

INTERNATIONAL STANDARD



Electricity metering data exchange – The DLMS/COSEM suite – Part 3-1: Use of local area networks on twisted pair with carrier signalling



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2021 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

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

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

About the IEC

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

About IEC publications

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

IEC publications search - webstore.iec.ch/advsearchform

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 It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

IEC Customer Service Centre - webstore.iec.ch/csc

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

IEC online collection - oc.iec.ch

Discover our powerful search engine and read freely all the publications previews. With a subscription you will always have access to up to date content tailored to your needs.

Electropedia - www.electropedia.org

The world's leading online dictionary on electrotechnology, containing more than 22 000 terminological entries in English and French, with equivalent terms in 18 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.



IEC 62056-3-1

Edition 2.0 2021-07
REDLINE VERSION

INTERNATIONAL STANDARD



**Electricity metering data exchange – The DLMS/COSEM suite –
Part 3-1: Use of local area networks on twisted pair with carrier signalling**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 17.220.20; 35.110; 91.140.50

ISBN 978-2-8322-1002-3

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

CONTENTS

FOREWORD	6
1 Scope	8
2 Normative references	8
3 Terms, definitions and abbreviated terms	9
3.1 Terms and definitions	9
3.2 Abbreviated terms	9
4 General description	10
4.1 Basic vocabulary	10
4.2 Profiles, layers and protocols	11
4.3 Specification language	12
4.4 Communication services for local bus data exchange without DLMS	12
4.5 Communication services for local bus data exchange with DLMS	21
4.6 Systems management	22
5 Local bus data exchange without DLMS	23
5.1 Physical layer	23
5.2 Data Link layer	35
5.3 Application layer	43
6 Local bus data exchange with DLMS	46
6.1 Physical layer	46
6.2 Data Link layer	47
6.3 Application layer	56
7 Local bus data exchange with DLMS/COSEM	56
7.1 Model	56
7.2 Physical Layer	56
7.3 Data Link layer	68
7.4 Support Manager layer	77
7.5 Transport Layer	81
7.6 Application Layer	85
8 Local bus data exchange – Hardware	86
8.1 General	86
8.2 General characteristics	86
8.3 Bus specification	91
8.4 Magnetic plug	93
8.5 Functional specifications of Primary Station transmitter (for 50 kHz signal)	95
8.6 Functional specifications of Primary Station receiver (for 50 kHz signal)	96
8.7 Functional specification of Secondary Station transmitter (for 50 kHz signal)	97
8.8 Functional specifications of Secondary Station receiver (for 50 kHz signal)	98
9 Unidirectional local data transmission interface	99
9.1 Introduction	99
9.2 General description	99
9.3 Historical TIC	99
9.4 Standard TIC	103
9.5 Unidirectional TIC Hardware	104
Annex A (normative) Specification language	111
A.1 Vocabulary and operating rules	111

A.2 Entity and Entity Invocation.....	112
Annex B (normative) Timing types and characteristics	113
B.1 Timing type definition.....	113
B.2 Timing measurements and characteristics.....	114
Annex C (normative) List of fatal errors.....	115
Annex D (normative) Coding the command code field of frames.....	116
D.1 Command codes for local bus data exchange (Table D.1).....	116
D.2 Codes of commands for data exchange on the local bus with DLMS or DLMS/COSEM	116
Annex E (normative) Principle of the CRC	118
E.1 General.....	118
E.2 Operations on the polynomials.....	118
E.3 Check procedure.....	118
E.4 Operating parameters	119
Annex F (normative) Random integer generation for response from forgotten stations	120
F.1 General.....	120
F.2 Criterion for a random integer	120
F.3 Operating parameters	120
Annex G (normative) Random number generation for authentication (profile without DLMS)	121
Annex H (normative) Systems management implementation	122
Annex I (informative) Information about exchanges.....	123
I.1 Non-energized station session (Figure I.1).....	123
I.2 Remote reading and programming exchanges (Figure I.2)	124
I.3 Bus initialization frame (Figure I.3).....	125
I.4 Forgotten station call exchange (Figure I.4)	126
Bibliography.....	127
Figure 1 – IEC 62056-3-1 communication profiles	11
Figure 2 – Alarm mechanism	21
Figure 3 – Exchanges in continuous operation.....	25
Figure 4 – Alarm event without any communication in progress	26
Figure 5 – Alarm event with a communication in progress.....	26
Figure 6 – Signal envelope on the bus	87
Figure 7 – Bus representation.....	88
Figure 8 – Power supply characteristics.....	88
Figure 9 – States associated to a session: for selected Secondary Station	89
Figure 10 – States associated to a session: for non-selected Secondary Station.....	89
Figure 11 – Simple and multiple Secondary stations	90
Figure 12 – Equivalent diagram of the test equipment.....	92
Figure 13 – Ferrite pot and bobbin	93
Figure 14 – Associated components of the magnetic plug	94
Figure 15 – Associated components of the energy supply plug	95
Figure 16 – Character transmission	100
Figure 17 – Historical TIC: information group structure	101
Figure 18 – Standard TIC: Application information group structure.....	103

Figure 19 – Standard TIC: Timestamped information group structure	103
Figure 20 – Equivalent diagram of the test equipment	107
Figure 21 – Signal envelope on the bus	108
Figure B.1 – Logical timing type	113
Figure B.2 – Physical timing type	113
Figure B.3 – Results processing for timing defined with low and high limits	114
Figure B.4 – Results processing for timing defined by a nominal value	114
Figure I.1 – Non-energized station session	123
Figure I.2 – Remote reading and programming exchanges	124
Figure I.3 – Bus initialization	125
Figure I.4 – Forgotten station call exchange	126
Table 1 – Primary Station timing	24
Table 2 – Secondary Station timing	25
Table 3 – Physical services and service primitives	26
Table 4 – <i>Physical-62056-3-1</i> state transitions: Primary Station	27
Table 5 – Power supply management state transitions (only for non-energized Secondary Station)	30
Table 6 – <i>Physical-62056-3-1</i> state transitions: Secondary Station	31
Table 7 – Meaning of the states listed in the previous tables	32
Table 8 – Definition of the procedures, functions and events classified in alphabetical order	33
Table 9 – Error summary table	35
Table 10 – Data Link services and service primitives	36
Table 11 – <i>Link-62056-3-1</i> state transitions: Primary Station	37
Table 12 – <i>Link-62056-3-1</i> State transitions: Secondary Station	40
Table 13 – Meaning of the states listed in the previous tables	41
Table 14 – Definition of the procedures and functions classified in alphabetical order	41
Table 15 – Error summary table	42
Table 16 – Application services and service primitives	43
Table 17 – <i>Application-62056-3-1</i> state transitions: Primary Station	44
Table 18 – <i>Application-62056-3-1</i> state transitions: Secondary Station	45
Table 19 – Meaning of the states listed in the previous tables	45
Table 20 – Definition of the procedures and functions classified in alphabetical order	46
Table 21 – Error summary table	46
Table 22 – Data Link services and service primitives	48
Table 23 – <i>Link-E/D</i> state transitions: Primary Station	49
Table 24 – <i>Link-E/D</i> state transitions: Secondary Station	51
Table 25 – Meaning of the states listed in the previous tables	53
Table 26 – Definition of the procedures and functions classified in alphabetical order	54
Table 27 – Error summary table	55
Table 28 – Client_connect function definition	56
Table 29 – E/COSEM Physical services and service primitives	57
Table 30 – <i>E/COSEM Physical</i> state transitions: Primary Station	59

Table 31 – Power supply management state transitions (only for non-energized Secondary Station)	61
Table 32 – <i>E/COSEM Physical</i> State transitions: Secondary Station	63
Table 33 – Meaning of the states listed in the previous tables.....	64
Table 34 – Definition of the procedures, functions and events classified in alphabetical order.....	65
Table 35 – Error summary table	68
Table 36 – Data Link services and service primitives	69
Table 37 – <i>DLMS/COSEM Data Link E/D</i> state transitions: Primary Station.....	71
Table 38 – <i>DLMS/COSEM Link E/D</i> state transitions: Secondary Station	73
Table 39 – Meaning of the states listed in the previous tables.....	75
Table 40 – Definition of the procedures and functions classified in alphabetical order.....	76
Table 41 – Commands managed by the Support Manager layer	77
Table 42 – List of parameters	79
Table 43 – Support Manager layer state transitions: Primary Station.....	79
Table 44 – Support Manager layer state transitions: Secondary Station	80
Table 45 – Meaning of the states listed in the previous table	80
Table 46 – Definition of procedures, functions and events	80
Table 47 – Transport services and services primitive	82
Table 48 – Transport state transitions	82
Table 49 – Meaning of the states listed in the previous table	84
Table 50 – Definition of the procedures and functions classified in alphabetical order.....	84
Table 51 – Primary station transmitter: Tev0 and Tev1 values	96
Table 52 – Primary station receiver: Tev0 and Tev1 values	96
Table 53 – Secondary station transmitter: Tev0 and Tev1 values.....	97
Table 54 – Secondary station receiver: Tev0 and Tev1 values	98
Table 55 – TIC terminal board pin out.....	105
Table 56 – Power supply characteristics	105
Table 57 – Signal characteristics	107
Table C.1 – FatalError error numbers	115
Table D.1 – Command codes for local bus data exchange	116
Table D.2 – Command codes with DLMS and DLMS/COSEM.....	117
Table H.1 – Discovery service	122
Table H.2 – Service specification.....	122

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**ELECTRICITY METERING DATA EXCHANGE –
THE DLMS/COSEM SUITE –****Part 3-1: Use of local area networks on twisted pair
with carrier signalling**

FOREWORD

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

This redline version of the official IEC Standard allows the user to identify the changes made to the previous edition IEC 62056-3-1:2013. A vertical bar appears in the margin wherever a change has been made. Additions are in green text, deletions are in strikethrough red text.

International Standard IEC 62056-3-1 has been prepared by IEC technical committee 13: Electrical energy measurement and control.

This second edition cancels and replaces the first edition of IEC 62056-3-1, issued in 2013, and constitutes a technical revision.

The main technical changes with regard to the previous edition are as follows:

- addition of a profile which makes use of the IEC 62056 DLMS/COSEM Application layer and COSEM object model;
- review of the data link layer which is split into two parts:
 - a pure Data Link layer;
 - a "Support Manager" entity managing the communication media;
- ability to negotiate the communication speed, bringing baud rate up to 9 600 bauds.

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

CDV	Report on voting
13/1794/CDV	13/1823/RVC

Full information on the voting for its approval can be found in the report on voting indicated in the above table.

The language used for the development of this International Standard is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available at www.iec.ch/members_experts/refdocs. The main document types developed by IEC are described in greater detail at www.iec.ch/standardsdev/publications.

A list of all parts of IEC 62056 series, published under the general title *Electricity metering data exchange – The DLMS/COSEM suite*, can be found on the IEC website.

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

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

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

ELECTRICITY METERING DATA EXCHANGE – THE DLMS/COSEM SUITE –

Part 3-1: Use of local area networks on twisted pair with carrier signalling

1 Scope

This part of IEC 62056 describes ~~three profiles for~~ two sets of profiles: the first set of profiles allows a bidirectional communication between a client and a server. This set of profiles is made of three profiles allowing local bus data exchange with stations either energized or not. For non-energized stations, the bus supplies energy for data exchange. Three different profiles are supported:

- base profile: this three-layer profile provides remote communication services;
NOTE 1 This first profile was published in IEC 61142:1993 and became known as the Euridis standard.
- profile with DLMS: this profile allows using DLMS services as specified in IEC 61334-4-41;
NOTE 2 This second profile was published in IEC 62056-31:1999.
- profile with DLMS/COSEM: this profile allows using the DLMS/COSEM Application layer and the COSEM object model as specified in IEC 62056-5-3 and in IEC 62056-6-2 respectively.

The three profiles use the same physical layer and they are fully compatible, meaning that devices implementing any of these profiles can be operated on the same bus. The transmission medium is twisted pair using carrier signalling and it is known as the Euridis Bus.

The second set of profiles allows unidirectional communication between a given Energy Metering device and a Customer Energy Management System. This second set is made up of three profiles.

Subclause 4.2.1 to Clause 8 included specify the bidirectional communication using twisted pair signalling and Clause 9 to 9.5 the unidirectional communication using twisted pair signalling.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61334-4-41:1996, *Distribution automation using distribution line carrier systems – Part 4: Data communication protocols – Section 41: Application protocols – Distribution line message specification*

IEC 62056-51:1998, *Electricity metering – Data exchange for meter reading, tariff and load control – Part 51: Application layer protocols*

IEC 62056-5-3:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 5-3: DLMS/COSEM application layer*

IEC 62056-6-2:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 6-2: COSEM interface classes*

ISO/IEC 8482:1993, *Information technology – Telecommunications and information exchange between systems – Twisted pair multipoint interconnections*

EIA 485, *Standard for Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems*

3 Terms, definitions and abbreviated terms

3.1 Terms and definitions

No terms and definitions are listed in this document.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

3.2 Abbreviated terms

ADP	Primary Station Address
ADG	General Secondary Address. Broadcast Address
ADS	Secondary Station Address
AGN	Normal Wakeup
AGT	General call for a General Energized Station
APDU	Application Protocol Data Unit
APG	General Primary Address
ARJ	COM field value: Rejection of authentication in remote programming exchange
ASDU	Application Service Data Unit
ASO	COM field value: Call to Forgotten Stations
AUT	COM field value: Authentication command
COM	Control field of the Data Link layer
COSEM	Companion Specification for Energy Metering
DAT	COM field value: Response of remote reading exchange
DES	Data Encryption Standard
DLMS	Distribution Line Message Specification (IEC 61334-4-41) Device Language Message Specification (IEC 62056-5-3)
DSDU	Data link Service Data Unit
DRJ	COM field value: Data Rejected Value of COM notifying the rejection of remote programming exchange data
Dsap	Transport data unit label. Coded over 3 bits. Its value is 6.
DTSAP	Destination of Transport Service Access Point
ECH	COM field value: Echo of remote programming exchange data
ENQ	Remote reading exchange request
EOS	COM field value: End of remote programming exchange
IB	Initialisation of the bus
LDTI	Local Data Transmission Interface
MaxRetry	Maximum number retransmissions. Limited to 2.
MaxRSO	Maximum number of RSO listening windows. Fixed at 3.

PDU	Protocol Data Unit
PRE	COM field value: Pre-selection of energised stations
REC	COM field value: Remote programming exchange request
RSO	COM field value: Response to a call to forgotten stations
SEL	COM field value: Acknowledgement of the pre-selection of energized stations
STSAP	Source Transport Service Access Point
TAB	In the case of the EURIDIS profiles without DLMS and without DLMS/COSEM: data code. In the case of profiles using DLMS or DLMS/COSEM: value at which the equipment is programmed for Discovery
TABi	List of TAB field
TASB	Duration of an Alarm Signal on the Bus
TIC	Transmission of Information to the Customer
TOAG	Maximum wait time for an energized station once selected, to recognise a general call AGN
TOALR	Wait before sending an AGN after reception of an AGN or AGT
TOL	Maximum waiting time for a request from the upper layer
TOPRE	Maximum waiting time for a response to a pre-selection.
TOU	Time of Use
TPDU	Transport Protocol Data Unit
TSDU	Transport Protocol Service Unit
TRA	COM field value: Acknowledgement of point to point transfer
TRB	COM field value: Broadcast remote transfer frame not acknowledged
TRF	COM field value: Point to point remote transfer exchange
T1	Time out to wait for a response according to a request
XBA	COM field value: Response to a change of speed request
XBR	COM field value: Change of speed request
ZA1	Field reserved for bidirectional programming authentication
ZA2	Field reserved for bidirectional programming authentication

4 General description

4.1 Basic vocabulary

All communication calls upon two systems called Primary Station and Secondary Station. The Primary Station is the system that decides to initialize a communication with a remote system called Secondary Station; these designations remain valid throughout the duration of the communication.

