INTERNATIONAL STANDARD

IEC 62056-46

Edition 1.1

2007-02

Edition 1:2002 consolidated with amendment 1:2006

**Electricity metering –
Data exchange for meter reading,
tariff and load control –**

**Part 46:
Data link layer using HDLC protocol**

Publication numbering

As from 1 January 1997 all IEC publications are issued with a designation in the 60000 series. For example, IEC 34-1 is now referred to as IEC 60034-1.

Consolidated editions

The IEC is now publishing consolidated versions of its publications. For example, edition numbers 1.0, 1.1 and 1.2 refer, respectively, to the base publication, the base publication incorporating amendment 1 and the base publication incorporating amendments 1 and 2.

Further information on IEC publications

The technical content of IEC publications is kept under constant review by the IEC, thus ensuring that the content reflects current technology. Information relating to this publication, including its validity, is available in the IEC Catalogue of publications (see below) in addition to new editions, amendments and corrigenda. Information on the subjects under consideration and work in progress undertaken by the technical committee which has prepared this publication, as well as the list of publications issued, is also available from the following:

- **IEC Web Site (www.iec.ch)**

- **Catalogue of IEC publications**

  The on-line catalogue on the IEC web site (www.iec.ch/searchpub) enables you to search by a variety of criteria including text searches, technical committees and date of publication. On-line information is also available on recently issued publications, withdrawn and replaced publications, as well as corrigenda.

- **IEC Just Published**

  This summary of recently issued publications (www.iec.ch/online_news/ justpub) is also available by email. Please contact the Customer Service Centre (see below) for further information.

- **Customer Service Centre**

  If you have any questions regarding this publication or need further assistance, please contact the Customer Service Centre:

  Email: custserv@iec.ch
  Tel: +41 22 919 02 11
  Fax: +41 22 919 03 00

INTERNATIONAL STANDARD

IEC
62056-46

Edition 1.1

2007-02

Edition 1:2002 consolidated with amendment 1:2006

Electricity metering –
Data exchange for meter reading,
tariff and load control –

Part 46:
Data link layer using HDLC protocol

Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

PRICE CODE **XB**

For price, see current catalogue

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

ELECTRICITY METERING – DATA EXCHANGE FOR METER READING, TARIFF AND LOAD CONTROL –

Part 46: Data link layer using HDLC protocol

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 provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.

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.

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this International Standard may involve the use of a maintenance service concerning the stack of protocols on which the present standard IEC 62056-46 is based.

The IEC takes no position concerning the evidence, validity and scope of this maintenance service.

The provider of the maintenance service has assured the IEC that he is willing to provide services under reasonable and non-discriminatory terms and conditions for applicants throughout the world. In this respect, the statement of the provider of the maintenance service is registered with the IEC. Information may be obtained from:

DLMS[1] User Association
Geneva / Switzerland
www.dlms.ch

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights other than those identified above. IEC shall not be held responsible for identifying any or all such patent rights.

_____

[1] Device Language Message Specification.

International Standard IEC 62056-46 has been prepared by IEC technical committee 13: Equipment for electrical energy measurement and load control.

This consolidated version of IEC 62056 is based on the first edition (2002) [documents 13/1267/FDIS and 13/1273/RVD] and its amendment 1 (2006) [documents 13/1376/FDIS and 13/1401/RVD].

It bears the edition number 1.1.

A vertical line in the margin shows where the base publication has been modified by amendment 1.

Annexes A, B and C are for information only.

The committee has decided that the contents of the base publication and its amendments will remain unchanged until the maintenance result date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

• reconfirmed,

• withdrawn,

• replaced by a revised edition, or

• amended.

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

INTRODUCTION (to amendment 1)

The amendment takes into account that in the third edition of ISO/IEC 13239, frame type 3 has been added as Annex H.4, as requested by IEC TC 13 WG 14, and that second editions of some parts of the IEC 62056 series are under preparation.

It specifies now that a secondary station may use more than one addressing scheme.

It contains some changes concerning the negotiation of the maximum information length field HDLC parameter for better efficiency.

References have been updated and some editorial errors have also been corrected.

**ELECTRICITY METERING – DATA ECHANGE
FOR METER READING, TARIFF AND LOAD CONTROL –**

Part 46: Data link layer using HDLC protocol

1   Scope

This part of IEC 62056 specifies the data link layer for connection-oriented, HDLC-based, asynchronous communication profile.

In order to ensure a coherent data link layer service specification for both connection-oriented and connectionless operation modes, the data link layer is divided into two sub-layers: the Logical Link Control (LLC) sub-layer and the Medium Access Control (MAC) sub-layer.

This specification supports the following communication environments:

- point-to-point and point-to-multipoint configurations;
- dedicated and switched data transmission facilities;
- half-duplex and full-duplex connections;
- asynchronous start/stop transmission, with 1 start bit, 8 data bits, no parity, 1 stop bit.

Two special procedures are also defined:

- transferring of separately received Service User layer PDU parts from the server to the client in a transparent manner. The server side Service user layer can give its PDU to the data link layer in fragments and the data link layer can hide this fragmentation from the client;
- event reporting, by sending UI frames from the secondary station to the primary station.

Annex B gives an explanation of the role of data models and protocols in electricity meter data exchange.

2   Normative references

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

IEC 60050-300:2001, *International Electrotechnical Vocabulary – Electrical and electronic measurements and measuring instruments – Part 311: General terms relating to measurements – Part 312: General terms relating to electrical measurements – Part 313: Types of electrical measuring instruments – Part 314: Specific terms according to the type of instrument*

IEC/TR 62051:1999, *Electricity metering – Glossary of terms*

IEC 62051-1:2004, *Electricity metering – Data exchange for meter reading, tariff and load control – Glossary of Terms – Part 1, Terms related to data exchange with metering equipment using DLMS/COSEM*

IEC 62056-42, *Electricity metering – Data exchange for meter reading, tariff and load control – Part 42: Physical layer services and procedures for connection oriented asynchronous data exchange* [1])

IEC 62056-53:2006, *Electricity metering – Data exchange for meter reading, tariff and load control – Part 53: COSEM Application layer*

IEC 62056-61:2006, *Electricity metering – Data exchange for meter reading, tariff and load control – Part 61: OBIS Object identification system*

IEC 62056-62:2006, *Electricity metering – Data exchange for meter reading, tariff and load control – Part 62: Interface classes*

ISO/IEC 8802-2:1998, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 2: Logical link control*

ISO/IEC 13239:2002, *Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures*

3   Terms, definitions and abbreviations

3.1   Terms and definitions

For the purposes of this document, the definitions given in IEC 60050-300, IEC 62051 and IEC 62051-1 apply.

