	this parameter shall contain the number of octets that shall be read beginning with the start address to increasing addresses
memory_address:	this parameter shall contain the 16 bit start address

A_Memory_Read.Lcon(ack_request, priority, hop_count_type, ASAP, number, memory_address, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be read beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_Memory_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_Memory_Read.req did not succeed

A_Memory_Read.ind(priority, hop_count_type, ASAP, number, memory_address)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be read beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

A_Memory_Read.res(ack_request, priority, hop_count_type, ASAP, number, memory_address, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that has been read beginning with the start address to increasing addresses; it shall contain zero to indicate a problem

memory_address: this parameter shall contain the 16 bit start address

data: this parameter shall contain the octet(s) that have been read

7.4.4 A_Memory_Write service

The A_Memory_Write.req primitive shall be applied by the user of application layer, to write between 1 and 12 octets in the address space of the remote communication controller. The parameter memory_address shall specify the 16 bit start address and the parameter number shall contain the number of octets that shall be written beginning with the start address to increasing addresses.

The service shall be a confirmed service if verify mode is active, otherwise it shall be an acknowledged service.

The local application layer shall accept the service request and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_Memory_Write-PDU, see Figure 45.

The remote application layer shall map a T_Data_Connected.ind primitive with TSDU = A_Memory_Write-PDU to an A_Memory_Write.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments TSAP and priority of the A_Memory_Write.ind primitive.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10 to N							
						APCI						Number				Address (high)								Address (low)								Data							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI																														
						1	0	1	0	0	0																												

Figure 45 – A_Memory_Write-PDU (example)

With inactive verify mode the remote application process shall not respond. Instead the local application layer shall map a T_Data_Connected.con primitive to an A_Memory_Write.Lcon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_Memory_Write.Lcon primitive; the parameters number, memory_address and data shall not be evaluated by the application layer.

With active verify mode the remote application process shall respond to the A_Memory_Write.ind primitive with an A_Memory_Write.res primitive containing the requested number of octets of the associated memory area. The value of the associated memory area shall be explicitly read back after writing to it. If the remote application process has a problem, for example memory area unreachable or protected, or an illegal number of octets is requested, then the parameter number shall be zero and shall contain no data.

If the remote application process is requested to write data to a protected area from any logical address which is not associated to physical memory then the service indication shall be ignored. In any case, physical memory shall not be addressable via different logical addresses. If only a part of the addressed memory is protected or does not exist, then the complete write operation shall fail. If verify mode is active, then in case of a failed write operation no data should be returned to indicate an error.

The remote application layer shall accept the service response and shall pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_Memory_Response-PDU.

The local application layer shall map a T_Data_Connected.ind primitive with TSDU = A_Memory_Response-PDU to an A_Memory_Write.Acon primitive if an A_Memory_Write-PDU has been sent before over this connection. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_Memory_Write.Acon primitive.

A_Memory_Write.req(ack_request, priority, hop_count_type, TSAP, number, memory_address, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be written beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

data: this parameter contains the octet(s) that shall be written

A_Memory_Write.Lcon(ack_request, priority, hop_count_type, ASAP, number, memory_address, data, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be written beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

data: this parameter contains the octet(s) that shall be written or no data

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_Memory_Write.req service has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_Memory_Write.req service did not succeed

A_Memory_Write.ind(priority, hop_count_type, TSAP, number, memory_address, data)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be written beginning with the start address to increasing addresses memory_address: this parameter shall contain the 16 bit start address

data: this parameter contains the octet(s) that shall be written

A_Memory_Write.res(ack_request, priority, hop_count_type, ASAP, number, memory_address, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that has been written beginning with the start address to increasing addresses; it shall contain zero to indicate a problem

memory_address: this parameter shall contain the 16 bit start address

data: this parameter shall contain the octet(s) that have been read back; it shall contain no data to indicate a problem

A_Memory_Write.Acon(priority, hop_count_type, ASAP, number, memory_address, data)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that has been written beginning with the start address to increasing addresses; it shall contain zero to indicate a problem

memory_address: this parameter shall contain the 16 bit start address

data: this parameter shall contain the octet(s) that have been read back; it shall contain no data to indicate a problem

7.4.5 A_MemoryBit_Write service

The A_MemoryBit_Write.req primitive shall be applied by the user of application layer, to modify between 1 bit and 40 bits in a contiguous block of up to 5 octets in the address space of the remote communication controller. The parameter memory_address shall specify the 16 bit start address and the parameter number shall contain the number of octets that shall be modified beginning with the start address to increasing addresses. The A_MemoryBit_Write shall allow to

- set individual bits of the contiguous block to zero,
- set individual bits of the contiguous block to one,
- leave individual bits of the contiguous block unmodified,
- invert individual bits of the contiguous block.