A communication is broken down into a certain number of transactions. Each transaction consists of a transmission from the Transmitter to the Receiver. During the sequence of transactions, the Primary Station and Secondary Station systems take turns to act as Transmitter and Receiver.

For the local bus data exchange profile with DLMS or DLMS/COSEM, the terms Client and Server have the same meaning as for the DLMS model (refer to IEC 61334-4-41 or IEC 62056-5-3). The Server (which is a Secondary Station) receives and processes all submissions of specific service requests. The Client (which is a Primary Station) is the system that uses the Server for a specific purpose by means of one or more service requests.

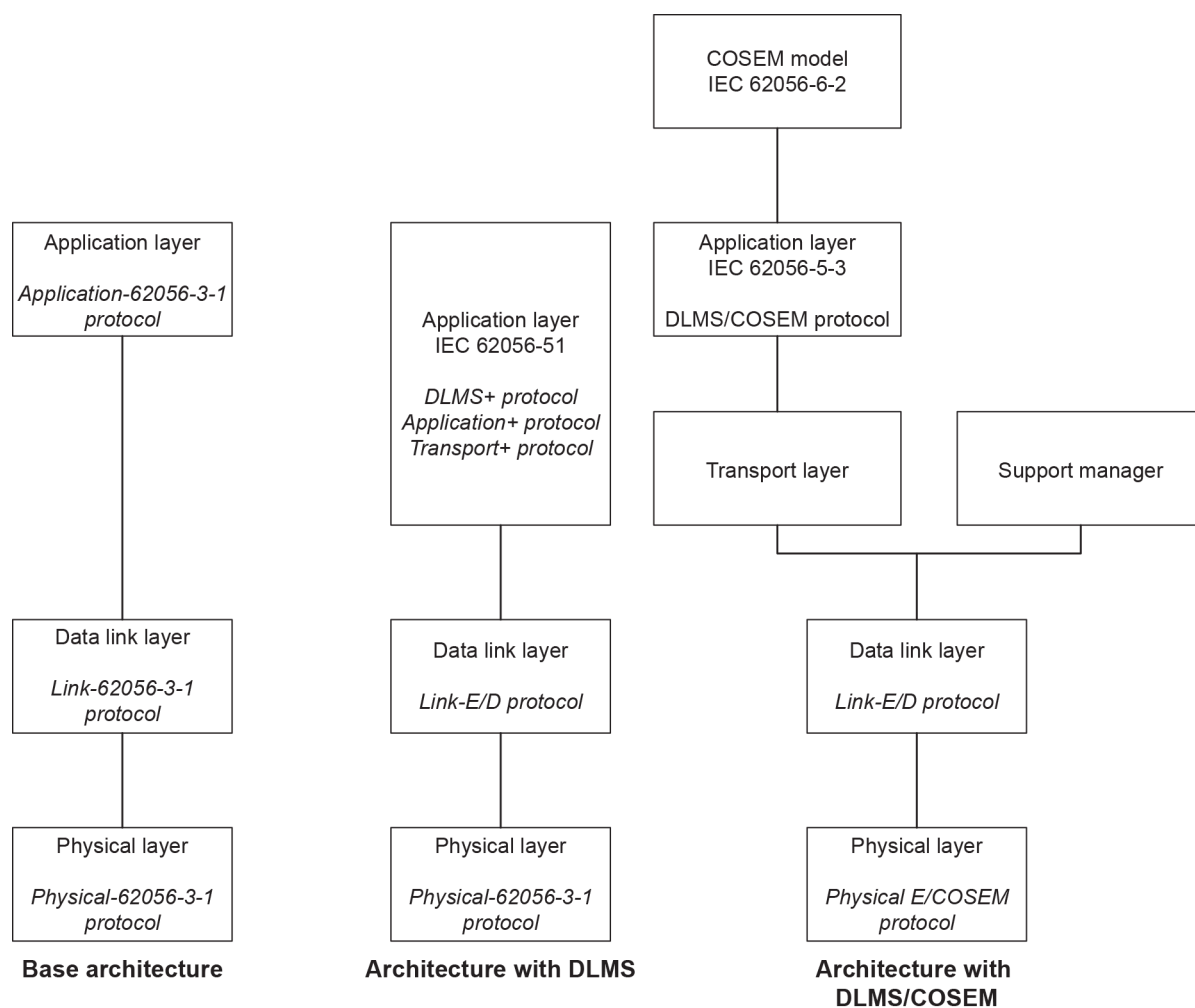
4.2 Profiles, layers and protocols

4.2.1 Overview

This document specifies three profiles as shown in Figure 1.

- the base profile (without DLMS), see 4.2.2;
- the profile with DLMS, see 4.2.3;
- the profile with DLMS/COSEM; see 4.2.4.

The physical layer in the three profiles is the same except that in the DLMS/COSEM profile speed negotiation is available. This common physical layer allows stations using different profiles to be installed on the same bus.



IEC

Figure 1 – IEC 62056-3-1 communication profiles

4.2.2 Base profile (without DLMS)

The base profile (without DLMS) uses three protocol layers:

- the physical layer with the *Physical-62056-3-1* protocol specified in 5.1;
- the data link layer with the *Link-62056-3-1* protocol, specified in 5.2, and
- the application layer with the *Application-62056-3-1* protocol specified in 5.3.

This profile allows remote reading, remote programming, point-to-point remote transfer – which is a simplified remote programming service – broadcast remote transfer, remote supply of secondary stations, detecting forgotten stations and alarm functions. The related communication services are specified in 4.4.

4.2.3 Profile with DLMS

The profile with DLMS uses three protocol layers:

- the same physical layer as the base profile, specified in 5.1;
- the data link layer using the *Link-E/D* protocol, specified in 6.2; and
- the application layer specified in IEC 62056-51, using the *Transport+*, *Application+* and *DLMS+* protocols, see 6.3.

This profile also allows using DLMS as specified in IEC 61334-4-41. The related communication services are specified in 4.5.

4.2.4 Profile with DLMS/COSEM

The profile with DLMS/COSEM uses four protocol layers:

- the physical layer, similar to the one used in the base profile and the profile with DLMS, specified in 5.1, but with speed negotiation, see 7.2;
- the data link layer using the *Link-E/D* protocol. This is the same as the data link layer of the profile with DLMS, except that it interfaces with the support manager layer and the transport layer. See 7.3;
- the support manager layer supports some specific process for the management of the bus, see 7.4;
- the transport layer provides segmentation and reassembly of APDUs, see 7.5;
- the application layer as specified in IEC 62056-5-3 taking into account some restrictions of the Euridis bus, see 7.6.

The profile with DLMS/COSEM allows using the COSEM object model and the DLMS services accessing the COSEM objects over the Euridis bus.

4.3 Specification language

In this document, the protocol of each layer is described by state transitions represented in the form of tables. The syntax used in making up these tables is defined by a specification language described in Annex A.

In the event of a difference in interpretation between part of the text and a state transition table, the table is always taken as the reference.

4.4 Communication services for local bus data exchange without DLMS

4.4.1 Overview

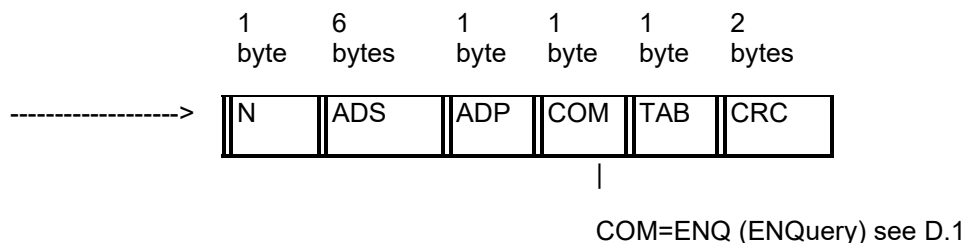
The list of available services (see Annex I) at the Application level layer is:

- a) remote reading of data, see 4.4.2;
- b) remote programming of data, see 4.4.3;
- c) point to point remote transfer, which is a simplified remote programming service, see 4.4.4;
- d) broadcast remote transfer, 4.4.5;
- e) bus initialization, 4.4.6;
- f) forgotten station call, 4.4.7.

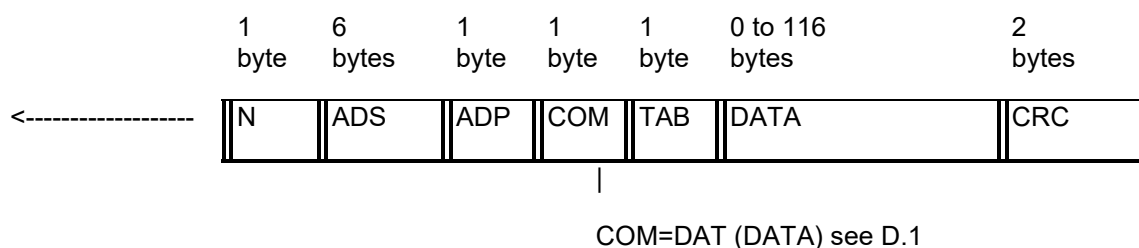
4.4.2 Remote reading exchange

The ENQ exchange consists of two frames arranged in one sequence:

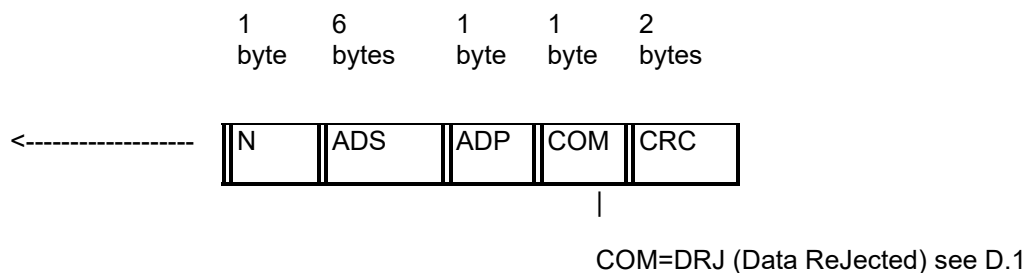
- remote reading frame containing the type of data to select in the TAB field



- positive acknowledgement frame with the selected data in the DATA field



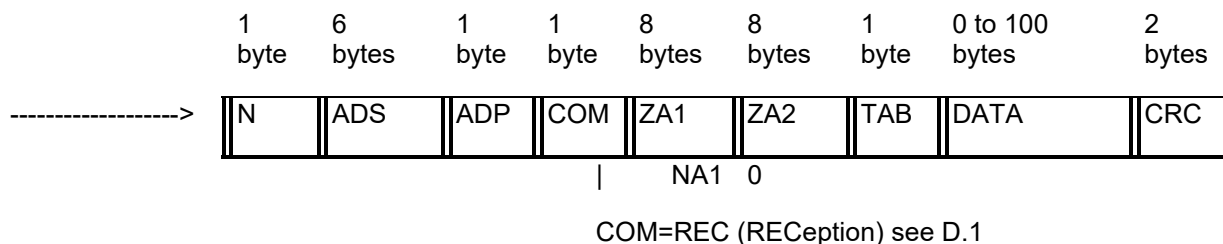
- negative acknowledgement frame (TAB identifier unknown)



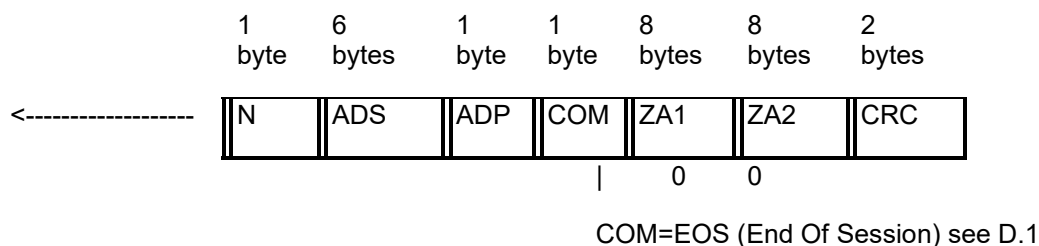
4.4.3 Remote programming exchange

The REC exchange consists of four frames arranged in two sequences. Since there is an internal sequence for authentication purpose, from the application point of view, it seems to be only one sequence with two frames:

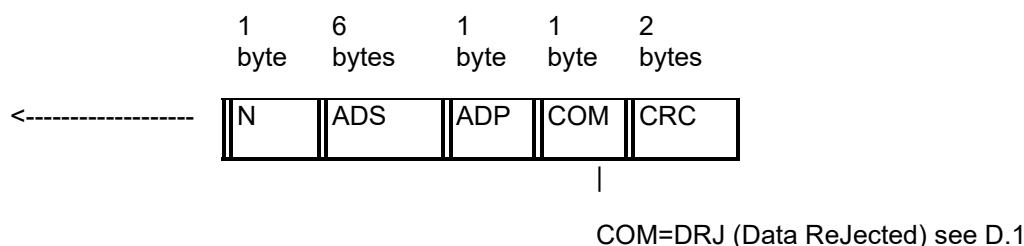
- remote programming frame containing data in the DATA field and their type in the TAB field



- positive acknowledgement frame (no authentication trouble)



- negative acknowledgement frame (no authentication trouble but remote programming data not validated)



Authentication is carried out by an exchange of random numbers ciphered using a secret key specific to each Secondary Station. The random numbers are defined in 8 bytes and they are ciphered with the DES algorithm using an 8-byte ciphering key K_i known both to the Primary and the Secondary station.

A random number NA1 is first generated by the Primary Station and transmitted into the ZA1 field of the remote programming frame while field ZA2 is set to zero.

On arrival at the Secondary Station, field ZA1 is ciphered by the DES algorithm with key K_i to get the ciphered random number NA1K. Then occurs the internal sequence for authentication which consists of two frames.

The first frame (from Secondary to Primary Station) contains this random number NA1K in field ZA1 and a random number NA2 generated by the Secondary station in field ZA2.

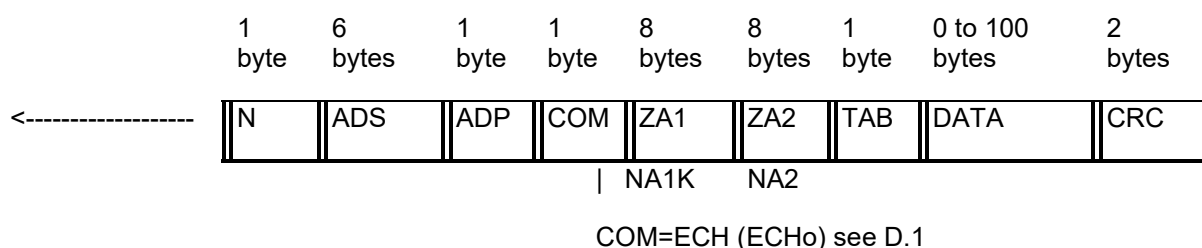
On reception of this frame, the Primary Station compares the ZA1 field to an NA1' number obtained by ciphering the transmitted NA1 number using the DES algorithm with key K_i . If $NA1' = ZA1$, then the Primary Station considers the called Secondary Station as authenticated. Otherwise, it considers the Secondary Station has not been authenticated and aborts the communication session.