3.2   Abbreviations

APDU	Application layer Protocol Data Unit
COSEM	COmpanion Specification for Energy Metering
DISC	DISConnect (an HDLC frame type)
DL	Data Link
DM	Disconnected Mode (an HDLC frame type)
DPDU	Data link Protocol Data Unit
DSAP	Data link Service Access Point
DSDU	Data link Service Data Unit
FCS	Frame Check Sequence
FRMR	FRaMe Reject (an HDLC frame type)
HCS	Header Check Sequence
HDLC	High-level Data Link Control
I	Information (an HDLC frame type)
LLC	Logical Link Control (Sub-layer)
LSAP	LLC sub-layer Service Access Point
LPDU	LLC Protocol Data Unit
LSB	Least Significant Bit
LSDU	LLC Service Data Unit
MAC	Medium Access Control (sub-layer)
MSAP	MAC sub-layer Service Access Point (here it is equal to the HDLC address)
MSB	Most Significant Bit
MSDU	MAC Service Data Unit

———————
[1]) To be published.

NDM	Normal Disconnected Mode
NRM	Normal Response Mode
N(R)	Receive sequence Number
N(S)	Send sequence Number
P/F	Poll/Final bit
PDU	Protocol Data Unit
PH	Physical layer
PSDU	Physical layer Service Data Unit
RNR	Receive Not Ready (an HDLC frame type)
RR	Receive Ready (an HDLC frame type)
SAP	Service Access Point
SDU	Service Data Unit
SNRM	Set Normal Response Mode (an HDLC frame type)
TWA	Two Way Alternate
UA	Unnumbered Acknowledgement (an HDLC frame type)
UI	Unnumbered Information (an HDLC frame type)
UNC	Unbalanced operation Normal response mode Class
USS	Unnumbered Send Status
V(R)	Receive state Variable
V(S)	Send state Variable

4 Overview

4.1 The LLC sub-layer

In the connection-oriented profile the only role of the LLC sub-layer is to ensure consistent Data Link addressing. It can be considered that the LLC sub-layer, defined in ISO/IEC 8802-2 is used in an extended class I operation, where the LLC sub-layer provides the standard connectionless data services via a connection-oriented MAC sub-layer.

The LLC sub-layer provides Data Link (DL) connection/disconnection services to the Service User layer, but it uses the services of the MAC sub-layer to execute these services.

The LLC sub-layer is specified in clause 5.

4.2 The MAC sub-layer

The MAC sub-layer – the major part of this data link layer specification – is based on ISO/IEC 13239 concerning high-level data link control (HDLC) procedures.

This standard includes a number of enhancements compared to the original HDLC, for example in the areas of addressing, error protection and segmentation. These enhancements have been incorporated in a new frame format, which meets the requirements of the environment found in telemetry applications for electricity metering and similar industries.

The MAC sub-layer is specified in clause 6.

4.3   Specification method

Sub-layers of the data link layer are specified in terms of **services** and **protocol**.

Service specifications cover the services required of, or by, the given sub-layer at the logical interfaces with the neighbouring other sub-layer or layer, using connection oriented procedures. Services are the standard way to specify communications between protocol layers. Through the use of four types of transactions, commonly known as service primitives (Request, Indication, Response and Confirm) the service provider co-ordinates and manages the communication between the users. Using service primitives is an abstract, imple-mentation-independent way to specify the transactions between protocol layers. Given this abstract nature of the primitives, their use makes good sense for the following reasons:

• they permit a common convention to be used between layers, without regard to specific operating systems and specific languages;

• they give the implementers a choice of how to implement the service primitives on a specific machine.

Service primitives include service parameters. There are three classes of service parameters:

• parameters transmitted to the peer layer, becoming part of the transmitted frame, for example addresses, control information;

• parameters which have only local significance;

• parameters which are transmitted transparently across the data link layer to the user of the data link.

NOTE   Data link layer management services are explained in Annex C.

This standard specifies values for parameters of the first category only.

The **protocol** specification for a protocol layer includes:

• the specification of the procedures for the transmission of the set of messages exchanged between peer-layers;

• the procedures for the correct interpretation of protocol control information;

• the layer behaviour.

The protocol specification for a protocol layer does not include:

• the structure and the meaning of the information which is transmitted by means of the layer (Information field, User data subfield);

• the identity of the Service User layer;

• the manner in which the Service User layer operation is accomplished as a result of exchanging Data Link messages;

• the interactions that are the result of using the protocol layer.

5   The LLC sub-layer

5.1   The role of the LLC sub-layer

The LLC sub-layer used in this profile is based on ISO/IEC 8802-2. The presence of this sub-layer in the connection-oriented profile is somewhat artificial: the LLC sub-layer is used as a kind of protocol selector, and the 'real' data link layer connection is ensured by the MAC sub-layer. It can be considered that the standard LLC sub-layer is used in an extended class I operation, where the LLC sub-layer provides the standard data-link-connectionless services via a connection-oriented MAC sub-layer. In order to be able to establish the data link connection, the LLC sub-layer provides transparent MAC connection/disconnection services to the service user protocol layer.

5.2   Service specification for the LLC sub-layer

This subclause specifies the services required of, or by, the LLC sub-layer at the logical interfaces with the Service User layer and the MAC sub-layer, using connection-oriented procedures. As the Service User layer 'sees' the services of the LLC sub-layer as the services of the data link layer, in this standard these services are called data link layer services and the prefix "DL" to designate these services is used.