This shall be done by using the parameters and_data and xor_data. Both parameters shall have the same number of octets as the contiguous block indicated in the parameter number. The resulting value for each individual bit in the contiguous block shall be computed using the two associated bits of and_data and xor_data with the following function:

result_bit(i) = (and_data_bit(i) AND block_bit(i)) XOR xor_data_bit(i)

The possible results are specified in Table 2:

Table 2 – Function table for A_MemoryBit_Write-Services

and_data_bit(i)	xor_data_bit(i)	result_bit(i)
0	0	0
0	1	1
1	0	block_bit(i)
1	1	NOT block_bit(i)

The service shall be a confirmed service if verify mode is active, otherwise it shall be an acknowledged service.

The local application layer shall accept the service request and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_MemoryBit_Write-PDU, see Figure 46.

With inactive verify mode the remote application process shall not respond. Instead the local application layer shall map a T_Data_Connected.con primitive to an A_MemoryBit_Write.Lcon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_MemoryBit_Write.Lcon primitive; the parameters number, memory_address and data shall not be evaluated by the application layer.

With active verify mode the remote application process shall respond to the A_MemoryBit_Write.ind primitive with an A_MemoryBit_Write.res primitive containing the requested number of octets of the associated memory area. The value of the associated memory area shall be explicitly read back after writing to it. If the remote application process has a problem, for example memory area unreachable or protected or an illegal number of octets is requested, then the parameter number shall be zero and shall contain no data.

If the remote application process is requested to write data to a protected area or from any logical address which is not associated to physical memory then the service indication shall be ignored. In any case, physical memory shall not be addressable via different logical addresses. If only a part of the addressed memory is protected or does not exist, then the complete write operation shall fail. If verify mode is active, then in case of a failed write operation no data should be returned to indicate an error.

<table>
<tr><td colspan="8">Octet 6</td><td colspan="8">Octet 7</td><td colspan="8">Octet 8</td><td colspan="8">Octet 9</td><td colspan="8">Octet 10</td></tr>
<tr><td colspan="6"></td><td colspan="10">APCI</td><td colspan="8">Number</td><td colspan="8">Address
(high)</td><td colspan="8">Address
(low)</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="6"></td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td>APCI</td><td colspan="8"></td><td colspan="8"></td><td colspan="8"></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td colspan="8">Octet 11 to Octet n</td><td colspan="8">Octet (n+1) to Octet m</td></tr>
<tr><td colspan="8">and_data</td><td colspan="8">xor_data</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8"></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8">n = 10 + number</td><td colspan="8">m = 10 + 2 x number</td></tr>
</table>

Figure 46 – A_MemoryBit_Write-PDU

The remote application layer shall accept the service response and shall pass it with a T_Data_Connected.req to the local transport layer. The parameters TSAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be a A_Memory_Response-PDU.

The local application layer shall map a T_Data_Connected.ind primitive with TSDU = A_Memory_Response-PDU to an A_MemoryBit_Write.Acon primitive if an A_MemoryBit_Write-PDU has been sent before over this connection. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_MemoryBit_Write.Acon primitive.

A_MemoryBit_Write.req(ack_request, priority, hop_count_type, ASAP, number, memory_address, and_data, xor_data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be modified beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

and_data: this parameter shall contain the and_data: see Table 2

xor_data: this parameter shall contain the xor_data: see Table 2

A_MemoryBit_Write.Lcon(ack_request, priority, hop_count_type, ASAP, number, memory_address, and_data, xor_data, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low” hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be modified beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

and_data: this parameter shall contain the and_data: see Table 2

xor_data: this parameter shall contain the xor_data: see Table 2

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_MemoryBit_Write.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_MemoryBit_Write.req did not succeed

A_MemoryBit_Write.ind(priority, hop_count_type, ASAP, number, memory_address, and_data, xor_data)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be modified beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

and_data: this parameter shall contain the and_data: see Table 2

xor_data: this parameter shall contain the xor_data: see Table 2

A_MemoryBit_Write.res(ASAP, priority, number, memory_address, data)

ASAP: this parameter shall be used to contain the service access point

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

number: this parameter shall contain the number of octets that has been modified beginning with the start address to increasing addresses; it shall contain zero to indicate a problem

memory_address: this parameter shall contain the 16 bit start address

data: this parameter shall contain the octet(s) that have been read back; it shall contain no data to indicate a problem

7.4.6 A_UserData

7.4.6.1 Usage

The A_UserData-service shall be used for Application Device Management. The Application Device Management is that part of the device-management, which is implemented in the user application.

The Application Device Management shall use a logical address-space of 1 Mb. The mapping from the logical address space to the individual addresses and vice versa shall be the task of the Application Device Management.

The Application Device Management may also influence directly or indirectly the application-program. But this should be defined individually for each device.

All services provided by the Application Device Management shall be based on the A_USER_DATA-messages and the corresponding service.

7.4.6.2 A_UserMemory_Read service

The A_UserMemory_Read.req primitive shall be applied by the user of application layer, to read between 1 and 11 octets in the address space of the remote application controller. The parameter memory_address shall specify the 20 bit start address (4 bit AddressExtension + 8 bit Address High + 8 bit Address Low) and the parameter number shall contain the number of octets that shall be read beginning with the start address to increasing addresses. The service shall be confirmed by the remote application process with the contents of the address space.