After correct authentication of the Secondary Station, the Primary Station first ciphers the random number NA2 by the DES algorithm with key K_i to get the ciphered random number NA2K and then transmits it into field ZA2 while field ZA1 is set to zero.

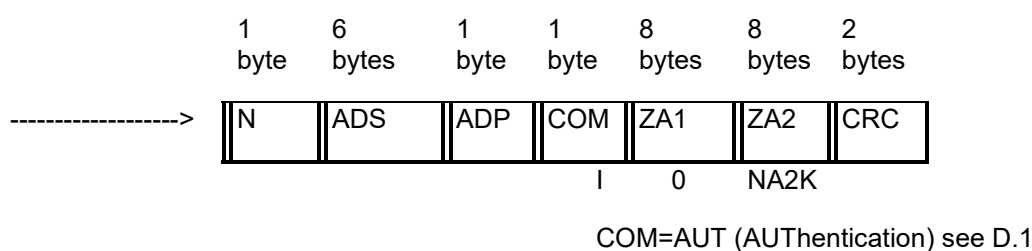
On reception of this response frame, the Secondary Station compares the ZA2 field to an NA2' number obtained by ciphering the transmitted NA2 number using the DES algorithm with key K_i . If $NA2' = ZA2$, then the Secondary Station considers the Primary Station as authenticated. Otherwise, it considers the Primary Station has not been authenticated and sends a negative acknowledgment frame.

The internal authentication exchange is the following:

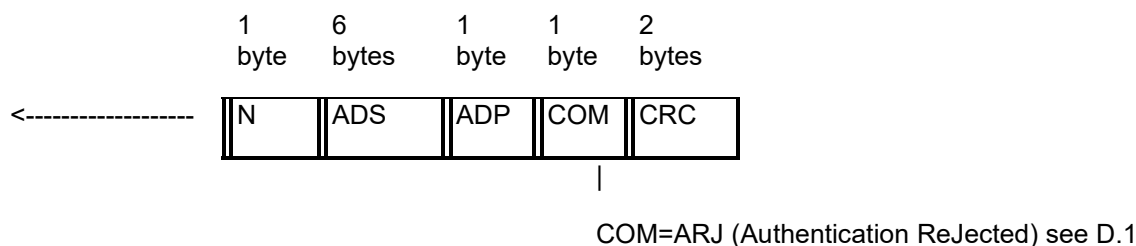
- internal authentication frame containing the ciphered random number NA1K in field ZA1 and the random number NA2 in field ZA2



- positive response frame containing the ciphered random number NA2K in field ZA2 (if the Secondary Station is considered as authenticated)



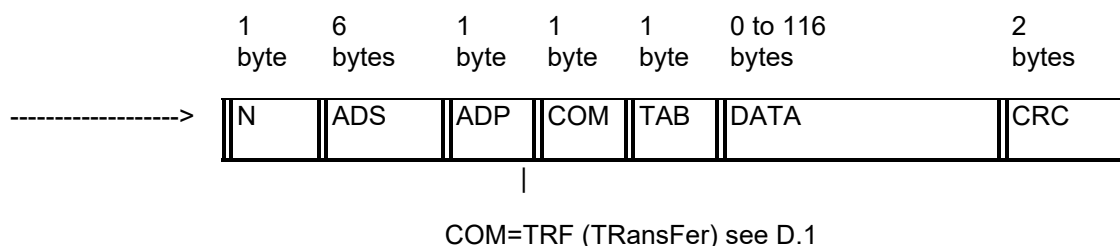
- an authentication rejection frame replaces the normal EOS or DRJ frame when the Primary Station is not considered authenticated



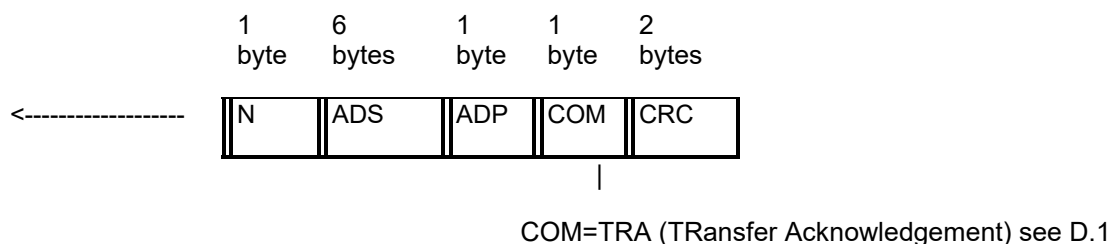
4.4.4 Point to point remote transfer exchange

This TRF exchange consists of two frames arranged in one sequence. From the application point of view, it seems to be a remote programming exchange in a single sequence with no authentication:

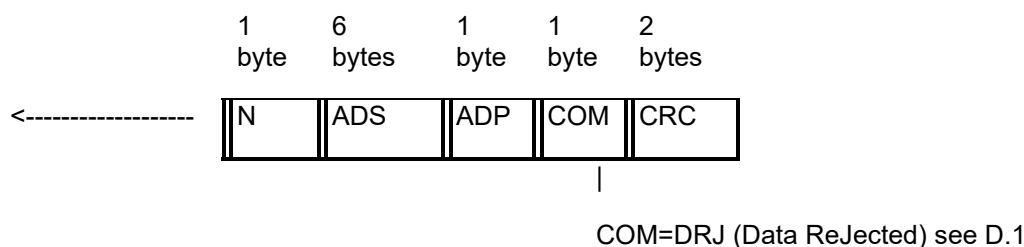
- point to point remote transfer frame containing data in the DATA field and their type in the TAB field



- positive acknowledgement frame



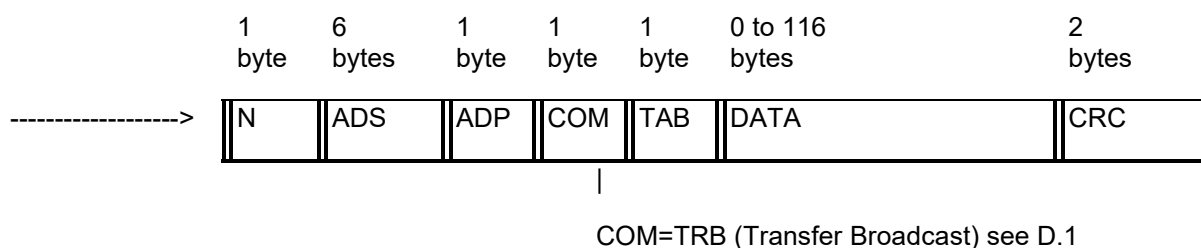
- negative acknowledgement frame (remote transfer data not validated)



4.4.5 Broadcast remote transfer frame

This TRB frame does not involve any frame answer. From the application point of view, it seems to be a point to point remote transfer, but without acknowledgement since it is a broadcast.

- broadcast remote transfer frame containing data in the DATA field and their type in the TAB field

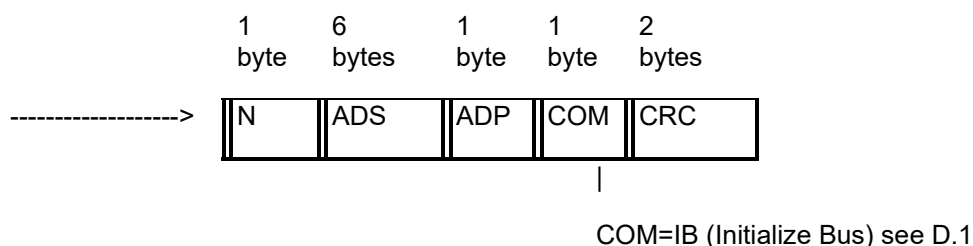


The secondary address (which defines the receiving Secondary Stations) shall be a broadcast address.

4.4.6 Bus initialization frame

This IB frame does not involve any frame answer. From the application point of view, it seems to be a broadcast remote transfer, but without data since its purpose is only to reset a special flag (called forgotten station flag) to TRUE for all Secondary Stations that have been programmed with the ADP address:

- bus initialization frame



The secondary address (which defines the receiving Secondary Stations) shall be a broadcast address.

After the bus initialization frame, any Secondary Station receiving a correct ENQ frame containing a known TAB identifier will then no longer be considered as a "forgotten station".

4.4.7 Forgotten station call exchange

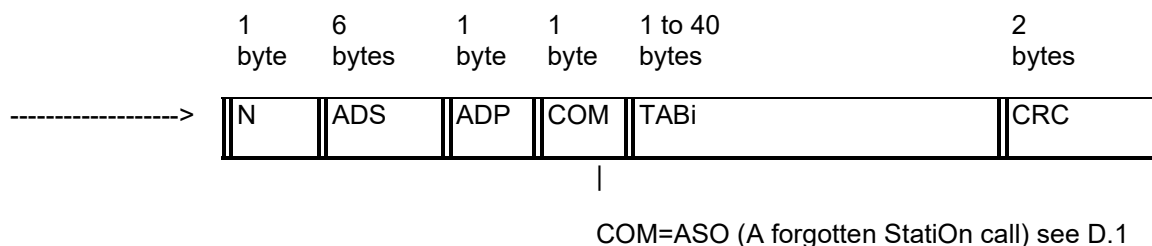
This ASO exchange consists of two frames arranged in one sequence. At the end of a remote reading sequence, the Primary Station can search for stations whose forgotten station flag is TRUE (maximum 5 in 100).

As a correct remote reading exchange sets the forgotten station flag of the corresponding station to FALSE, the ASO exchange normally occurs after the completion of a remote reading sequence that is one or several remote reading exchanges preceded by a bus initialization frame.

The Primary Station manages several time slots. When detecting a collision, it has to retry an ASO exchange. Nevertheless, each time a correct Secondary Station answer is received, the Primary Station shall eliminate it from the list of forgotten stations by operating a correct remote reading exchange with this station.

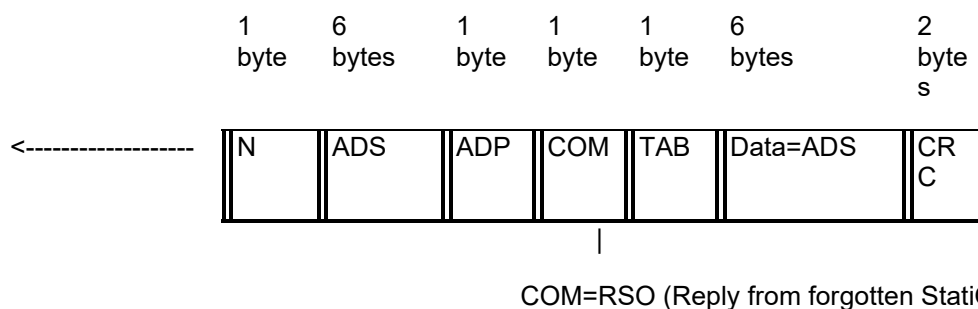
In order to ensure the selection constraints (described in 4.4.9), the non-energized stations shall answer in the first time slot of the first ASO exchange. Then, only the forgotten stations are selected and the usual principle can be used for the following ASO exchanges.

- forgotten station call frame containing selection criteria in the TAB_i field (1 to 40 TAB identifiers)



The secondary address (which defines the receiving Secondary Stations) should be a broadcast address.

- acknowledgement frame containing the first TAB recognized by the unit and the ADS of the station



- The data field containing the ADS of the secondary station responds to the call to forgotten stations.

4.4.8 Frame fields

N	total number of bytes in the frame, including N.
ADS	absolute physical address of the Secondary Station coded as a 48-bit string. There is only one broadcast physical address which is the general broadcast ADG coded as "000000000000" in hexadecimal ¹ . The ADS also corresponds exactly to the System Title of the Secondary Station.
ADP	physical address of the Primary Station coded as an 8-bit string. The value "00"H is reserved for the coding of the physical address of the general primary APG ² . Any Secondary Station solicited by a Primary Station whose physical address is APG, replies with the first primary physical address with which it has been programmed.
COM	command code depending on the exchange and the frame direction (see Annex D).
ZA1, ZA2	fields reserved for authentication operated during the remote programming exchange.
TAB	type of data selected associated with some command codes (ENQ, DAT, REC, TRF, TRB or RSO). The value "00"H is reserved for systems management, the value "FF"H for alarm management.
DATA	information packet from the host application. This field can be eventually empty depending on the command code.
CRC	Cyclic Redundancy Check field corresponds to the 16 redundant bits of the CRC whose principle is described in Annex E.

The frame fields are transmitted in an ascending order (from N to CRC). When a field contains data over several bytes, the transmission begins with the least significant byte and ends with the most significant one. However, the DATA field is considered as a byte string and is transmitted in an ascending order.

4.4.9 Principle of the energy remote supply

The general principle of the data exchanges is preserved for the non-energized stations. The notion of energy remote supply is only added for communication between a Primary Station and one or more Secondary Stations.

To begin a communication session, the Primary Station ~~shall~~ **should** send a "Wakeup Call" designed to alert the communications system of every Secondary Station connected to the bus. This call is a continuous carrier for a nominal time depending on the energy remote supply mechanism:

- the "Wakeup Call" signal duration is AGT to wake up non-energized stations;
- the "Wakeup Call" signal duration is AGN to wake up energized stations.

Remark: A Secondary Station can be configured in Alarm mode. It is then remote supplied continuously and so can transmit the alarm to the Primary Station (see 4.4.11).

Then, whatever type of remote station is selected (energized or not), an intermediate AGN "Wakeup Call" signal shall also be required at the Primary Station side in the following circumstances:

- before the first ENQ or TRF exchange;

¹ Other broadcast addresses could be defined depending on the naming rules adopted in companion standards for the semantics of the System Titles which are often based on a manufacturer code, a manufacture year and an equipment type.

² Other general addresses could be defined depending on the naming rules adopted in companion standards for the semantics of operator identifiers which are often based on a utility code.

- before the sixth consecutive and successful ENQ or TRF exchange with the same Secondary Station;
- before the first ENQ or TRF exchange with a different Secondary Station to the one previously selected in the preceding ENQ or TRF exchange;
- before any REC exchange;
- before any TRB frame;
- before any IB frame;
- before any ASO exchange.

For non-energized stations, it means that the Primary Station can avoid to wake up all the remote stations when not necessary, and then save its energy.

A Primary Station can use a specific modem ensuring both the energy remote supply as well as the modulation and demodulation functions. The communication time and the number of non-energized stations shall be optimized in order to save the battery of the Primary Station.

As another possibility, the Primary Station might only focus on the modulation and demodulation functions. In this case, an auxiliary station continuously supplies the bus with energy.

A Secondary Station generally contains only one logical application referenced by its ADS. Such a station may or may not be energized.

A multiple Secondary Station (containing several logical applications corresponding to several ADSs) should be a non-energized station. This feature is described more fully in Clause 8.

4.4.10 Non-energized station preselection exchange

To optimize the bus consumption, a preselection exchange enables the Primary Station to select a non-energized Secondary Station.