5.2.1   Setting up the Data Link Connection

Overview

Figure 1 shows the services provided by the primary station (client side) and secondary station (server side) data link layers to the service user layer for data link connection establishment.
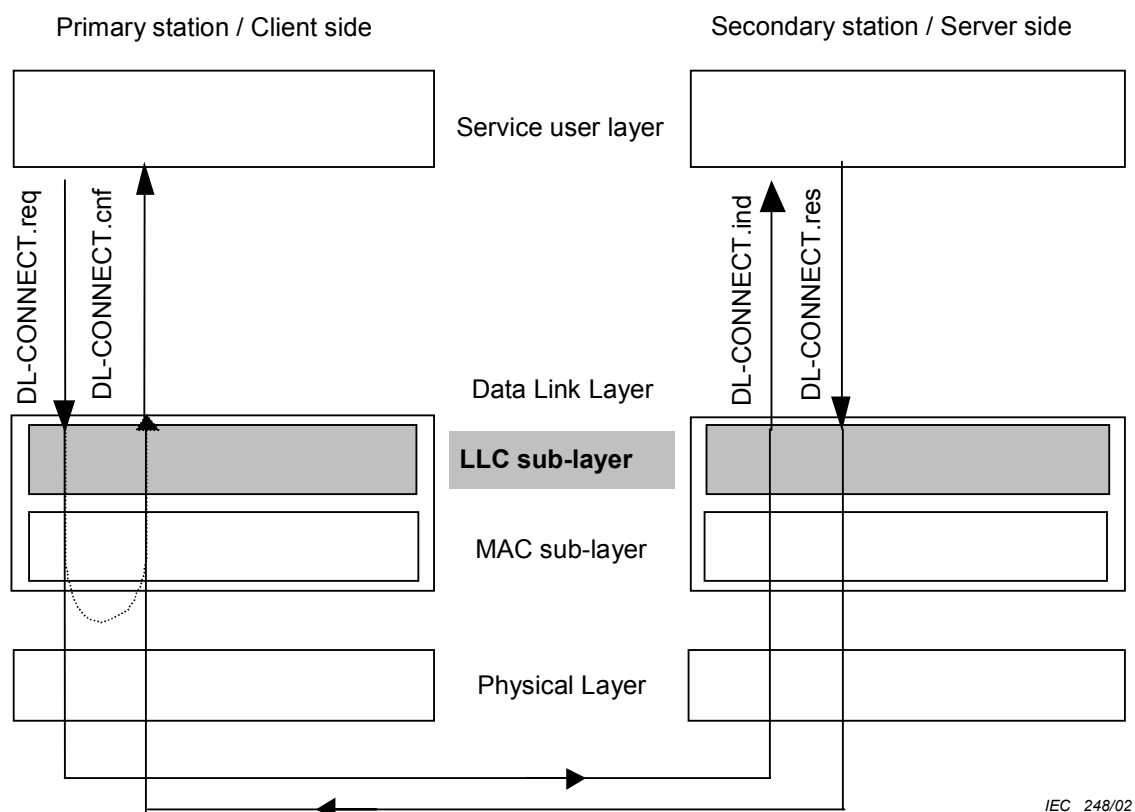
Figure 1 – Data Link (LLC) services for setting up the Data Link Connection

Data link connection establishment can only be requested by the primary station, so the DL-CONNECT.request and .confirm services are provided only at the client (primary station) side. On the other hand, the DL-CONNECT.indication and .response services are provided only at the server (secondary station) side.

The DL-CONNECT.request service primitive – in case of a locally detected error – can be also locally confirmed.

All these services are in fact, provided by the MAC sub-layer: the LLC sub-layer shall transparently transmit these services to/from the "real" service provider MAC sub-layer as the appropriate MA-CONNECT.xxx service primitive.

5.2.1.1   DL-CONNECT.request

Function

This service primitive is provided only at the client side. The Service User layer invokes this primitive to request set-up of a data link connection.

Service parameters

The semantics of the primitive is as follows:

```
DL-CONNECT.request
(
    Destination_MSAP1),
    Source_MSAP,
    User_Information
)
```

The Destination_MSAP and Source_MSAP parameters identify the referenced data link layer connection. The addressing scheme for the MAC layer is discussed in 6.4.2. The specification of the contents of the optional User_information parameter is not within the scope of this standard.

Use

The client side Service User layer entity invokes the DL-CONNECT.request primitive, when it wants to set up a connection with a peer data link layer.

5.2.1.2   DL-CONNECT.indication

Function

This service primitive is provided only at the server side. The LLC sub-layer uses this primitive to indicate to the Service User layer that the peer data link layer requested a Data Link connection.

Service parameters

The semantics of the primitive is as follows:

```
DL-CONNECT.indication
(
    Destination_MSAP,
    Source_MSAP,
    User_Information
)
```

The Destination_MSAP and Source_MSAP identify the referenced data link layer connection. The addressing scheme for the MAC layer is discussed in 6.4.2. The specification of the contents of the optional User_information parameter is not within the scope of this standard.

Use

The server side LLC sub-layer generates this primitive following the reception of an MA-CONNECT.indication primitive from the MAC sub-layer.

————————

1) MSAP in this environment is equal to the HDLC address.

5.2.1.3   DL-CONNECT.response

Function

This service primitive is provided only at the server side. The Service User layer invokes this service primitive in order to indicate to the local data link layer whether the previously proposed data link connection can be accepted by the service user layer or not.

Service parameters

The semantics of the primitive is as follows:

```
DL-CONNECT.response
(
    Destination_MSAP,
    Source_MSAP,
    Result,
    User_Information
)
```

The Destination_MSAP and Source_MSAP parameters identify the referenced data link layer connection. The Result parameter (OK, NOK, NO_RESPONSE) indicates whether the proposed connection could be accepted or not, and whether a response frame should be sent or not.

- Result == OK. This means that the received connect request can be accepted by the service user layer.

- Result == NOK. This means that the received connect request cannot be accepted by the service user layer.

- RESULT == NO_RESPONSE: This means that no response to the DL-CONNECT.indication shall be sent.

The User_Information parameter may be present only when the Result is NOK. The specification of its content is not within the scope of this standard.

NOTE   The Result parameter indicates only whether the Data Link Connection can or cannot be accepted by the service user higher layers. The data link layer itself may refuse a proposed connection, (e.g. because it supports only one connection at a given moment, thus it is not able to support a second one) even if the higher layers could accept it (Result==OK).

Use

The server side Service User layer entity invokes the DL-CONNECT.response primitive to indicate the result of a previously received request for connection.

5.2.1.4   DL-CONNECT.confirm

Function

This service primitive is provided only at the client side and it can be originated remotely or locally. The data link layer generates this primitive to indicate to the Service User layer the result of a previously received DL-CONNECT.request service.

Service parameters

The semantics of the primitive is as follows:

    DL-CONNECT.confirm
    (
        Destination_MSAP,
        Source_MSAP,
        Result,
        User_Information
)

The Destination_MSAP and Source_MSAP parameters reference data link layer connection, which is confirmed by the service. The Result parameter (OK, NOK-REMOTE, NOK-LOCAL, NO_RESPONSE) indicates the result of the previously invoked DL-CONNECT.request service.

- Result == OK. This means that the connect request was accepted by the remote station.

- Result == NOK-REMOTE. This means that the connect request was not accepted by the remote station.

- Result == NOK-LOCAL. This means that a local error has occurred, for example the service user layer tried to establish an already existing data link connection.

- RESULT == NO_RESPONSE. This means that there was no response from the remote station to the connect request.