The local application layer shall accept the service request and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_UserMemory_Read-PDU, see Figure 47.

The remote application layer shall map a T_Data_Connected.ind primitive with TSDU = A_UserMemory_Read-PDU to an A_UserMemory_Read.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_UserMemory_Read.ind primitive.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10							
																Address extension				Number				Address (high)								Address (low)							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																								
						1	0	1	1	0	0	0	0	0	0																								

Figure 47 – A_UserMemory_Read-PDU (example)

The remote application process shall respond to the A_UserMemory_Read.ind primitive with an A_UserMemory_Read.res primitive containing the number of octets read beginning with the start address to increasing addresses. If the remote application process has a problem, for example address space unreachable or protected or an illegal number of octets is requested, then the parameter number of the A_UserMemory_Response-PDU shall be zero and shall contain no data.

If the remote application process is requested to read data from a protected area or from any logical address that is not associated to physical memory then no data shall be returned to indicate an error. The same shall apply if only part of the memory to be read is protected or physically existing.

Octet 6								Octet 7								Octet 8								Octet 9								Octet 10								Octet 11 to N							
																Address extension				Number				Address (high)								Address (low)								Data							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
						APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI																																
						1	0	1	1	0	0	0	0	0	1																																

Figure 48 – A_UserMemory_Response-PDU

The remote application layer shall accept the service response and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be a A_UserMemory_Response-PDU, see Figure 48.

The local application layer shall map a T_Data_Connected.ind primitive with TSDU = A_UserMemory_Response-PDU to an A_UserMemory_Read.Acon primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_UserMemory_Read.Acon primitive.

A_UserMemory_Read.req(ack_request, priority, hop_count_type, ASAP, number, memory_address)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be read beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

A_UserMemory_Read.Lcon(ack_request, priority, hop_count_type, ASAP, number, memory_address, a_status)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge has been indicated as mandatory or optional in the transmitted frame

priority: this parameter shall be used to indicate the priority that has been used to transmit the requested frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count of the transmitted frame has been set to 7 or if the network layer parameter has been used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be read beginning with the start address to increasing addresses

memory_address: this parameter shall contain the 16 bit start address

a_status: ok: the value of this parameter shall be used to indicate that the transmission of the A_UserMemory_Read.req has been successful

not_ok: the value of this parameter shall be used to indicate that the transmission of the A_UserMemory_Read.req did not succeed

A_UserMemory_Read.ind(priority, hop_count_type, ASAP, number, memory_address)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that shall be read beginning with the start address to increasing addresses memory_address: this parameter shall contain the 16 bit start address

A_UserMemory_Read.res(ack_request, priority, hop_count_type, ASAP, number, memory_address, data)

ack_request: this parameter shall be used to indicate whether a layer-2 acknowledge is mandatory or optional

priority: this parameter shall be used to contain the priority that shall be used to transmit the requested service; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop_count shall be set to 7 or if the network layer parameter shall be used

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that has been read beginning with the start address to increasing addresses; it shall contain zero to indicate a problem

memory_address: this parameter shall contain the 16 bit start address

data: this parameter shall contain the octet(s) that have been read

A_UserMemory_Read.Acon(priority, hop_count_type, ASAP, number, memory_address, data)

priority: this parameter shall be used to indicate the priority of the received frame; it shall be “system”, “urgent”, “normal” or “low”

hop_count_type: this parameter shall be used to indicate whether the hop count of the received frame equals 7 or not

ASAP: this parameter shall be used to contain the service access point

number: this parameter shall contain the number of octets that has been read beginning with the start address to increasing addresses; it shall contain zero to indicate a problem

memory_address: this parameter shall contain the 16 bit start address

data: this parameter shall contain the octet(s) that have been read

7.4.6.3 A_UserMemory_Write service

The A_UserMemory_Write.req primitive shall be applied by the user of application layer, to write between 1 and 11 octets in the individual address space of the remote application controller. The parameter memory_address shall specify the 20 bit start address (4 bit AddressExtension + 8 bit Address High + 8 bit Address Low) and the parameter number shall contain the number of octets that shall be written beginning with the start address to increasing addresses.

The service shall be a confirmed service if verify mode is active, otherwise it shall be an acknowledged service.

The local application layer shall accept the service request and pass it with a T_Data_Connected.req to the local transport layer. The parameters ASAP and priority shall be mapped to the corresponding parameters of the T_Data_Connected.req primitive, the TSDU shall be an A_UserMemory_Write-PDU, see Figure 49.

The remote application layer shall map a T_Data_Connected.ind primitive with TSDU = A_UserMemory_Write-PDU to an A_UserMemory_Write.ind primitive. The arguments TSAP and priority shall be mapped to the corresponding arguments ASAP and priority of the A_UserMemory_Write.ind primitive.