The preselection exchange occurs after an AGT "Wakeup Call" signal addressed to all non-energized stations of the bus. To limit the bus consumption, the first frame sent by the Primary Station should be short enough and the addressed Secondary Station should answer before the triggering of the TOPRE wakeup. Not seeing an answer in time, the modem of the Secondary Station goes back in a low consumption state.

During the preselection exchange, all the non-energized stations consume energy. The bus voltage and the energy storage capacitors decrease until the non-selected stations goes back in a low consumption state. Then the continuously sent energy charges the energy storage capacitors and the bus voltage increases.

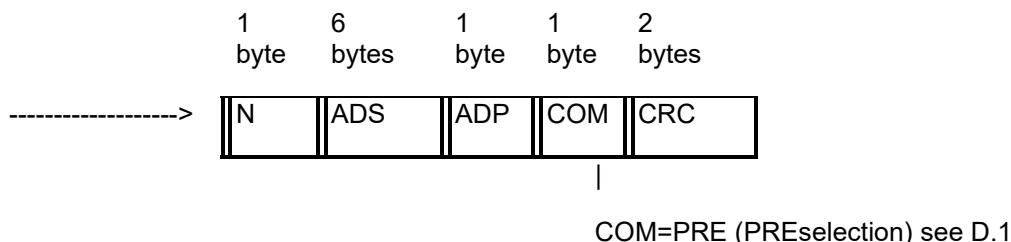
The modem of the Primary Station should store sufficient energy before the first preselection. This step is guaranteed by a wait-time controlled thanks to the TICB wakeup. At the end of a preselection, the energy storage capacitors are empty and the Primary Station shall wait for the bus voltage increase before a second preselection.

As the preselection frame shall not be more than 18 bytes long, it can be:

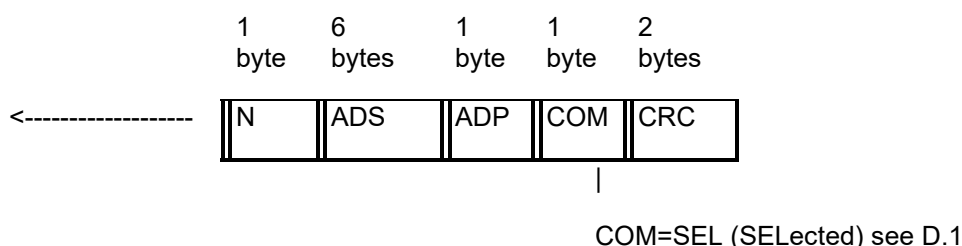
- an ENQ frame;
- a TRB or TRF frame, if and only the data field is less than or equal to 6 bytes long;
- an IB frame;
- an ASO frame, if and only the number of TABi fields is less than or equal to 7.

As the first frame of REC and TRF exchanges may be too long, an additional service is provided for preselection. This fully transparent preselection exchange consists of two frames arranged in one sequence.

- non-energized station preselection frame



- acknowledgement frame



To save the energy of the Primary Station, there is no retry during a preselection exchange. If an addressed non-energized station does not answer correctly, it is not selected and the Primary Station shall send a new AGT "Wakeup Call" signal.

4.4.11 Communication exchange after preselection

After preselection, the modem of a non-energized station can stay awake for the continuing communication and delays are not critical since the number of connected devices is limited. The Primary Station supplies the selected station and charges the capacitive reservoirs of the non-selected stations.

The normal end of the communication session occurs differently depending on the energy remote supply mechanism:

- after a short period of inactivity during the communication session when no intermediate AGN "Wakeup Call" signal is required for energized stations. This period is checked by the wakeup TOL;
- after a longer period of inactivity during the communication session for non-energized stations. This period is checked by the wakeup TOAG.

Note that for a non-energized station, as far as there is no timeout of TOAG wakeup, an intermediate AGN "Wakeup Call" signal is enough to go on the current communication session.

4.4.12 Alarm function

A device integrated in a simple or multiple Secondary Station (see 8.2.3) can transmit alarms to the primary station, providing it can integrate functions of interface as described hereinafter.

An alarm shall be fetched from the Secondary Station in 10 s maximum.

A programmable configuration on the Interface and one on each device selects the status of Alarm mode: Active or Inactive.

When Alarm mode is active, the device can generate an alarm, inside the secondary station. The Alarm function is effective only if the supply is present and permanent on the bus.

The device sends the alarm during TASB. TASB is long enough to force an "0" state on the secondary bus and to be detected by the Interface, even if a communication is in progress.

Alarm mechanism is described in Figure 2.

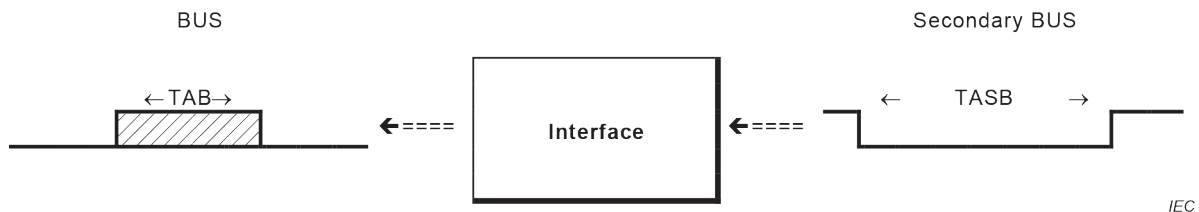


Figure 2 – Alarm mechanism

The alarm is not directly transmitted towards the primary station. The interface receives the alarm and transmits it by sending a "0" (50 kHz carrier) during TAB on the bus when it is possible:

- a) No communication on the bus.
When the interface receives the alarm on the secondary bus, it transmits it on the bus.
- b) On synchronization of AGN or AGT when a communication is in progress. When a communication is in progress on the bus, the Interface memorises the alarm received. It transmits it to the bus after one of the following events:
 - TOALR after the end of AGN or AGT reception;
 - when the normal end of the communication session occurs.

In this way, the interface can filter the alarm to avoid conflict on the bus.

After the alarm generation, the Secondary Station will be considered as a "forgotten station" with a selection criterion equal to FF.

The primary station configured in Alarm mode listens to the bus when there is no communication on the bus and after transmission of an AGN or AGT in order to detect an alarm. When the primary station receives an alarm, it enters in Forgotten Station Call procedure with a selection criteria in the TABi field equal to FF (see 4.4.7).

Timing diagrams explain Alarm management in 5.1.3.

4.5 Communication services for local bus data exchange with DLMS

DLMS does not offer services to operate the bus initialization and forgotten station call mechanisms. Nevertheless, the IB frame and the ASO exchange are supported and managed as they are with the local bus data exchange profile without DLMS except that the forgotten station flag is considered as a global variable shared with the Application Programming Interface.

Remote reading of data and point to point remote transfer are directly foreseen by DLMS. But the (redundant) remote programming of data is not supported since authentication is reserved for the *Application* layer.

As data semantics is managed by DLMS, the frame format is very simple and only unmarked frames are required. To ensure compatibility with the profile without DLMS, this frame format is defined by the following nine fields:

1 byte	6 bytes	1 byte	3 bits	1 bit	2 bits	2 bits	0 to 117 bytes	2 bytes
Size	ADS	ADP	DATA+	Priority	Send	Confirm	Text	CRC

Size total number of bytes in the frame, including Size. If its value is not 11, the Receiver knows that the frame contains data in the Text field

ADS same rules as for local bus profile without DLMS

ADP same rules as for local bus profile without DLMS

DATA+ always coded "111"B

Priority transmission priority level of the current frame. The *Application* layer sets this priority according to the requested service

Send number of the last frame sent

Confirm number of the last frame received without error

Text DSDU (Data link Service Data Unit) from the higher level. A frame does not necessarily contain text. If data from the *Application* layer is available when the frame is sent, then the *Text* field will contain data, otherwise it will be empty. This mechanism provides the conditions for balanced bi-directional data transmission. In order not to confuse DATA+ frame with frames from the profile without DLMS, the DATA+, Priority, Send and Confirm fields make up a special command code COM whose values are different from the already reserved COM values (see Annex D)

CRC same rules as for local bus profile without DLMS

The frame fields are transmitted in ascending order (from Size to CRC). When a field is coded on several bytes, the transmission begins with the least significant byte and ends with the most significant one. However, the Text field is considered as a byte string and transmitted in ascending order.

4.6 Systems management

The purpose of Systems management is to allow an enrolment. This enrolment includes an identification of Secondary Stations on a bus. The Discover service is provided for this purpose.

The enrolment consists of a sequence of Discover requests issued by the active initiator located inside the Primary Station. Each Discover service is provided to inform the remaining new stations that they will have a chance to respond in the next time slots.

A Discover request conveys a specific response-probability argument as an integer in the range [0, 100]. It expresses the probability, in per cent, that a new station responds. When it is set to 100, all the new stations on the bus shall respond.

On reception of a Discover indication, each Secondary Station tests the value of its flag Discovered. If it is set to TRUE, the indication is discarded; otherwise it draws a random number between 1 and 100. If this number is smaller than or equal to the response-probability argument, the new station will issue a Discover response and set its flag Discovered to TRUE.

The flag Discovered is always reset on a receiving of an IB frame.

To ensure a maximum compatibility (for stations including DLMS/COSEM, DLMS or otherwise), it is proposed to implement the systems management as indicated in Annex H.



5 Local bus data exchange without DLMS

5.1 Physical layer

5.1.1 Physical-62056-3-1 protocol

The *Physical-62056-3-1* protocol of the *Physical* layer of the local bus data exchange profile without DLMS behaves asymmetrically. The state machine of the Primary Station is therefore different from that of the Secondary Station.

The *Physical-62056-3-1* protocol supports the Secondary Stations whether or not they are energized. As already stated in the general description, the remote stations are woken up either by an AGN or an AGT "Wakeup Call" signal and a communication session ends after expiry of TOL or TOAG wakeup.

After a "Wakeup Call" signal, a communication session is then made asynchronously and by half-duplex at 1 200 bits/s on the bus.

5.1.2 Physical parameters

The value of the maximum size of a frame being received, *MaxIndex*, is set to 128.

The value of the maximum number of RSO time slots for the processing of a "Forgotten Stations Call", *MaxRSO*, is set to 3.

The AGN duration of an AGN "Wakeup Call" signal shall be in the range [50, 150[ms, while the AGT duration of an AGT "Wakeup Call" signal shall be in the range [200, 300[ms.

Timing type and characteristics are described in Annex B.

The values of Table 1 are defined for a Primary Station.

Table 1 – Primary Station timing

	Min. ms	Nominal ms	Max. ms	See Type B.1	Definition
TA10	–	–	120	TSL1	Maximum waiting period of the first byte of a frame being received
TAB		100		TC	Duration of an alarm signal on the bus
TAGN	–	100	–	TPDF	Duration of an AGN "Wakeup Call" signal
TAO	–	–	40	TC	Maximum waiting period of one byte of a frame being received whose expiry indicates the end of a frame
TARSO	–	500	–	TC	Length of an RSO time slot
TASB		1 200		TC	Waiting period after the beginning of an alarm signal
TEMPO	–	40	–	TC	Safety delay at the end of transmission of a Wake up signal or a frame
TOE	–	–	2 500	TL	Safety delay for transmitting to protect against defective hardware
TOL	–	–	100	TSL2	Maximum waiting time for a request coming from the upper layer
T1	–	10 000	–	TL	Maximum waiting time for a response from the secondary station
Non-energized station specific (Supply)					
TAGT	–	250	–	TPDF	Duration of an AGT "Wakeup Call" signal
TICB	8 000	–	–	Ta	Initial bus charge period
TOAG	–	–	3 000	TPFD	Maximum delay for a selected non-energized station to recognize an AGN "Wakeup Call" signal

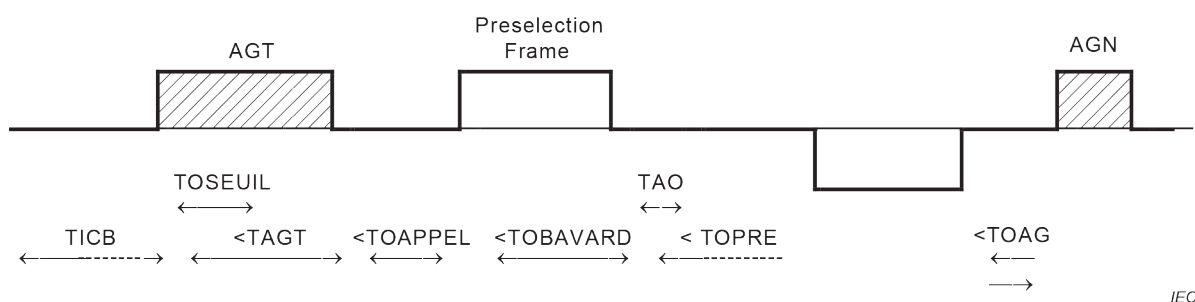
The values of Table 2 are defined for a Secondary Station.

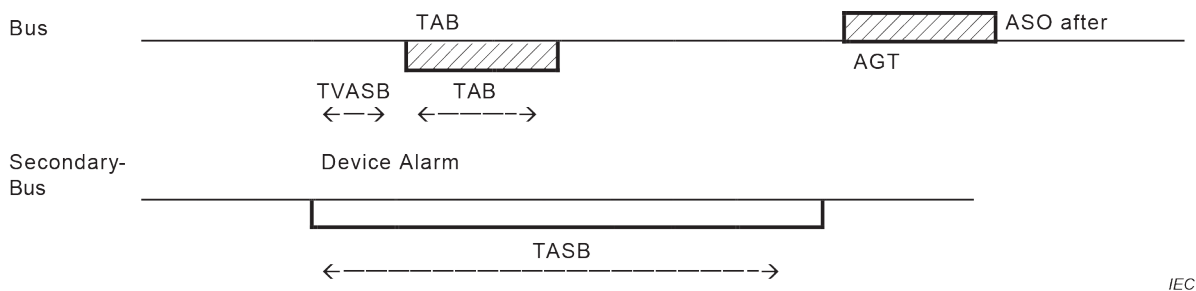
Table 2 – Secondary Station timing

	Min. ms	Nominal ms	Max. ms	Type ^a	Definition
TA10	30 ^b	–	160	TSL1	Maximum waiting period of the first byte of a frame being received
TAB		100	–	TC	Duration of an alarm signal on the bus
TAGN	50	100	150	TPDF	Duration of an AGN "Wakeup Call" signal
TAO	–	–	40	TC	Maximum waiting period of one byte of a frame being received whose expiry indicates the end of a frame
TARSO	–	–	500	TC	Length of an RSO time slot
TOALR	20	–	–	TL	Waiting to send an AGN after an AGN or AGT reception
TOE	–	–	2 500	TL	Safety delay for transmitting to protect against defective hardware
TOL	–	–	100	TSL2	Maximum waiting time for a request coming from the upper layer
Non-energized station specific (Supply)					
TAGT	200	250	300	TPDF	Duration of an AGT "Wakeup Call" signal
TASB	–	1 200	–	TL	Duration of a secondary-bus alarm signal
TICB	8 000	–	–	Ta	Initial bus charge period
TOAG	–	–	3 000	TPFD	Maximum delay for a selected non-energized station to recognize an AGN "Wakeup Call" signal
TOAGN	–	–	300	Tc	Maximum delay of inactivity to recognize the end of a communication session with an energized station
TOAPPEL	–	–	180	TPFD	Maximum waiting period of the first byte of a preselection frame being received
TOBAVAR D	–	–	260	TPDF	Safety delay to protect against defective length of preselection frame
TOPRE	–	–	130	TPFD	Maximum waiting for a preselection answer
TOSEUIL	–	150	–	TC	Duration of a "Wakeup Call" signal which wakes up a non-energized station
TVASB	40	–	–	TL	Minimum duration of an alarm signal on the secondary bus
^a For the definition of different timing types, see Clause B.1. ^b After a "Wakeup Call", a minimum duration of 30 ms is necessary.					