The User_Information parameter is present only when the Result is NOK-REMOTE. The specification of its content is not within the scope of this standard.

Use

The LLC sub-layer indicates the reception of an MA-CONNECT.confirm primitive to the Service User layer by using this primitive.

5.2.2   Disconnecting the Data Link Connection

5.2.2.1   Overview

Figure 2 shows the services provided by the client and server side data link layers to the Service User layer for disconnecting a Data Link connection.

Primary station / Client side   Secondary station / Server side
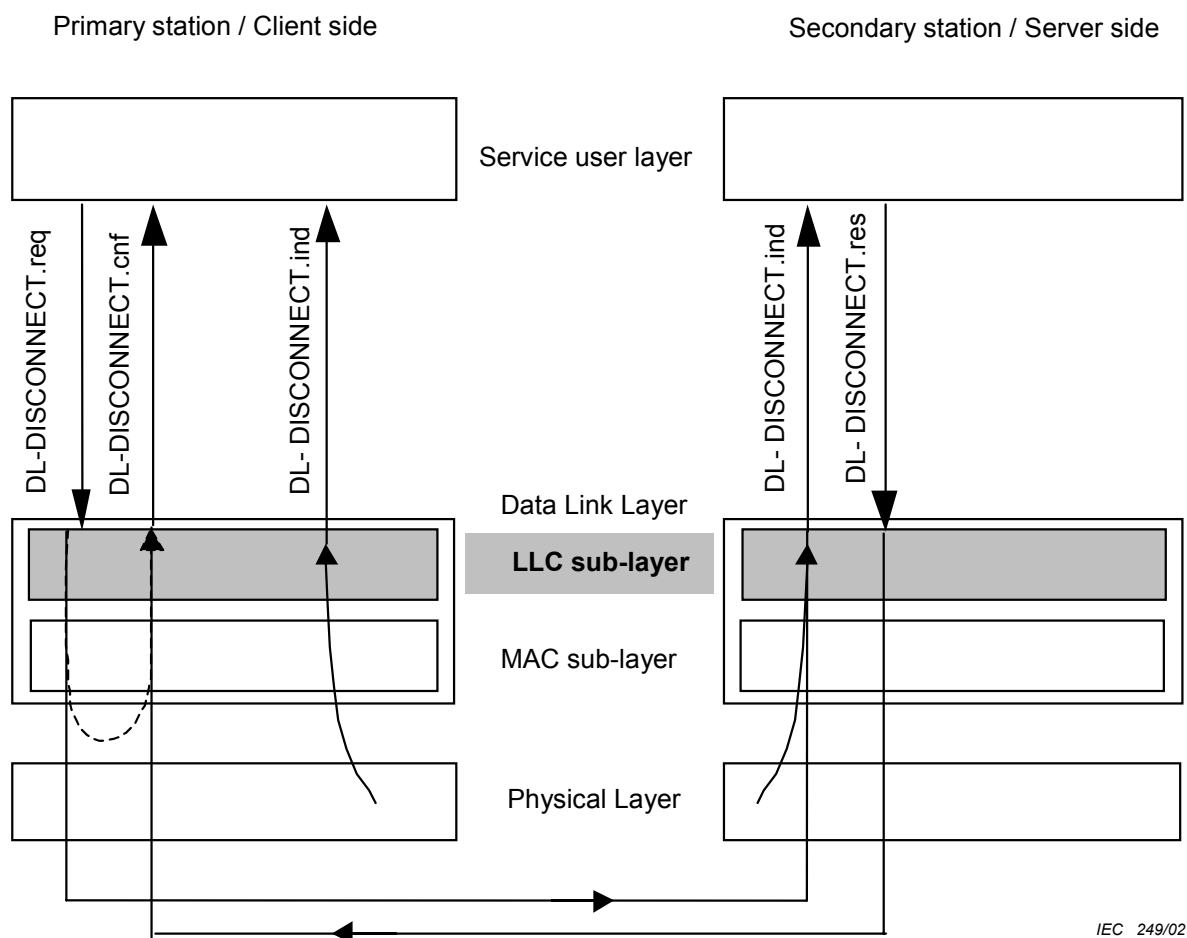
Figure 2 – Data Link (LLC) services for disconnecting the Data Link Connection

Data link disconnection can only be requested by the client device, so the DL-DISCONNECT.request and .confirm services are provided only at the client side. On the other hand, the remotely initiated (by the client) DL-DISCONNECT.indication and .response services are provided only at the server side.

NOTE   When this data link layer is used together with the COSEM application layer as defined in IEC 62056-53, DL-DISCONNECT services are used to release existing Application Associations.

Both the client and server side LLC sub-layers provide a locally initiated DL-DISCONNECT.indication service, to signal a non-solicited disconnection, due to an unexpected loss of the data link and/or physical connection.

The DL-DISCONNECT.request service primitive – in case of a locally detected error – can be also locally confirmed.

These services are in fact provided by the MAC sub-layer: the LLC sub-layer shall transparently transmit these services to/from the MAC sub-layer as the appropriate MA-DISCONNECT.xxx service primitive.

5.2.2.2   DL-DISCONNECT.request

Function

This service primitive is provided only at the client side. It is invoked by the Service User layer to request disconnecting of an existing Data Link connection.

Service parameters

The semantics of the primitive is as follows:

DL-DISCONNECT.request
(
　　Destination_MSAP,
　　Source_MSAP,
　　User_Information
)

The Destination_MSAP and Source_MSAP parameters specify the data link connection to be disconnected. The specification of the contents of the optional User_Information parameter is not within the scope of this standard.

Use

The client side Service User layer entity invokes this primitive to request a disconnection of a data link connection to peer data link layer(s).

5.2.2.3　DL-DISCONNECT.indication

Function

This service primitive is provided at the client side and at the server side.

- The server side data link layer generates this primitive to indicate to the Service User layer that the peer data link layer requests the disconnection of a data link connection.

- On both the server and client sides, this primitive is used to indicate that the data link and/or physical connection abort occurred in a non-solicited manner (e.g. the physical line has been disconnected).

Service parameters

The semantics of the primitive is as follows:

DL-DISCONNECT.indication
(
　　Destination_MSAP,
　　Source_MSAP,
　　Reason,
　　Unnumbered Send Status,
　　User_Information
)

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAPs of the terminated connection.

The Reason parameter (REMOTE, LOCAL_PHY, LOCAL_DL) indicates the reason for the DL-DISCONNECT.indication invocation.

- Reason == REMOTE. This means that the data link layer received a disconnection request from the client side. This case may happen only at the server side.