5.1.3 Timing diagrams

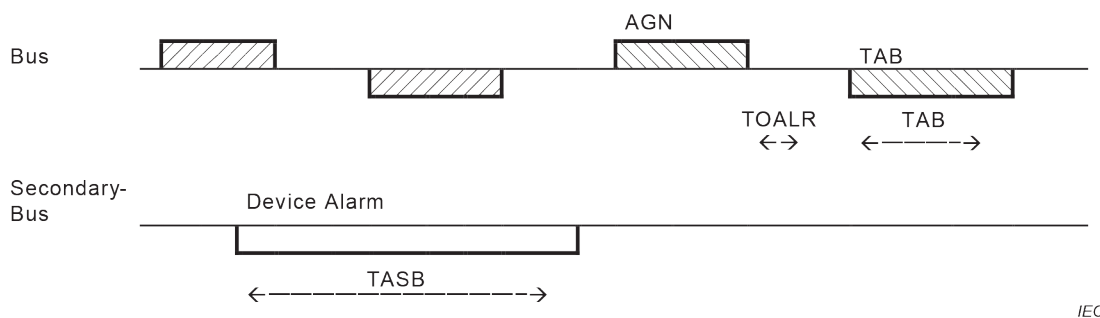
Figure 3, Figure 4 and Figure 5 can be used to show different types of session of the protocol for non-energized secondary stations.

**Figure 3 – Exchanges in continuous operation**



IEC

Figure 4 – Alarm event without any communication in progress



IEC

Figure 5 – Alarm event with a communication in progress

5.1.4 Physical services and service primitives

The user of the *Physical-62056-3-1* protocol can use the services and service primitives given in Table 3.

Table 3 – Physical services and service primitives

Service	Service primitive
Phy_DATA	Phy_DATA.req(Frame) Phy_DATA.ind(Frame)
Phy_UNACK	Phy_UNACK.req(Frame)
Phy_APPG	Phy_APPG.req(TypeAG) Phy_APPG.ind()
Phy_ASO	Phy_ASO.req(Frame) Phy_ASO.ind(Frame)
Phy_RSO	Phy_RSO.req(Frame, Window)
Phy_COLL	Phy_COLL.ind()
Phy_ALARM	Phy_ALARM.req() Phy_ALARM.ind()
Phy_ABORT	Phy_ABORT.req() Phy_ABORT.ind(ErrorNb)

The role assigned to each primitive is as follows:

- Phy_DATA.req(Frame) enables the *Data Link* layer to request the *Physical* layer to transmit a frame Frame;
- Phy_DATA.ind(Frame) enables the *Physical* layer to inform the *Data Link* layer that a frame Frame is available;

- `Phy_UNACK.req(Frame)` enables the *Data Link* layer to request the *Physical* layer to transmit a frame `Frame` without waiting for acknowledgement;
- `Phy_APPG.req(TypeAG)` enables the *Data Link* layer to request the *Physical* layer to transmit a "Wakeup Call" signal. The duration `TypeAG` of this signal is either `AGN` or `AGT`;
- `Phy_APPG.ind()` enables the *Physical* layer to inform the *Data Link* layer of the end of the transmission of a "Wakeup Call" signal;
- `Phy_ASO.req(Frame)` enables the *Data Link* layer to request the *Physical* layer to transmit a "Forgotten Stations Call" frame;
- `Phy_ASO.ind(Frame)` enables the *Physical* layer to inform the *Data Link* layer that a frame `Frame` has been received in one of the time slots of the forgotten stations;
- `Phy_RSO.req(Frame, Window)` enables the *Data Link* layer to request the *Physical* layer to transmit a Forgotten Stations Call frame `Frame` in the time slot number `Window`;
- `Phy_COLL.ind()` enables the *Physical* layer to inform the *Data Link* layer that a collision has been detected in one of the time slots of the forgotten stations;
- `Phy_ALARM.req()` enables the *Data Link* layer to request the *Physical* layer to transmit an Alarm;
- `Phy_ALARM.ind()` enables the *Physical* layer to inform the *Data Link* layer of the arrival of an alarm;
- `Phy_ABORT.req()` enables the *Data Link* layer to request the *Physical* layer to end its activity;
- `Phy_ABORT.ind(ErrorNb)` enables the *Physical* layer to inform the *Data Link* layer of the occurrence of a fatal error identified by the number `ErrorNb`.

5.1.5 State transitions

The *Physical-62056-3-1* state transitions are as specified in Table 4, Table 5, Table 6, Table 7 and Table 8.

Table 4 – *Physical-62056-3-1* state transitions: Primary Station

Initial state	Triggering condition	Set of actions	Final state
<u>Initial</u>	<code>\$true()</code>	<code>MaxRSO=3</code> <code>MaxIndex=128</code> <code>Collision=FALSE</code> <code>SessionAGT=FALSE</code> <code>wait_time(TICB)</code>	Stopped
<u>Stopped</u>	<code>Phy_APPG.req(AG) & AG=AGN</code>	<code>stop_timer(TOAG)</code> <code>FlagAbort=FALSE</code> <code>TypeAG=AGN</code> <code>send_AG(TypeAG)</code>	W.AG
<u>Stopped</u>	<code>Phy_APPG.req(AG) & AG=AGT</code>	<code>SessionAGT=TRUE</code> <code>FlagAbort=FALSE</code> <code>TypeAG=AGT</code> <code>send_AG(TypeAG)</code>	W.AG
<u>Stopped</u>	<code>time_out(TOAG)</code>	<code>Phy_ABORT.ind(EP-2)</code> <code>SessionAGT=FALSE</code>	<u>Stopped</u>
<u>Stopped</u>	<code>Phy_ABORT.req()</code>	<code>\$none()</code>	<u>Stopped</u>
<u>Stopped</u>	<code>data-carrier_on</code>	<code>init_timer(TAB)</code> <code>init_timer (TASB)</code>	W.ETABS
W.ETABS	<code>data_carrier_off</code>	<code>stop_timer(TASB)</code> <code>stop_timer(TAB)</code>	<u>Stopped</u>
W.ETABS	<code>time_out(TAB)</code>	<code>Phy_ABORT.ind(EP-3)</code> <code>Phy_ALARM.ind()</code>	W.TASB
W.AG	<code>AG_sent_event</code>	<code>Phy_APPG.ind()</code> <code>init_timer(TEMPO)</code>	W.TAB

Initial state	Triggering condition	Set of actions	Final state
W.AG	Phy_ABORT.req()	FlagAbort=TRUE	W.AG
W.TAB	data-carrier_on	Carrier = TRUE init_timer(TAB) init_timer (TASB)	W.TAB
W.TAB	data-carrier_off	Carrier = FALSE stop_timer(TAB) stop_timer (TASB)	W.TAB
W.TAB	time_out(TEMPO) & not(FlagAbort) & not(Carrier)	init_timer(TOL)	M.Send
W.TAB	time_out(TEMPO) & FlagAbort & not(Carrier)	wait_time (TOL)	<i>T.Session</i>
W.TAB	time_out(TEMPO) & Carrier	init_timer(TOL)	W.ETAB
W.TAB	Phy_ABORT.req()	FlagAbort=TRUE	W.TAB
W.ETAB	time_out(TAB)	Phy_ABORT.ind(EP-3) Phy_ALARM.ind() stop_timer(TOL)	W.TASB
W.ETAB	data_carrier_off & not(FlagAbort)	stop_timer(TAB) stop_timer (TASB)	M.Send
W.ETAB	data_carrier_off & FlagAbort	stop_timer(TAB) stop_timer (TASB)	W.TOL
W.ETAB	Phy_ABORT.req()	FlagAbort=TRUE	W.ETAB
W.TASB	time_out(TASB)	\$none()	<u>Stopped</u>
W.TOL	time_out(TOL)	\$none()	<i>T.Session</i>
M.Send	Phy_DATA.req(Frame)	Service=NORMAL	<i>SendFirst</i>
M.Send	Phy_UNACK.req(Frame)	Service=UNACKNOWLEDGED	<i>SendFirst</i>
M.Send	Phy_ASO.req(Frame)	Service=ASO	<i>SendFirst</i>
M.Send	Phy_ABORT.req()	\$none()	M.Send
M.Send	time_out(TOL)	\$none()	<i>T.Session</i>
<i>T.Session</i>	SessionAGT=TRUE	init_timer(TOAG) Phy_ABORT.ind(EP-1) wait_time(TEMPO)	<u>Stopped</u>
<i>T.Session</i>	SessionAGT=FALSE	Phy_ABORT.ind(EP-1) wait_time(TEMPO)	<u>Stopped</u>
<i>SendFirst</i>	\$true()	stop_timer(TOL) Size=size(Frame) Index=1 send_octet(Frame, Index) Size=Size-1 init_timer(TOE)	Sending
Sending	octet_sent_event & Size>0	Index=Index+1 send_octet(Frame, Index) Size=Size-1	Sending
Sending	octet_sent_event & Size=0	stop_timer(TOE) wait_time(TAO) Index=1 Frame=""	<i>Answer</i>
Sending	Phy_ABORT.req()	stop_timer(TOE) wait_time(TAO) init_timer(TA10) FlagAbort=TRUE	M.Rec
Sending	time_out(TOE)	Phy_ABORT.ind(EP-3F) wait_time(TAO) init_timer(TA10) FlagAbort=TRUE	M.Rec
<i>Answer</i>	Service=NORMAL Service=UNACKNOWLEDGED	init_timer(TA10)	M.Rec
<i>Answer</i>	Service=ASO	WinRSO=1 init_timer(TARSO) init_timer(TA10)	M.Rec

Initial state	Triggering condition	Set of actions	Final state
M.Rec	octet_received_event	stop_timer(TA1O) Index=Index+1 read_data(RecB) concat(Frame, RecB) init_timer(TAO)	Receiving
M.Rec	collision_detected_event	stop_timer(TA1O) Collision=TRUE init_timer(TAO)	Receiving
M.Rec	time_out(TA1O)	\$none()	<i>Received</i>
M.Rec	Phy_ABORT.req()	FlagAbort=TRUE	M.Rec
Receiving	octet_received_event & Index<=MaxIndex	stop_timer(TAO) Index=Index+1 read_data(RecB) concat(Frame, RecB) init_timer(TAO)	Receiving
Receiving	octet_received_event & Index>MaxIndex	Phy_ABORT.ind(EP-4F) wait_time(TAO) FlagAbort=TRUE	<i>Received</i>
Receiving	collision_detected_event	stop_timer(TAO) Collision=TRUE init_timer(TAO)	Receiving
Receiving	time_out(TAO)	\$none()	<i>Received</i>
Receiving	time_out(TARSO)	Phy_ABORT.ind(EP-5F) wait_time(TAO) FlagAbort=TRUE	<i>Received</i>
Receiving	Phy_ABORT.req()	Flagabort = TRUE	Receiving
<i>Received</i>	Service=NORMAL & not(Flagabort)	Phy_DATA.ind(Frame) init_timer(TOL)	M.Send
<i>Received</i>	(Service=NORMAL & Flagabort) Service=UNACKNOWLEDGED	wait_time(TOL)	<i>T.Session</i>
<i>Received</i>	Service=ASO & Collision & not(Flagabort)	Phy_COLL.ind() Collision=FALSE	<i>T.RSO</i>
<i>Received</i>	Service=ASO & not(Collision) & not(Flagabort)	Phy_AS0.ind(Frame)	<i>T.RSO</i>
<i>Received</i>	Service=ASO & Flagabort	\$none()	<i>T.RSO</i>
<i>T.RSO</i>	(TypeAG=AGT) (WinRSO>=MaxRSO) & (TypeAG=AGN)	stop_timer(TARSO)	<i>T.Session</i>
<i>T.RSO</i>	(WinRSO<MaxRSO) & (TypeAG=AGN)	Index=1 Frame=""	W.RSO
W.RSO	time_out(TARSO)	WinRSO=WinRSO+1 init_timer(TARSO) init_timer(TA1O)	M.Rec
W.RSO	Phy_ABORT.req()	Flagabort=TRUE	W.RSO

**Table 5 – Power supply management state transitions
(only for non-energized Secondary Station)**

Initial state	Triggering condition	Set of actions	Final state
<u>Initial</u>	alarm_detection()	Flagalarm = TRUE FlagSendAlarm =FALSE station_power(ON)	<u>Stopped</u>
<u>Initial</u>	not(alarm_detection())	Flagalarm = FALSE	<u>Stopped</u>
Stopped	occur(cpt_carrier_on) & Flagalarm	init_timer(TVASB)	W.TVASB2
Stopped	occur(data_carrier_on)	init_timer(TOSEUIL) init_timer(TAGT)	W.TOSEUIL
W.TOSEUIL	time_out(TOSEUIL)& not(Flagalarm)	station_power(ON)	W.AGT
W.TOSEUIL	occur(data_carrier_off) & not(Flagalarm)	stop_timer(TOSEUIL) stop_timer(TAGT)	<u>Stopped</u>
W.TOSEUIL	time_out(TOSEUIL) & Flagalarm	station_signal(ON) Tend = TOAG	W.AGT
W.TOSEUIL	occur(data_carrier_off) & Flagalarm	stop_timer(TOSEUIL) stop_timer(TAGT) Tend = TOAGN init_timer(Tend)	Hide
W.TOSEUIL	occur(cpt_carrier_on) & Flagalarm	init_timer(TVASB)	W.TVASB1
W.AGT	occur(data_carrier_off)	stop_timer(TAGT) init_timer(TOAPPEL)	W.Sel
W.AGT	time_out(TAGT) & not(Flagalarm)	station_power(OFF)	<u>Stopped</u>
W.AGT	time_out(TAGT) & Flagalarm	init_timer(Tend)	Hide
W.Sel	occur(octet_received_event)	stop_timer(TOAPPEL) init_timer(TOBAVARD) init_timer(TAO)	Select
W.Sel	time_out(TOAPPEL) & not(Flagalarm)	station_power(OFF)	Stopped
W.Sel	time_out(TOAPPEL) & Flagalarm	station_signal(OFF)	Stopped
W.Sel	occur(cpt_carrier_on) & Flagalarm & not(FlagSendalarm)	init_timer(TVASB)	W.TVASB1
Select	occur(octet_received_event)	stop_timer(TAO) init_timer(TAO)	Select
Select	time_out(TAO)	stop_timer(TOBAVARD) init_timer(TOPRE)	W.Answer
Select	time_out(TOBAVARD) & not(Flagalarm)	stop_timer(TAO) station_power(OFF)	<u>Stopped</u>
Select	time_out(TOBAVARD) & Flagalarm	stop_timer(TAO) init_timer(Tend)	Hide
W.Answer	occur(octet_sent_event)	stop_timer(TOPRE) init_timer(Tend)	Hide
W.Answer	time_out(TOPRE) & not(Flagalarm)	station_power(OFF)	<u>Stopped</u>
W.Answer	time_out(TOPRE) & Flagalarm	init_timer(Tend)	Hide
W.Answer	occur(cpt_carrier_on) & Flagalarm & not(FlagSendalarm)	init_timer(TVASB)	W.TVASB1
Hide	occur(octet_received_event) occur(octet_sent_event) (occur(data_carrier_on) & not(FlagSendAlarm))	stop_timer(Tend) init_timer(Tend)	Hide
Hide	occur(data_carrier_on) & FlagSendAlarm	stop_timer(Tend)	W.AGend
Hide	time_out(Tend) & not(Flagalarm)	station_power(OFF)	<u>Stopped</u>
Hide	time_out(Tend) & Flagalarm & not(FlagSendAlarm)	station_signal(OFF)	<u>Stopped</u>
Hide	time_out(Tend) & Flagalarm & FlagSendAlarm	Send_AG(AGN)	W.AB
Hide	occur(cpt_carrier_on) & Flagalarm & not(FlagSendalarm)	init_timer(TVASB)	W.TVASB1
W.AGend	occur(data_carrier_off)	wait_time(TOALR) Send_AG(AGN)	W.AB