- Reason == LOCAL_DL. This means that there was a fatal data link connection failure.

- Reason == LOCAL_PHY. This means that there was a fatal physical connection failure.

The value of the USS parameter indicates whether at the moment of the DL-DISCONNECT.indication service invocation the data link layer has (USS==TRUE) or does not have (USS==FALSE) pending UI message(s).

The User_Information field may be present only when Reason == REMOTE. The specification of the contents of this parameter is not within the scope of this standard.

Use

The LLC sub-layer generates this primitive following the reception of an MA-DISCONNECT.indication primitive.

5.2.2.4   DL-DISCONNECT.response

Function

This service primitive is provided only at the server side. The Service User layer invokes this service primitive in order to indicate to the data link layer whether the previously proposed data link disconnection can be accepted by the Service User layer or not. As in this environment the server has no right to refuse the disconnection, the response depends only whether the referenced Data Link connection is existing or not.

Service parameters

The semantics of the primitive is as follows:

```
DL-DISCONNECT.response
(
    Destination_MSAP,
    Source_MSAP,
    Result
)
```

The Destination_MSAP and Source_MSAP parameters specify the remote and local MSAPs involved in the connection being disconnected. The Result parameter can have values OK, NOK and NO_RESPONSE.

- Result == OK. This means that the received disconnect request refers to an existing higher layer connection.
- Result == NOK. This means that the received disconnect request refers to a non-existing higher layer connection.
- RESULT == NO_RESPONSE: This means that no response to the DL-DISCONNECT.indication shall be sent.

Use

The server side Service User layer invokes the DL-DISCONNECT.response primitive to indicate the result of a previously received request for disconnecting the data link connection.

5.2.2.5   DL-DISCONNECT.confirm

Function

This service primitive is provided only at the client side. The data link layer generates this primitive to indicate to the Service User layer the result of a previously received DL-DISCONNECT.request service. This service can be originated remotely or locally.

Service parameters

The semantics of the primitive is as follows:

    DL-DISCONNECT.confirm
    (
        Destination_MSAP,
        Source_MSAP,
        Result
)

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAPs of the terminated connection. The Result parameter (OK, NOK, NO_RESPONSE) indicates the result of the attempt to close the data link connection.

- Result == OK. This means that the disconnect request was accepted by the remote station.

- Result == NOK. This means that the disconnect request was not accepted by the remote station.

- Result == NO_RESPONSE. This means that there was no response from the remote station to the disconnect request.

Use

The client side LLC sub-layer indicates the reception of an MA-DISCONNECT.confirm primitive to the Service User layer using this primitive.

5.2.3   Data communication

5.2.3.1   Overview

Figure 3 shows the data communication services provided by the data link layer to the Service User layer to exchange data.

Primary station / Client side   Secondary station / Server side
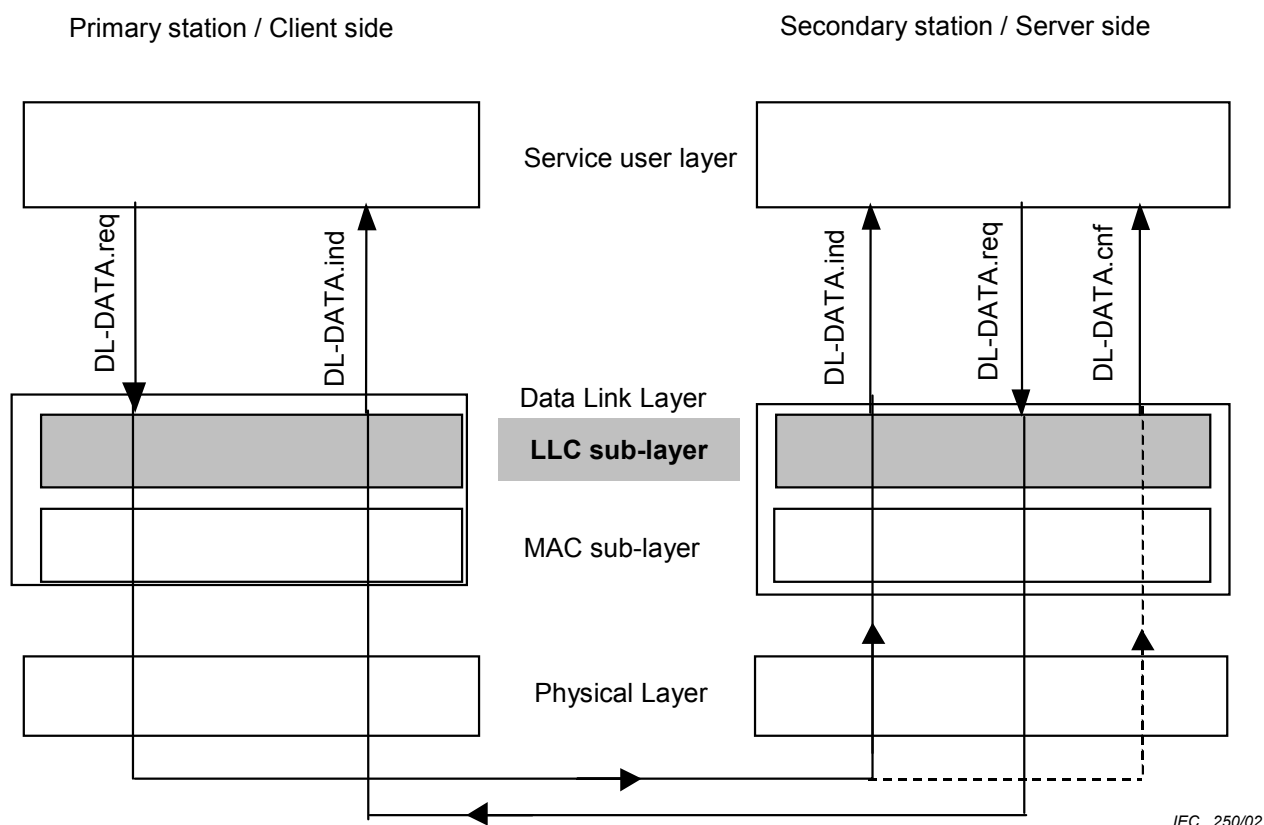
Figure 3 – Data link layer data communication services

In addition to the two standard .request and .indication services, a DL-DATA.confirm service is also provided at the server side. This service is necessary for transparent long message transfers, specified in 6.4.4.5.

5.2.3.2   DL-DATA.request

Function

The Service User layer invokes this primitive when data need to be transmitted to the peer layer entity(ies).

Service parameters

The semantics of the primitive is as follows:

```
DL-DATA.request
(
    Destination_LSAP,
    Source_LSAP,
    LLC_Quality,
    Destination_MSAP,
    Source_MSAP,
    Frame_type,
    Data
)
```

The Destination_LSAP and Source_LSAP parameters specify the referenced data link layer connection[1]. The value of the LLC_Quality parameter is used as the Control field of the PDU[2]. See also 5.3.2.

The Destination_MSAP and Source_MSAP parameters specify the remote and local MSAPs involved in the data unit transmission. The Destination_MSAP can be an individual address, a group address, or a special HDLC address (ALL_STATION, NO_STATION, etc.). Please refer to 6.4.2.4.

The Frame_Type parameter indicates for the data link layer which type of frame shall be sent. Valid frame types are different for the client and server sides. Client side valid frame types are I_COMPLETE and UI. On the server side valid frame types are I_COMPLETE, I_FIRST_FRAGMENT, I_FRAGMENT, I_LAST_FRAGMENT, and UI. See also 6.4.3.

The Data parameter contains the Service User LSDU to be transferred to the peer layer. This parameter may be empty (e.g. when Frame_type == UI, but the UI frame contains no data).

Use

The DL-DATA.request primitive is invoked by the Service User layer entity to request sending of a protocol data unit to a single peer application entity or, in the case of multicasting and broadcasting, to multiple peer application entities.

The receipt of this primitive shall cause the LLC sub-layer to append the LLC specific fields (the two LLC addresses and the LLC_Quality parameter)[3] to the received LSDU, and to pass the properly formed LPDU to the MAC sub-layer (by invoking the MA-DATA.request primitive) for transferring it to the peer LLC sub-layer.

5.2.3.3   DL-DATA.indication

Function

This primitive is used to transfer the received data from the data link layer to its Service User layer.

Service parameters

The semantics of the primitive is as follows:

    DL-DATA.indication
    (
        Destination_LSAP,
        Source_LSAP,
        LLC_Quality,
        Destination_MSAP,
        Source_MSAP,
        Frame_type,
        Data
)

---

[1] For COSEM purposes, the value of the Destination_LSAP parameter is constant and equal to E6$_H$. The value of the Source_LSAP parameter is E6H or E7H depending on whether the Data field corresponds to a command or to a response.

[2] For COSEM purposes, the value of the LLC_Quality parameter is set to 0, and reserved for future use.

[3] When Frame_type == I_FRAGMENT or I_LAST_FRAGMENT, no LLC specific field is appended to the LSDU.

The Destination_LSAP and Source_LSAP parameters specify the referenced data link layer connection. The value of the LLC_Quality parameter is used as the Control field of the PDU. See also 5.3.2.

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAPs involved in the data unit transmission. The Destination_MSAP can be an individual address, a group address, or a special HDLC address (ALL_STATION, NO_STATION, etc.). Please refer to 6.4.2.4.

The Frame_Type parameter indicates for the Service User layer which type of frame has been received. Valid frame types are I_COMPLETE and UI. See also 6.4.3.

The Data parameter contains the Service User layer protocol data unit sent by the peer.

Use

The DL-DATA.indication primitive is used to indicate to the Service User layer entity the receipt of a protocol data unit from a peer layer entity.

This primitive is generated following the reception of an MA-DATA.indication service issued by the local MAC sub-layer. First, the LLC sub-layer shall check the LLC addresses. If these are correct, it shall remove the LLC specific fields (the two LLC addresses and the LLC_Quality parameter) from the received LPDU, and shall pass the remaining, properly formatted LSDU to the service user protocol layer with the help of the DL-DATA.indication service primitive. Otherwise the received LPDU shall be discarded.

5.2.3.4   DL-DATA.confirm

Function

This service primitive is provided only at the server side. The data link layer generates this primitive to indicate to the Service User layer the result of a previously received DL-DATA.request service, when this .request service has been invoked with Frame_type = I_FIRST_FRAGMENT, I_FRAGMENT or I_LAST_FRAGMENT. The DL-DATA.confirm service primitive is generated when the previously requested LSDU has been successfully sent to the peer data link layer.

Service parameters

The semantics of the primitive is as follows:

```
DL-DATA.confirm
(
    Destination_LSAP,
    Source_LSAP,
    Destination_MSAP,
    Source_MSAP,
    Frame_type,
    Result
)
```

The Destination_LSAP and Source_LSAP parameters identify the referenced data link layer connection.

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAP-s involved in the data unit transmission.

The Frame_Type parameter indicates the type of the frame which is confirmed. Valid frame types are I_FIRST_FRAGMENT, I_FRAGMENT and I_LAST_FRAGMENT.

The value of the Result parameter indicates the result of the transmission of the received LSDU. Possible values are OK and NOK.

Use

The server side LLC sub-layer indicates the reception of an MA-DATA.confirm primitive to the Service User layer by using this primitive.

5.3   Protocol specification for the LLC sub-layer

5.3.1   Overview

The LLC sub-layer specification is based on LLC type 1, specified in ISO/IEC 8802-2, which provides a data-link-connectionless-mode service across a data link with minimum protocol complexity. The presence of this sub-layer in this connection-oriented profile is somewhat artificial: the LLC sub-layer is used as a kind of a protocol selector, and the 'real' data link layer functionality is ensured by the MAC sub-layer.

The standard LLC frame format is shown in Figure 4. The LLC frame format for this standard is specified in 5.3.2.

Destination (remote) LSAP	Source (local) LSAP	Control	Information
8 bits	8 bits	8 or 16 bits	n*8 bits

IEC   251/02

Figure 4 – The ISO/IEC 8802-2 LLC protocol data unit format

The LLC sub-layer shall transparently transmit the Information Field between the Service User layer and the MAC sub-layer.

When a DATA.request service invocation is received from the service user protocol layer, the LLC sub-layer, if required, shall append the LLC specific fields (the two LLC addresses and the LLC_Quality parameter) to the LSDU. When an MA-DATA.indication service invocation is received from the MAC sub-layer, it shall check and remove these LLC specific fields from the received LPDU.

5.3.2   LLC protocol data unit (LPDU) structure

The only role of the LLC sub-layer is to select the Service User layer protocol; the selection is done based on the Destination_LSAP and the Source-LSAP addresses. The Control byte is referred to the LLC_Quality parameter in the LLC service primitives.