Initial state	Triggering condition	Set of actions	Final state
W.TVASB1	occur(cpt_carrier_off)	stop_timer(TVASB) init_timer(Tend)	Hide
W.TVASB1	time_out(TVASB)	FlagSendAlarm = TRUE init_timer(Tend)	Hide
W.TVASB1	time_out(Tend)	\$none()	W.TVASB2
W.TVASB2	occur(cpt_carrier_off)	stop_timer(TVASB) station_signal(OFF)	<u>Stopped</u>
W.TVASB2	occur(data_carrier_on)	\$none()	W.TVASB1
W.TVASB2	time_out(TVASB)	Send_AG(AGN)	W.AB
W.AB	AG_sent_event	FlagSendAlarm = FALSE station_signal(OFF)	<u>Stopped</u>

Table 6 – *Physical-62056-3-1* state transitions: Secondary Station

Initial state	Triggering condition	Set of actions	Final state
<u>Initial</u>	energized()	MaxIndex=128 FlagRSO=FALSE FirstWinRSO=FALSE	<u>Stopped</u>
<u>Initial</u>	not(energized())	MaxIndex=18 FlagRSO=FALSE FirstWinRSO=TRUE	<u>Stopped</u>
<u>Stopped</u>	AG_received_event	Stop_timer(TOAG) init_timer(TA1O)	M.Rec
<u>Stopped</u>	Phy_ALARM.req()	TypeAG=ASB Send_AG(TypeAG)	W.ASB
<u>Stopped</u>	time_out(TOAG)	MaxIndex=18 FirstWinRSO=TRUE	<u>Stopped</u>
M.Rec	octet_received_event	Stop_timer(TA1O) Index=2 Frame="" Read_data(RecB) Concat(Frame, RecB) init_timer(TAO)	Receiving
M.Rec	time_out(TA1O)	Phy_ABORT(EP-1)	WTOAG
M.Rec	Phy_ABORT.req()	Stop_timer(TA1O)	WTOAG
Receiving	octet_received_event & Index<=MaxIndex	Stop_timer(TAO) Index=Index+1 Read_data(RecB) Concat(Frame, RecB) init_timer(TAO)	Receiving
Receiving	octet_received_event & Index>MaxIndex	Stop_timer(TAO) Phy_ABORT.ind(EP-4F)	WTOAG
Receiving	time_out(TAO)	Phy_DATA.ind(Frame) init_timer(TOL)	M.Send
Receiving	Phy_ABORT.req()	Stop_timer(TAO)	WTOAG
M.Send	Phy_DATA.req(Frame)	Stop_timer(TOL) MaxIndex=128 Size=size(Frame) Index=1 Send_octet(Frame, Index) Size=Size-1 init_timer(TOE)	Sending

Initial state	Triggering condition	Set of actions	Final state
M.Send	Phy_RSO.req(Frame, Window)	Stop_timer(TOL) MaxIndex=128 Wait_window(FirstWinRSO, Window) FirstWinRSO=FALSE Size=size(Frame) Index=1 Send_octet(Frame, Index) Size=Size-1 FlagRSO=TRUE init_timer(TOE)	Sending
M.Send	time_out(TOL)	init_timer(TA1O)	M.Rec
M.Send	Phy_ABORT.req()	Stop_timer(TOL)	WTOAG
Sending	octet_sent_event & Size>0	Index=Index+1 Send_octet(Frame, Index) Size=Size-1	Sending
Sending	octet_sent_event & Size=0 & not(FlagRSO)	Stop_timer(TOE) init_timer(TA1O)	M.Rec
Sending	octet_sent_event & Size=0 & FlagRSO	Stop_timer(TOE) Wait_time(TAO) FlagRSO=FALSE	WTOAG
Sending	Phy_ABORT.req()	Stop_timer(TOE)	WTOAG
Sending	time_out(TOE)	Phy_ABORT.ind(EP-3F)	WTOAG
W.ASB	time_out(TOAG)	MaxIndex=18 FirstWinRSO=TRUE	W.ASB
W.ASB	AG_sent_event	\$none()	<u>Stopped</u>
WTOAG	not(energized)	init_timer(TOAG)	<u>Stopped</u>
WTOAG	Energized	\$none()	<u>Stopped</u>

Table 7 – Meaning of the states listed in the previous tables

State	Meaning
<u>Initial</u>	Initialization of the variables of the layer
<u>Stopped</u>	Waiting state for a "Wakeup Call" signal
W.ETABS (Wait for end of "Alarm-Bus")	Waiting for the end of an "Alarm-bus" signal received in a Stopped state
W.AG (Wait for end of "Wakeup Call")	Waiting for the end of a "Wakeup Call" signal transmission
W.TAB (Wait "Alarm-Bus")	Waiting for an "Alarm-Bus" signal during delay at the end of transmission of a "Wakeup call" signal
W.ETAB (Wait for end of "Alarm-Bus")	Waiting for the end of an "Alarm-Bus" signal received after the transmission of a "Wakeup call" signal
W.TASB	Waiting for the triggering of wakeup TASB after the beginning of the reception of an "Alarm-Bus" signal
W.TOL	Waiting for the triggering of wakeup TOL
M.Send (Must Send)	Initial state of the transmitter waiting for a frame to send
<i>T.Session</i>	Testing the type of the session (with an energized or not energized Secondary Station)
<i>SendFirst</i>	Sending the first byte of the frame to be sent
Sending	Recurrent state of the transmitter transmitting one byte at a time
<i>Answer</i>	Branching depending on the service requested

State	Meaning
M.Rec (Must Receive)	Initial state of the receiver waiting for the first byte of a frame
Receiving	Recurrent state of the receiver receiving one byte at a time
<i>Received</i>	Processing the received frame depending on the service requested
T.RSO (Test last RSO)	Testing the end of the last time slot for RSO frame reception
W.RSO (Wait for end of an RSO time slot)	Waiting for the end of a time slot for RSO frame reception
W.ASB	Waiting for the end of a "Alarm Secondary-Bus" signal transmission
W.TOAG	Initializing the "end of session" TOAG timer if needed
W.TOSEUIL	Waiting for the triggering of wakeup TOSEUIL
W.AGT	Waiting for an AGT "Wakeup Call" signal
W.Sel (Wait for preSelection)	Waiting for a preselection frame
Select	Receiving a preselection frame
W.Answer	Waiting for an answer frame from a selected station
Hide	Waiting for the end of selection
W.AGend	Waiting for the end of AG reception
W.TVASB1	Waiting for the triggering of wakeup TVASB for an "Alarm Secondary-Bus" signal during a session
W.TVASB2	Waiting for the triggering of wakeup TVASB for an "Alarm Secondary-Bus" signal at the end of session
W.AB	Waiting for the end of a "Alarm Bus" signal transmission

**Table 8 – Definition of the procedures, functions and events
classified in alphabetical order**

Procedure, function or event	Definition
AG_received_event	Event from the modem reporting that an AGN "Wakeup Call" signal has been correctly detected
AG_sent_event	Event from the modem reporting the end of the transmission of a "Wakeup Call" signal
alarm_detection()	Check that the station status of alarm mode is Active
collision_detected_event	Event from the modem reporting the detection of a framing error on reception of a byte
concat(Frame, RecB)	Concatenation of the byte RecB in the being built frame Frame
data_carrier_on, data_carrier_off	Occurrence of the detection on the bus of the data carrier on, the data carrier off
energized()	Check that the station is energized
init_timer(TOAPPEL), init_timer(TOSEUIL), init_timer(TAGT), init_timer(TOBAVARD), init_timer(TOPRE), init_timer(TOL), init_timer(TOE), init_timer(TAO), init_timer(TA1O), init_timer(TARSO), init_timer(TOAG), init_timer(TVASB) or init_timer(TAB)	setting of wakeup TOAPPEL, TOSEUIL, TAGT, TOBAVARD, TOPRE, TOL, TOE, TAO, TA1O, TARSO, TOAG, TVASB or TAB

Procedure, function or event	Definition
occur(cpt_carrier_on), occur(cpt_carrier_off), occur(data_carrier_on), occur(data_carrier_off), occur(octet_received_event) or occur(octet_sent_event)	Occurrence (reporting without consuming) of the detection on the secondary bus of the data carrier on, the data carrier off, on the bus of the data carrier on, the data carrier off, the reception of a byte or the emission of a byte
octet_received_event	Event from the modem reporting that a byte has been received
octet_sent_event	Event from the modem reporting that a byte has been sent
read_data(RecB)	Processing of the byte_received_event by reading the received RecB byte (bits are transmitted in ascending order)
send_AG(TypeAG)	Request to the modem for transmission of a "Wakeup Call" signal of duration TypeAG (AGN or AGT)
send_octet(Frame, Index)	Transmission of the byte of rank Index in the frame Frame (bits are transmitted in ascending order)
size(Frame)	Calculation of the number of bytes of the frame Frame
station_power(ON) or station_power(OFF)	Turning ON or OFF the energy supply to the device
station_signal(ON) or station_signal(OFF)	Turning ON or OFF the signal transmission to the device on the secondary bus
stop_timer(TOAPPEL), stop_timer(TOSEUIL), stop_timer(TAGT), stop_timer(TOBAVARD), stop_timer(TOPRE), stop_timer(TOL), stop_timer(TOE), stop_timer(TAO), stop_timer(TA1O), stop_timer(TVASB) or stop_timer(TAB)	Stopping of wakeup TOAPPEL, TOSEUIL, TAGT, TOBAVARD, TOPRE, TOL, TOE, TAO, TA1O, TVASB or TAB
stop_timer(TOAG) or stop_timer(TARSO)	Stopping of wakeup TOAG or TARSO only if it has previously been set
time_out(TOAPPEL), time_out(TOSEUIL), time_out(TAGT), time_out(TOBAVARD), time_out(TOPRE), time_out(TOL), time_out(TOE), time_out(TAO), time_out(TA1O), time_out(TARSO), time_out(TOAG), time_out(TVASB) or stop_timer(TAB)	Triggering of wakeup TOAPPEL, TOSEUIL, TAGT, TOBAVARD, TOPRE, TOL, TOE, TAO, TA1O, TARSO, TOAG, TVASB or TAB
wait_time(TAO), wait_time(TICB), wait_time(TOL) or wait_time(TOALR)	Calculated delay during time TAO, TICB, TOL or TOALR
wait_window(FirstWinRSO, Window)	Wait-time calculated as follows: when FirstWinRSO=TRUE or Window=0 ==> 0 ms when FirstWinRSO=FALSE and Window>0 ==> 40 ms + (TARSO*Window) ms (The 40 ms delay guarantees that the transmission has taken place in the time slot)

5.1.6 List and processing of errors

Errors are listed using the following codes:

EP = error in the *Physical* layer

– = separator

N = error number

F = fatal error

Errors related to *Physical 62056-3-1* are specified in Table 9.

Table 9 – Error summary table

EP-1	Expiry of TOL wakeup (Primary Station) before the <i>Data Link</i> layer requests a frame transmission or expiry of TA1O wakeup (Secondary Station) before any character has been received from the Primary station
	This error leads to the expectation of a "Wakeup Call" signal after having informed the <i>Data Link</i> layer
EP-2	Expiry of TOAG wakeup before any "Wakeup Call" signal
	This error leads to the expectation of a "Wakeup Call" signal after having informed the <i>Data Link</i> layer
EP-3	An alarm has been received
	This error leads to the reinitialization of the <i>Physical</i> layer after having informed the <i>Data Link</i> layer
EP-3F	Abnormal length of transmission detected after expiry of TOE wakeup
	This error leads to the reinitialization of the <i>Physical</i> layer after having informed the <i>Data Link</i> layer
EP-4F	Number of bytes received higher than MaxIndex (Transmitter too talkative)
	This error leads to the reinitialization of the <i>Physical</i> layer after having informed the <i>Data Link</i> layer
EP-5F	Expiry of TARSO wakeup while receiving an RSO frame (Primary Station only)
	This error leads to the reinitialization of the <i>Physical</i> layer after having informed the <i>Data Link</i> layer

If any of these errors occurs, it is sent up locally by means of the *Phy_ABORT.ind* service primitive. The complete list of fatal error numbers is given in Annex C.

5.2 Data Link layer

5.2.1 Link-62056-3-1 protocol

The *Link-62056-3-1* protocol of the *Data Link* layer of the local bus data exchange profile without DLMS behaves asymmetrically. The state machine of the Primary Station is therefore different from that of the Secondary Station.

The *Data Link* layer transforms the physical channel used by the *Physical* layer to a logic channel able to transmit reliable information. Its main functions are:

- to carry out a serialization and a deserialization of the data (if the physical channel functions serially one bit at a time);
- to synchronize the transmission and reception frames;
- to filter the frames according to primary and secondary addresses;
- to ensure efficient protection against transmission errors.

5.2.2 Management of exchanges

On the Primary station, the *Link-62056-3-1* protocol takes over the transmission of an AGN or AGT "Wakeup Call" signal according to the type of Secondary Station. Detection of incompatibility in the addresses of a DSDU received from the upper layer indicates a fatal error and the stop of the *Link-62056-3-1* protocol.

On the Secondary Station, reception of an incorrect frame does not require any processing, as recovering is left to the Primary Station.

For non-energized stations, the detection of a "Forgotten Stations Call" after an AGT "Wakeup Call" signal leads to an RSO response which always takes place in the first RSO time slot. Thus, the Primary Station, when detecting a collision after such a sequence, performs a second "Forgotten Stations Call", but this time after an AGN "Wakeup Call" signal. Nevertheless, when no collision is detected after the first call, there is one or no forgotten station and no need for a second call.

5.2.3 Data Link services and service primitives

The user of the *Link-62056-3-1* protocol can use the services and service primitives given in Table 10.

Table 10 – Data Link services and service primitives

Service	Service primitive
DL_DATA	DL_DATA.req(DSDU) DL_DATA.ind(DSDU)
DL_ALARM	DL_ALARM.req() DL_ALARM.ind()
DL_ABORT	DL_ABORT.req() DL_ABORT.ind(ErrorNb)

The role assigned to each primitive is as follows:

- DL_DATA.req(DSDU) enables the *Application* layer to request the *Data Link* layer to transfer a DSDU data packet;
- DL_DATA.ind(DSDU) enables the *Data Link* layer to inform the *Application* layer of the arrival of a DSDU data packet;
- DL_ALARM.req() enables the *Application* layer to request the *Data Link* layer to transfer an alarm;
- DL_ALARM.ind() enables the *Data Link* layer to inform the *Application* layer of the arrival of an alarm;
- DL_ABORT.req() enables the *Application* layer to request the *Data Link* layer to end its activity;
- DL_ABORT.ind (ErrorNb) enables the *Data Link* layer to inform the *Application* layer of the occurrence of a fatal error identified by the number ErrorNb.

5.2.4 Data Link parameters

For the Primary Station, the value of the number of repeat transmissions for a given frame before disconnection, MaxRetry, is set to 2.

The value of the number of sequences linked with no "Wakeup Call" signal for remote reading and remote transfer, MaxChain, is set to 5, for compatibility with Secondary Stations using a previous version of the protocol.

The value of the maximum number, MaxRSO, of RSO time slots for the processing of a "Forgotten Stations Call" is set to 3 after an AGN "Wakeup Call" signal, at 1 after an AGT "Wakeup Call" signal.

The Secondary Station shall know the list of Primary Station addresses and the list of TABi to which it has been programmed.

The station may also be solicited by the general primary address APG. In this case, it replies with the first primary address to which it has been programmed.

5.2.5 State transitions

Link-62056-3-1 state transitions are as specified in Table 11, Table 12, Table 13 and Table 14.

Table 11 – *Link-62056-3-1* state transitions: Primary Station

Initial state	Triggering condition	Set of actions	Final state
<i>Initial</i>	\$true()	MaxRetry=2 MaxChain=5	Stopped
Stopped	DL_DATA.req(DSDU) & check_req(DSDU)	NbChain=0 MaxRSO=3 PreSel=FALSE NoRetry=FALSE RepeatASO=FALSE EP-1=FALSE context(ADS, ADP, TypeAG) Com=command(DSDU) init(Com, TypeAG) Phy_APPG.req(TypeAG)	W.AG
Stopped	DL_DATA.req(DSDU) & not(check_req(DSDU))	DL_ABORT.ind(EL-1F)	Stopped
Stopped	Phy_ALARM.ind()	DL_ALARM.ind()	Stopped
W.AG	Phy_APPG.ind() & not(PreSel) & not(NoRetry) & not(RepeatASO)	\$none()	<i>T.Req</i>
W.AG	Phy_APPG.ind() & PreSel	Index=MaxRetry+1 Fr=PRE Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_DATA.req(Fr)	M.Rec
W.AG	Phy_APPG.ind() & NoRetry	Index=MaxRetry+1 Fr=DSDU Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) NbChain=1 Phy_DATA.req(Fr) NoRetry=FALSE	M.Rec
W.AG	Phy_APPG.ind() & RepeatASO	RepeatASO=FALSE Phy_ASQ.req(Fr)	M.RSO
W.AG	DL_ABORT.req()	Phy_ABORT.req()	W.EndS
W.EndS	Phy_ABORT.ind(ErrorNb)	\$none()	<i>T.Error</i>
W.EndS	Phy_ALARM.ind()	DL_ALARM.ind()	Stopped
<i>T.Error</i>	((Error_Nb = EP-1 & TypeAG = AGN) (Error_Nb = EP-2 & TypeAG = AGT)) & Com <> IB & Com <> TRB	\$none()	Stopped
<i>T.Error</i>	((Error_Nb = EP-1 & TypeAG = AGN) (Error_Nb = EP-2 & TypeAG = AGT)) & (Com=IB Com=TRB)	DL_ABORT.ind(Error_Nb)	Stopped

Initial state	Triggering condition	Set of actions	Final state
<i>T.Error</i>	Error_Nb <> EP-1 & Error_Nb <> EP-2	DL_ABORT.ind(Error_Nb)	W.EndS
<i>T.Error</i>	(Error_Nb = EP-1) & TypeAG = AGT	\$none()	W.EndS
<i>T.Req</i>	Com=IB Com=TRB	Fr=DSDU Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_UNACK.req(Fr)	W.EndS
<i>T.Req</i>	Com=ASO & TypeAG=AGN	MaxRSO=3 NbRSO=1 ListRSO="" Collision=FALSE Fr=DSDU Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_ASQ.req(Fr)	M.RSQ
<i>T.Req</i>	Com=ASO & TypeAG=AGT	MaxRSO=1 NbRSO=1 ListRSO="" Collision=FALSE Fr=DSDU Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_ASQ.req(Fr)	M.RSQ
<i>T.Req</i>	((NbChain<MaxChain) & (Com=ENQ Com=TRF)) (NbChain=0 & Com=REC)	Fr=DSDU Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Index=1 NbChain=NbChain+1 Phy_DATA.req(Fr)	M.Rec
<i>T.Req</i>	Com=AUT	Fr=DSDU Fr=concat(size_frame(Fr),ADS,ADP,Fr) Fr=concat(Fr, crc(Fr)) Index=1 NbChain=MaxChain Phy_DATA.req(Fr)	M.Rec
<i>T.Req</i>	(NbChain>=MaxChain) (NbChain<>0 & Com=REC)	NbChain=0 Phy_APPG.req(AGN)	W.AG
M.Rec	Phy_DATA.ind(Frame) & check_frame(Frame) & not(PreSel)	DSDU=extract_DSDU(Frame) DL_DATA.ind(DSDU)	M.Send
M.Rec	Phy_DATA.ind(Frame) & check_frame(Frame) & command(Frame)=SEL & PreSel	PreSel=FALSE	<i>T.Req</i>
M.Rec	Phy_DATA.ind(Frame) & check_frame(Frame) & command(Frame)<>SEL & PreSel	Phy_ABORT.req() DL_ABORT.ind(EL-2F)	W.EndS
M.Rec	Phy_DATA.ind(Frame) & not(check_frame(Frame)) & Index<=MaxRetry	Index=Index+1 Phy_DATA.req(Fr)	M.Rec
M.Rec	Phy_DATA.ind(Frame) & not(check_frame(Frame)) & Index>MaxRetry	Phy_ABORT.req() DL_ABORT.ind(EL-2F)	W.EndS
M.Rec	DL_ABORT.req()	Phy_ABORT.req()	W.EndS
M.Rec	Phy_ABORT.ind(ErrorNb)	DL_ABORT.ind(ErrorNb)	W.EndS
M.RSQ	Phy_ASQ.ind(Frame) & size(Frame)=0	\$none()	<i>T.RSQ</i>
M.RSQ	Phy_ASQ.ind(Frame) & size(Frame)<>0 & check_frame(Frame) & command(Frame)=RSQ	build_RSQ(ListRSQ, Frame)	<i>T.RSQ</i>
M.RSQ	Phy_ASQ.ind(Frame) & size(Frame)<>0 & not(check_frame(Frame))	Collision=TRUE	<i>T.RSQ</i>

Initial state	Triggering condition	Set of actions	Final state
M.RSO	Phy_COLL.ind()	Collision=TRUE	T.RSO
M.RSO	DL_ABORT.req()	Phy_ABORT.req()	W.EndS
M.RSO	Phy_ABORT.int(ErrorNb)	DL_ABORT.ind(ErrorNb)	W.EndS
T.RSO	MaxRSO=1 & Collision	MaxRSO=3 Collision=FALSE RepeatASO=TRUE Phy_APPG.req(AGN)	W.AG
T.RSO	(MaxRSO=1 & not(Collision)) (MaxRSO<>1 & NbRSO>=MaxRSO)	DSDU=rso(RSO, Collision, ListRSO) DL_DATA.ind(DSDU)	W.EndS
T.RSO	NbRSO<MaxRSO	NbRSO=NbRSO+1	M.RSO
M.Send	DL_DATA.req(DSDU) & check_req(DSDU) & not(EP-1)	Com=command(DSDU)	T.Req
M.Send	DL_DATA.req(DSDU) & check_req(DSDU) & EP-1	Com=command(DSDU) NbChain=0 EP-1=FALSE Phy_APPG.req(AGN)	W.AG
M.Send	DL_DATA.req(DSDU) & not(check_req(DSDU))	Phy_ABORT.req() DL_ABORT.ind(EL-1F)	W.EndS
M.Send	Phy_ABORT.ind(EP-1)	EP-1=TRUE	M.Send
M.Send	Phy_ABORT.ind(EP-2)	\$none()	Stopped
M.Send	DL_ABORT.req() & not(EP_1 & TypAG=AGN)	Phy_ABORT.req()	W.EndS
M.Send	DL_ABORT.req() & EP_1 & TypAG=AGN	\$none()	Stopped
M.Send	Phy_ABORT.ind(ErrorNb) & ErrorNb<>EP-1 & ErrorNb<>EP-2	DL_ABORT.ind(ErrorNb)	W.EndS

Table 12 – *Link-62056-3-1* State transitions: Secondary Station

Initial state	Triggering condition	Set of actions	Final state
<i>Initial</i>	\$true()	FlagDSO=TRUE Discovered=FALSE Flag_alarm = FALSE	Stopped
Stopped	Phy_DATA.ind(Frame) & check_frame(Frame) & check_address(Frame)	ADP=extract_ADP(Frame) Com=command(Frame)	<i>T.Com</i>
Stopped	Phy_DATA.ind(Frame) & check_frame(Frame) & not(check_address(Frame))	Phy_ABORT.req()	Stopped
Stopped	Phy_DATA.ind(Frame) & not(check_frame(Frame))	\$none()	Stopped
Stopped	Phy_ABORT.ind(ErrorNb)	DL_ABORT.ind(ErrorNb)	Stopped
Stopped	DL_ALARM.req()	Phy_ALARM.req() Flag_alarm = TRUE	Stopped
<i>T.Com</i>	Com=IB	FlagDSO=TRUE Discovered=FALSE Phy_ABORT.req()	Stopped
<i>T.Com</i>	Com=TRB	DSDU=extract_DSDU(Frame) DL_DATA.ind(DSDU) Phy_ABORT.req()	Stopped
<i>T.Com</i>	Com=PRE	Fr=SEL Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_DATA.req(Fr)	Stopped
<i>T.Com</i>	Com=ASO & test_TABi(Frame, TAB)	Fr=concat(RSO, TAB, ADS) Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_RSO.req(Fr, window_RSO())	Stopped
<i>T.Com</i>	Com=ASO & not(test_TABi(Frame, TAB))	Phy_ABORT.req()	Stopped
<i>T.Com</i>	Com=ENQ Com=REC Com=TRF COM=AUT	DSDU=extract_DSDU(Frame) DL_DATA.ind(DSDU)	M.Send
M.Send	DL_DATA.req(DSDU)	Fr=DSDU Fr=concat(size_frame(Fr), ADS, ADP, Fr) Fr=concat(Fr, crc(Fr)) Phy_DATA.req(Fr) update_flag_DSO(command(Fr))	Stopped
M.Send	Phy_ABORT.ind(ErrorNb)	DL_ABORT.ind(ErrorNb)	Stopped

Table 13 – Meaning of the states listed in the previous tables

State	Meaning
<i>Initial</i>	Initialization of the variables of the layer
Stopped	Waiting state for the first request from the upper layer or the first indication from the lower layer
W.AG (Wait for end of "Wakeup Call")	Waiting for the end of a requested "Wakeup Call" signal
W.EndS (Wait for End of Session)	Waiting for the end of the session
<i>T.Req</i> (Test Request)	Test the nature of a request from the upper layer
M.Rec (Must Receive)	Waiting state for an indication from the lower layer
M.RSO (Must receive RSO)	Waiting for an RSO frame following an ASO frame sent
<i>T.RSO</i> (Test last RSO)	Testing the end of the last time slot for RSO frame reception
M.Send (Must Send)	Waiting state for a request from the upper layer
<i>T.Com</i> (Test Command)	Test of the command code of a received frame

Table 14 – Definition of the procedures and functions classified in alphabetical order

Procedure or function	Definition
build_RSO(ListRSO, Frame)	Extraction of the RSO elements (TAB and ADS fields) from the received RSO frame Frame and concatenation with the preceding list ListRSO
check_address(Frame)	Check that the ADP and ADS addresses are recognized according to the following criteria: <ul style="list-style-type: none"> – ADP is APG or the station has been programmed to the ADP address; – if the command code is ASO, IB or TRB, then ADS is ADG; – if the command code is not ASO, IB nor TRB, then ADS is the secondary station address
check_frame(Frame)	Check that the frame Frame received is correct: <ul style="list-style-type: none"> – number of bytes greater than or equal to 11 and lower than or equal to 128; – CRC correct; – number of bytes compatible with the field N; – command code recognized and number of bytes compatible with the command code
check_req(DSDU)	Check that the requested command code inside a DSDU is compatible with ADP and ADS addresses defined in the communication context
command(DSDU) or command(Frame)	Extraction of the value of the command code of a DSDU to send or a received frame Frame
concat(N, ADS, ADP, DSDU) or concat(Frame, CRC)	Concatenation of the fields N, ADS and ADP with the DSDU or concatenation of the CRC at the end of the frame Frame
context(ADS, ADP, TypeAG)	Extraction of the corresponding values from the communication context
crc(Frame)	Calculation of the CRC of the frame Frame to send
extract_ADP(Frame)	If either the ADP value used in the frame is not APG or if it is the APG value but the list of the ADP values to which the Secondary Station has been programmed is empty, then extraction of this value, otherwise extraction of the first ADP value to which the Secondary Station has been programmed

Procedure or function	Definition
extract_DSDU(Frame)	Extraction of the DSDU (COM and DATA fields) from the frame Frame received
init(COM, TypeAG)	Set PreSel to TRUE if both TypeAG equals AGT and the size of the frame is greater than 18 bytes. Set NoRetry to TRUE if both TypeAG equals AGT and the size of the frame is lower than or equal to 18 bytes.
rso(RSO, Collision, ListRSO)	Concatenation of the RSO command code, the collision indicator and the list of the RSO elements (TAB and ADS fields)
size(Frame)	Calculation of the size of the frame Frame received
size_frame(DSDU)	Calculation of the size of the frame associated with a DSDU to be sent (size(DSDU) + 10)
test_TABi(Frame, TAB)	If the first TABi contained in the received ASO frame Frame equals 00, check that Discovered=FALSE then, after provision of a whole random integer between 0 and 100, check that this integer is smaller than the response probability (second TABi). In this case, 00 is memorized in the TAB variable; If the first TABi contained in the received ASO frame Frame equals FF, check that Flag_alarm=TRUE then, in this case, Flag_alarm is set to FALSE and FF is memorized in the TAB variable; If the first TABi contained in the received ASO frame Frame does not equal 00 or FF, check that FlagDSO=TRUE and check that the Secondary Station has been programmed to one of the TABi contained in the received ASO frame Frame. In this case, the first of these values is memorized in the TAB variable
update_flag_DSO(COM)	Set FlagDSO to FALSE and Discovered to TRUE if COM equals ENQ
window_RSO()	Provision of a whole random integer between 0 and MaxRSO-1 used as the number of the RSO time slot in which the station shall reply (refer to Annex F)

5.2.6 List and processing of errors

Errors are listed using the following codes:

- EL = error in the *Data Link* layer
- = separator
- N = error number
- F = fatal error

Errors related to *Link-62056-3-1* are specified in Table 15.