The LLC protocol data unit is as follows[1]):

8 bit	8 bit	8 bit	n * 8 bit
Destination (remote) LSAP	Source (local) LSAP	Quality	LLC+1 Layer PDU
			octets of LLC+1 Layer PDU

IEC   252/02

Figure 5 – The used LLC protocol data unit format

---

[1] In COSEM, the value of the Destination_LSAP is 0xE6. The value of the Source_LSAP is 0xE6 or 0xE7. The last bit is used as a flag: z=0 means 'command' and z=1 means "response". The Control byte (LLC_Quality) is reserved for future use and its value must always be 0x00.

The destination LSAP 0xFF is used for broadcasting purposes. Devices in this environment shall never send messages with this broadcast address, but they shall accept messages containing this broadcast destination address as if it would be addressed to them.

5.3.3   State transition tables for the LLC sub-layer

As the role of the LLC sub-layer is limited to protocol selection, its state-transition diagram is quite simple: after being initialized, the LLC sub-layer enters into its only stable state, the IDLE state, and shall return into this state after any possible events. The state transitions of the client side LLC sub-layer is shown in Table 1:

Table 1 – State transition table of the client side LLC sub-layer

Current state	Event	Action	Next state
IDLE	DL-CONNECT.request	Invoke MA-CONNECT.request	IDLE
IDLE	Receive MA-CONNECT.confirm	Generate DL-CONNECT.confirm	IDLE
IDLE	DL-DISCONNECT.request	Invoke MA-DISCONNECT.request	IDLE
IDLE	Receive MA-DISCONNECT.indication	Generate DL-DISCONNECT.indication	IDLE
IDLE	Receive MA-DISCONNECT.confirm	Generate DL-DISCONNECT.confirm	IDLE
IDLE	DL-DATA.request	Add LLC addresses and control byte (3 bytes) to the received LSDU; Invoke MA-DATA.request; [2]	IDLE

IDLE	Receive MA-DATA.indication	Check LLC addresses (3 bytes); [2]
		
		If address==O.K
		  {
		    remove LLC addresses;
		    generate DL-DATA.indication;
		  }
		else
		  {
		    discard the received packet;
		  }
			IDLE

The state transitions of the server side LLC sub-layer are shown in Table 2.

---

[2] : LLC specific fields are not present, when Frame_type == I_FRAGMENT or I_LAST FRAGMENT. See Footnote 3 in 5.2.3.2.

Table 2 – State transition table of the server side LLC sub-layer

Current state	Event	Action	Next state
IDLE	Receive MA-CONNECT.indication	Generate DL-CONNECT.indication	IDLE
IDLE	DL-CONNECT.response	Invoke MA-CONNECT.response	IDLE
IDLE	Receive MA-DISCONNECT.indication	Generate DL-DISCONNECT.indication	IDLE
IDLE	DL-DISCONNECT.response	Invoke MA-DISCONNECT.response	IDLE
IDLE	DL-DATA.request and Frame_type is I_COMPLETE, UI or I_FIRST_FRAGMENT	Add LLC addresses and control byte to the received LSDU; Invoke MA-DATA.request;	IDLE
IDLE	DL-DATA.request and Frame_type is I_FRAGMENT or I_LAST_FRAGMENT	Invoke MA-DATA.request	IDLE

IDLE	Receive MA-DATA.indication	Check LLC addresses;
		
		If address==OK
		  {
		    remove LLC addresses;
		    generate DL-DATA.indication;
		  }
		else
		  {
		    discard the received packet;
		  }
			IDLE

IDLE	Receive MA-DATA.confirm	Generate DL-DATA.confirm	IDLE

6 The MAC sub-layer

The MAC sub-layer is based on ISO/IEC 13239.

The MAC sub-layer – similarly to the LLC sub-layer – is specified in terms of **services** and **protocols**. As the MAC sub-layer behaviour is quite complex, some aspects of the service invocation handling are discussed in the service specification part, although these are normally part of the protocol specification.

6.1 HDLC selections

For the purpose of this standard, the following selections from the HDLC standard ISO/IEC 13239 have been made:

- unbalanced connection-mode data link operation[1];
- two-way alternate data transfer;
- the selected HDLC class of procedure is UNC, extended with UI frames;
- frame format type 3;
- non-basic frame format transparency.

In the unbalanced connection-mode data link operation, two or more stations are involved. The primary station assumes responsibility for the organization of data flow and for unrecoverable data link level error conditions by sending command and supervisory frames. The secondary station(s) respond by sending response frames.

---

[1] In COSEM environment, the choice of an unbalanced mode of operation is natural: it is the consequence of the fact that communication in this environment is based on the Client/Server paradigm.

The basic repertoire of commands and responses of the UNC class of procedures is extended with the UI frame to support multicasting and broadcasting and non-solicited information transfer from server to the client.

Using the unbalanced connection-mode data link operation implies that the client and server side data link layers are different in terms of the sets of HDLC frames and their state machines.

6.2 Service specification for the MAC sub-layer

This subclause specifies the services required of, or by, the MAC sub-layer at the logical interfaces with the service user and the Physical (PH) layers, using connection-oriented procedures. As the client and the server side MAC sub-layers are different, services are specified for both sides.

6.2.1 Setting up the MAC connection

6.2.1.1 Overview

Figure 6 shows the services provided by the client and server side MAC sub-layers to the service user layer for MAC connection establishment.
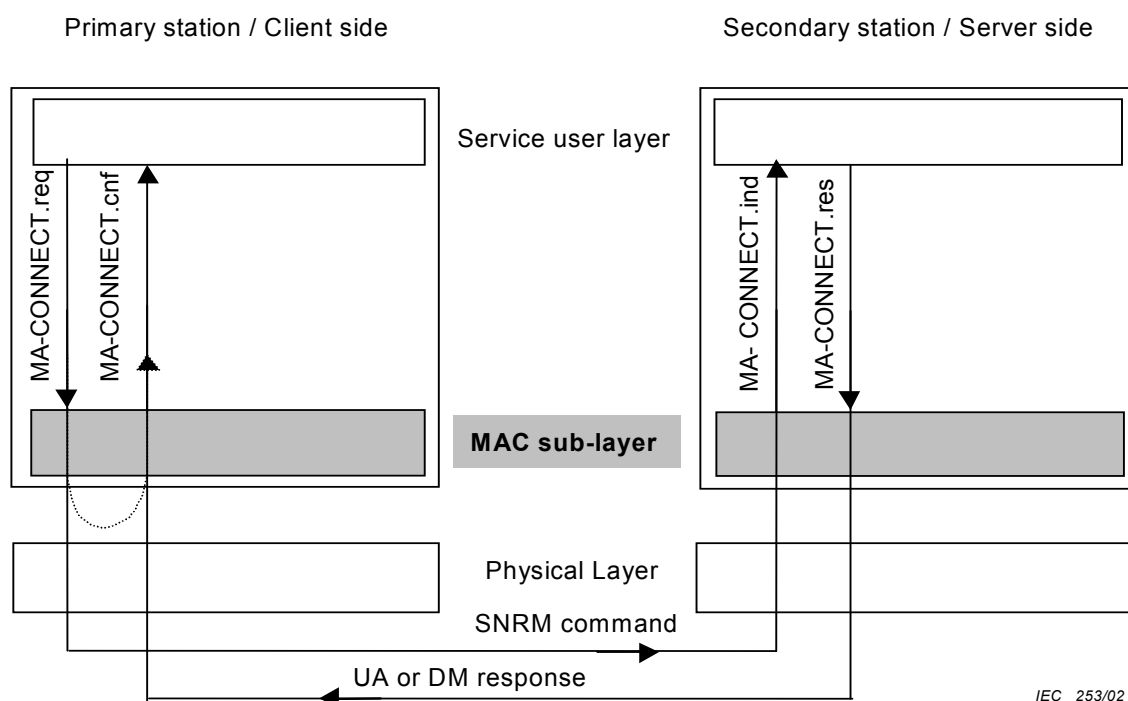
Figure 6 – MAC sub-layer services for setting up the MAC (DL) connection at the client and server sides