Table 15 – Error summary table

EL-1F	Reception of a command code not compatible with the ADP and ADS addresses memorized in the communication context (Primary Station only)
	This error leads to the reinitialization of the <i>Data Link</i> layer after having informed the <i>Application</i> layer
EL-2F	Incorrect response from the Secondary Station after either a non-energized station preselection frame or MaxRetry repeated transmissions of a request (Primary Station only)
	This error leads to the re-initialization of the <i>Data Link</i> layer after having informed the <i>Application</i> layer

If any of these errors occurs, it is sent up locally by means of the DL_ABORT.ind service primitive. The complete list of fatal error numbers is given in Annex C.

5.3 Application layer

5.3.1 Application-62056-3-1 protocol

The *Application-62056-3-1* protocol of the *Application* layer of the local bus data exchange profile without DLMS behaves asymmetrically. The state machine of the Primary Station is therefore different from that of the Secondary Station.

The *Application-62056-3-1* protocol of the *Application* layer of the local bus data exchange profile without DLMS controls and links successive messages by analysing the command code supplied by the user application.

5.3.2 Application services and service primitives

The user of the *Application-62056-3-1* protocol can use the services and service primitives given in Table 16.

Table 16 – Application services and service primitives

Service	Service primitive
A_DATA	A_DATA.req(COM, ASDU) A_DATA.ind(ASDU)
A_ALARM	A_ALARM.req() A_ALARM.ind()
A_ABORT	A_ABORT.req() A_ABORT.ind(ErrorNb)

The role assigned to each primitive is as follows:

- A_DATA.req(COM, ASDU) enables the application to request the *Application* layer to transfer a COM (ENQ, REC, TRF, TRB, IB or ASO for a Primary Station and DAT, DRJ, EOS or TRA for a Secondary Station) command code linked with an ASDU information unit;
- A_DATA.ind(ASDU) enables the *Application* layer to inform the application of the arrival of an ASDU information unit;
- A_ALARM.req() enables the application to request the *Application* layer to send an alarm;
- A_ALARM.ind() enables the *Application* layer to inform the application of the arrival of an alarm;
- A_ABORT.req() enables the application to request the *Application* layer to end its activity;
- A_ABORT.ind(ErrorNb) enables the *Application* layer to inform the application of the occurrence of a fatal error identified by the number ErrorNb.

5.3.3 Application parameters

The Primary Station shall know the DES ciphering keys of all Secondary Stations for which a remote programming is to be carried out.

In case of remote programming, the Secondary Station shall know the DES ciphering key that could be used by the Primary Station.

5.3.4 State transitions

Application-62056-3-1 state transitions are as specified in Table 17, Table 18, Table 19 and Table 20.

Table 17 – Application-62056-3-1 state transitions: Primary Station

Initial state	Triggering condition	Set of actions	Final state
Stopped	A_DATA.req(Com, ASDU) & (Com=ASO Com=ENQ Com=TRF)	APDU=concat(Com, __, __, ASDU) DL_DATA.req(APDU)	M.Rec
Stopped	A_DATA.req(Com, ASDU) & (Com=IB Com=TRB)	APDU=concat(Com, __, __, ASDU) DL_DATA.req(APDU)	W.EndS
Stopped	A_DATA.req(Com, ASDU) & Com=REC	Na1=randomize() Zdt=zdt(ASDU) APDU=concat(REC, Na1, 0, Zdt) DL_DATA.req(APDU) Na1k=cipher(Na1)	M.Rec
Stopped	DL_ALARM.ind()	A_ALARM.ind()	Stopped
Stopped	A_ABORT.req()	DL_ABORT.req()	Stopped
M.Rec	DL_DATA.ind(DSDU)	Resp=command(DSDU)	T.Resp
M.Rec	A_ABORT.req()	DL_ABORT.req()	Stopped
M.Rec	DL_ABORT.ind(ErrorNb) & Error_Nb <> EP_1 & Error_Nb <> EP_2	A_ABORT.ind(ErrorNb)	Stopped
M.Rec	DL_ABORT.ind(ErrorNb) & (Error_Nb = EP_1 Error_Nb = EP_2)	\$none()	M.Rec
W.EndS	DL_ABORT.ind(ErrorNb)	A_ABORT.ind(ErrorNb)	Stopped
W.EndS	A_ABORT.req()	DL_ABORT.req()	W.EndS
T.Resp	(Com=ASO & Resp=RSO) (Com=ENQ & Resp=DAT) (Com=ENQ & Resp=DRJ) (Com=TRF & Resp=TRA) (Com=TRF & Resp=DRJ) (Com=AUT & Resp=EOS) (Com=AUT & Resp=DRJ)	A_DATA.ind(DSDU)	Stopped
T.Resp	Com=REC & Resp=ECH & na1k(DSDU)=Na1k & Zdt=zdt(DSDU)	Na2=na2(DSDU) Na2k=cipher(Na2) Com=AUT APDU=concat(AUT, 0, Na2k, "") DL_DATA.req(APDU)	M.Rec
T.Resp	(Com=ASO & Resp<>RSO) (Com=ENQ & Resp<>DAT & Resp<>DRJ) (Com=TRF & Resp<>TRA & Resp<>DRJ) (Com=REC & Resp<>ECH) (Com=AUT & Resp<>ARJ & Resp<>DRJ & Resp<>EOS)	A_ABORT.ind(EA-1F) DL_ABORT.req()	Stopped
T.Resp	Com=REC & (Resp<>ECH (na1k(DSDU)<>Na1k) (Zdt<>zdt(DSDU)))	A_ABORT.ind(EA-2F) DL_ABORT.req()	Stopped
T.Resp	Com=AUT & Resp=ARJ	A_ABORT.ind(EA-3F) DL_ABORT.req()	Stopped

Table 18 – Application-62056-3-1 state transitions: Secondary Station

Initial state	Triggering condition	Set of actions	Final state
Stopped	DL_DATA.ind(DSDU) & (command(DSDU)=ENQ command(DSDU)=TRF)	A_DATA.ind(DSDU) Req=command(DSDU)	M.Send
Stopped	DL_DATA.ind(DSDU) & command(DSDU)=TRB	A_DATA.ind(DSDU)	Stopped
Stopped	DL_DATA.ind(DSDU) & command(DSDU)=REC	Zdt=zdt(DSDU) Na1=na1(DSDU) Na1k=cipher(Na1) Na2=randomize() APDU=concat(ECH, Na1k, Na2, Zdt) DL_DATA.req(APDU) Na2k=cipher(Na2) Req=REC	W.AUT
Stopped	A_ALARM.req() & alarm_detection()	DL_ALARM.req()	Stopped
M.Send	A_DATA.req(COM, ASDU) & (((COM=DAT COM=DRJ) & Req=ENQ) ((COM=TRA COM=DRJ) & Req=TRF) (COM=DRJ & Req=REC))	APDU=concat(COM, _, _, ASDU) DL_DATA.req(APDU)	Stopped
M.Send	A_DATA.req(COM, ASDU) & (COM=EOS & Req=REC)	APDU=concat(EOS, 0, 0, "") DL_DATA.req(APDU)	Stopped
M.Send	DL_ABORT.ind(ErrorNb)	A_ABORT.ind(ErrorNb)	Stopped
W.AUT	DL_DATA.ind(DSDU) & command(DSDU)=AUT & na2k(DSDU)=Na2k	ASDU=concat(REC, _, _, Zdt) A_DATA.ind(ASDU)	M.Send
W.AUT	DL_DATA.ind(DSDU) & command(DSDU)=AUT & na2k(DSDU)≠Na2k	APDU=concat(ARJ, _, _, "") DL_DATA.req(APDU)	Stopped
W.AUT	DL_ABORT.ind(ErrorNb)	A_ABORT.ind(ErrorNb)	Stopped

Table 19 – Meaning of the states listed in the previous tables

State	Meaning
Stopped	Waiting state for the first request from the upper layer or the first indication from the lower layer
M.Rec (Must Receive)	Standby for the response to the request transmitted
T.Resp (Test Response)	Processing of the response received
M.Send (Must Send)	Waiting for a response to a received request
W.AUT (Wait for AUT frame)	Waiting for an AUT frame following an ECH response frame sent

**Table 20 – Definition of the procedures and functions
classified in alphabetical order**

Procedure or function	Definition
alarm_detection()	Check that the status of Alarm mode is Active
cipher(Na1) or cipher(Na2)	Ciphering of random number Na1 or Na2 by means of DES algorithm with key contained in the communication context
command(DSDU)	Extraction of the value of the command code of a received DSDU
concat(COM, _, _, ASDU), concat(COM, ZA1, ZA2, ZDT) or concat(COM, 0, 0, _)	Concatenation of a COM command code and an ASDU or concatenation of a COM command code with an encrypted value ZA1, an encrypted value ZA2 and a data field ZDT (TAB and DATA fields) or concatenation of a COM command code and the ZA1 = 0 and ZA2 = 0 fields
na1(DSDU)	Extraction of the Na1 value from the ZA1 field of a received REC frame
na1k(DSDU)	Extraction of the Na1k value from the ZA1 field of a received ECH frame
na2(DSDU)	Extraction of the Na2 value from the ZA2 field of a received ECH frame
na2k(DSDU)	Extraction of the Na2k value from the ZA2 field of a received AUT frame
randomize()	Generation of a random number according to the procedure described in Annex G
zdt(ASDU) or zdt(DSDU)	Extraction of data (TAB and DATA fields) from a REC request, a REC frame or an ECH frame

5.3.5 List and processing of errors

Errors are listed using the following codes:

- EA = error in the *Application* layer
- = separator
- N = error number
- F = fatal error

Errors related to *Link-62056-3-1* are specified in Table 21.

Table 21 – Error summary table

EA-1F	The response command code of the received frame does not correspond with the request command code (Primary Station only)
	This error leads to the re-initialization of the <i>Application</i> layer after having informed the application
EA-2F	The Secondary Station has not been correctly authenticated (Primary Station only)
	This error leads to the re-initialization of the <i>Application</i> layer after having informed the application
EA-3F	The Primary Station has not been correctly authenticated (Primary Station only)
	This error leads to the re-initialization of the <i>Application</i> layer after having informed the application

If any of these errors occurs, it is sent up locally by means of the A_ABORT.ind service primitive. The complete list of fatal error numbers is given in Annex C.

6 Local bus data exchange with DLMS

6.1 Physical layer

The *Physical-62056-3-1* protocol of the *Physical* layer of the local bus data exchange profile with DLMS is exactly the same as the one defined for the profile without DLMS.

6.2 Data Link layer

6.2.1 Link-E/D protocol

The *Link-E/D* protocol of the *Data Link* layer of the local bus data exchange profile with DLMS behaves asymmetrically. The state machine of the Primary Station is therefore different from that of the Secondary Station.

The *Data Link* layer transforms the physical channel used by the *Physical* layer into a logic channel able to transmit reliable information. Its main functions are:

- to manage directly the Bus Initialization and Forgotten Stations Call services;
- to carry out serialization and deserialization of the data (if the physical channel functions serially one bit at a time);
- to synchronize the transmission and reception of the frames;
- to filter the frames according to primary and secondary addresses;
- to ensure efficient protection against transmission errors.

6.2.2 Management of exchanges

Bus Initialization, Alarm and Forgotten Stations Call services are provided by the *Link E/D* protocol of the *Data Link* layer of the local bus data exchange profile with DLMS, but their operation takes place outside the *Application* layer. In particular, the forgotten station flag can be updated by the Application Programming Interface when a remote reading exchange occurs.

Except during opening of the communication session and Bus Initialization, Alarm report or management of Forgotten Stations Call, the protocol is perfectly symmetrical, and both stations take turns to act as Transmitter and Receiver.

After sending a frame, the *Data Link* layer of the Transmitter end always waits for a frame from the *Data Link* layer of the Receiver before transmitting again. This wait is controlled by the T1 wakeup, whose duration is 10 s.

After sending a frame and receiving an acknowledgement of the previously sent one, the current frame is retransmitted. The number of repeat transmissions is limited to MaxRetry. Above this number, the communication is stopped at the *Data Link* level and the *Application* layer is informed.

Each time a frame is received by one of the systems, an answer frame is immediately transmitted. This transmitted frame may contain data from the *Application* layer. It always contains a *Send* number and a *Confirm* number calculated according to the values of those previously sent and received. The following algorithm is used to calculate these numbers:

- if the last frame received is error-free and its *Send* number is equal to the 1's complement of the previous *Confirm* number in transmission, then the data packet is transmitted to the *Application* layer and the next frame sent will have a *Confirm* number equal to the received *Send* number. If not, the *Confirm* number is not modified and the data packet is not transmitted to the *Application* layer;
- if the last frame received is error-free and its *Confirm* number is identical to the previous *Send* number in transmission, then the *Send* number in transmission is made up to 1 for the next frame on the assumption that a new data packet has to be sent;
- if the last frame received is incorrect or if its *Confirm* number is not identical to the previous *Send* number in transmission, then the same frame is transmitted again, on condition that the number of repeat transmissions remains less than or equal to MaxRetry.

6.2.3 Data Link services and service primitives

The user of the *Link-E/D* protocol can use the services and service primitives given in Table 22.

Table 22 – Data Link services and service primitives

Service	Service primitive
DL_DATA	DL_DATA.req(Pr, DSDU) DL_DATA.ind(Pr, DSDU)
DL_IB	DL_IB.req()
DL_ASO	DL_ASO.req(DSDU) DL_ASO.ind(Collision, List)
DL_ALARM	DL_ALARM.req()
DL_ABORT	DL_ABORT.req(Strong) DL_ABORT.ind(ErrorNb)

The role assigned to each primitive is as follows:

- DL_DATA.req(Pr, DSDU) enables the *Application* layer to request the *Data Link* layer to transfer a DSDU data packet with the priority Pr ³;
- DL_DATA.ind(Pr, DSDU) enables the *Data Link* layer to inform the *Application* layer of the arrival of a DSDU data packet with the priority Pr;
- DL_IB.req() enables the ~~Application Programming Interface~~ *Support Manager* to request the *Data Link* layer to send a Bus Initialization frame;
- DL_ASO.req(DSDU) enables the ~~Application Programming Interface~~ *Support Manager* to request the *Data Link* layer to send a Forgotten Stations Call frame in accordance with a DSDU data packet;
- DL_ASO.ind(Collision, List) enables the *Data Link* layer to inform the *Application Programming Interface* of the result of a Forgotten Stations Call;
- DL_ALARM.req() enables the ~~Application Programming Interface~~ *Support Manager* to request the *Data Link* layer to transfer an alarm;
- DL_ABORT.req(Strong) enables the *Application* layer to request the *Data Link* layer to terminate its activity with the priority Strong ⁴;
- DL_ABORT.ind(ErrorNb) enables the *Data Link* layer to inform the *Application* layer of the occurrence of a fatal error identified by the number ErrorNb.

6.2.4 Data Link parameters

The value of the number of repeat transmissions for a given frame before disconnection, MaxRetry, is set to 2.

For the Primary Station, the value of the maximum number, MaxRSO, of RSO time slots for the processing of a "Forgotten Stations Call" is set to 3.

The Secondary Station shall know the list of Primary Station addresses and the list of TABi to which it has been programmed.

³ The priority level Pr differentiates the processing of emergency services such as InformationReport (level Pr=1) from that of the other DLMS services (level Pr=0).

⁴ The Strong parameter differentiates the processing of fatal errors (Strong=1) from that of the other physical disconnection requests (Strong=0) initialized by the *Application* sub-layer.