As data link connection establishment can only be requested by the client device, the MA-CONNECT.request and .confirm services are provided only at the client side. On the other hand, the corresponding MA-CONNECT.indication and .response services are provided only at the server side. The MA-CONNECT.request service primitive, in case of a locally detected error, can also be locally confirmed.

6.2.1.2   MA-CONNECT.request

Function

This service primitive is provided only at the client side. The service user layer invokes this primitive to request the set-up of an MAC connection. Upon reception of this primitive, the MAC layer shall send out a correctly formatted SNRM[1] frame.

Service parameters

The semantics of the primitive is as follows:

        MA-CONNECT.request
        (
            Destination_MSAP,
            Source_MSAP,
            User_Information
)

The Destination_MSAP and Source_MSAP identify the referenced data link layer connection. The addressing scheme for the MAC sub-layer is discussed in 6.4.2.

The User_Information parameter – if present – shall be inserted in the User data subfield of the Information field of the SNRM frame. The specification of the contents of this parameter is not within the scope of this standard.

Use

The client side Service User layer entity requesting the set-up of an MAC connection with a peer MAC layer invokes this primitive.

NOTE   If the secondary station receives an SNRM frame with the address parameters of an already existing data link connection, it shall respond with an UA frame and the send and receive state variables shall be set to zero. See also 6.4.3.6.

6.2.1.3   MA-CONNECT.indication

Function

This primitive is provided only at the server side. The MAC sub-layer, following the reception of an SNRM frame shall invoke this service primitive, in order to indicate to the service user layer that the peer MAC sub-layer requested the establishment of an MAC Connection.

Service parameters

The semantics of the primitive is as follows:

        MA-CONNECT.indication
        (
            Destination_MSAP,
            Source_MSAP,
            User_Information
)

The Destination_MSAP and Source_MSAP parameters identify the referenced data link layer connection. The User_Information parameter – if present – shall carry the contents of the User data subfield of the Information field of the SNRM frame received. The specification of the contents of this parameter is not within the scope of this standard.

---

[1]   SNRM is the HDLC frame used for demanding an MAC connection establishment, see 6.4.3.6.

Use

The server side MAC sub-layer indicates the reception of a correctly formatted SNRM frame to the service user protocol layer using this primitive.

6.2.1.4   MA-CONNECT.response

Function

This primitive is provided only at the server side. The Service User layer invokes this service primitive in order to indicate to the remote client whether the previously proposed MAC connection is accepted by the server or not.

Service parameters

The semantics of the primitive is as follows:

    MA-CONNECT.response
    (
        Destination_MSAP,
        Source_MSAP,
        Result,
        User_Information
)

The Destination_MSAP and Source_MSAP parameters identify the referenced MAC connection. The Result parameter (OK, NOK, NO_RESPONSE) indicates whether the proposed connection could be accepted or not, and whether a response frame should be sent or not.

• Result == OK. This means that the received connect request can be accepted by the service user layer.

• Result == NOK. This means that the received connect request cannot be accepted by the service user layer.

• RESULT == NO_RESPONSE: This means that no response to the MA-CONNECT .indication shall be sent.

The User_Information parameter may be present only when the Result is NOK. It shall be inserted in the User data subfield of the Information field of the DM frame. The specification of its content is not within the scope of this standard.

NOTE   The Result parameter indicates only whether the MAC connection can or cannot be accepted by the service user higher layers. The MAC sub-layer itself may refuse a proposed connection (e.g. because it supports only one connection at a given moment, thus it is not able to support a second one), even if the higher layers could accept it (Result==OK).

Use

The MA-CONNECT.response primitive is invoked by the Service User layer entity to indicate the result of a previously received request for connection.

6.2.1.5   MA-CONNECT.confirm

Function

This primitive is provided only at the client side. The MAC sub-layer invokes this primitive in order to indicate to the Service User layer the result of a previously received MA-CONNECT.request service.

Service parameters

The semantics of the primitive is as follows:

> MA-CONNECT.confirm
> (
>   Destination_MSAP,
>   Source_MSAP,
>   Result,
>   User_Information
>)

The Destination_MSAP and Source_MSAP parameters hold the local and remote MSAPs of the MAC connection, which is confirmed by the service. The Result parameter (OK, NOK-REMOTE, NOK-LOCAL, NO_RESPONSE) indicates the result of the previously invoked MA-CONNECT.request service.

- Result == OK. This means that the connect request was accepted by the remote station.

- Result == NOK-REMOTE. This means that the connect request was not accepted by the remote station.

- Result == NOK-LOCAL. This means that a local error has occurred, for example the service user layer tried to establish an already existing data link connection.

- Result == NO_RESPONSE. This means that there was no response from the remote station to the connect request.

The User_Information parameter shall be present only when the Result is NOK-REMOTE. It shall carry the contents of the User data subfield of the Information field of the DM frame received. The specification of the contents of this parameter is not within the scope of this standard.

Use

The client side MAC sub-layer invokes this primitive in order to indicate to the service user protocol layer the result of a previously received MA-CONNECT.request service.

6.2.2 Disconnecting the MAC connection

6.2.2.1 Overview

Figure 7 shows services provided by the client and server side MAC sub-layers to the Service User layer for disconnecting the MAC connection.

Primary station / Client side   Secondary station / Server side
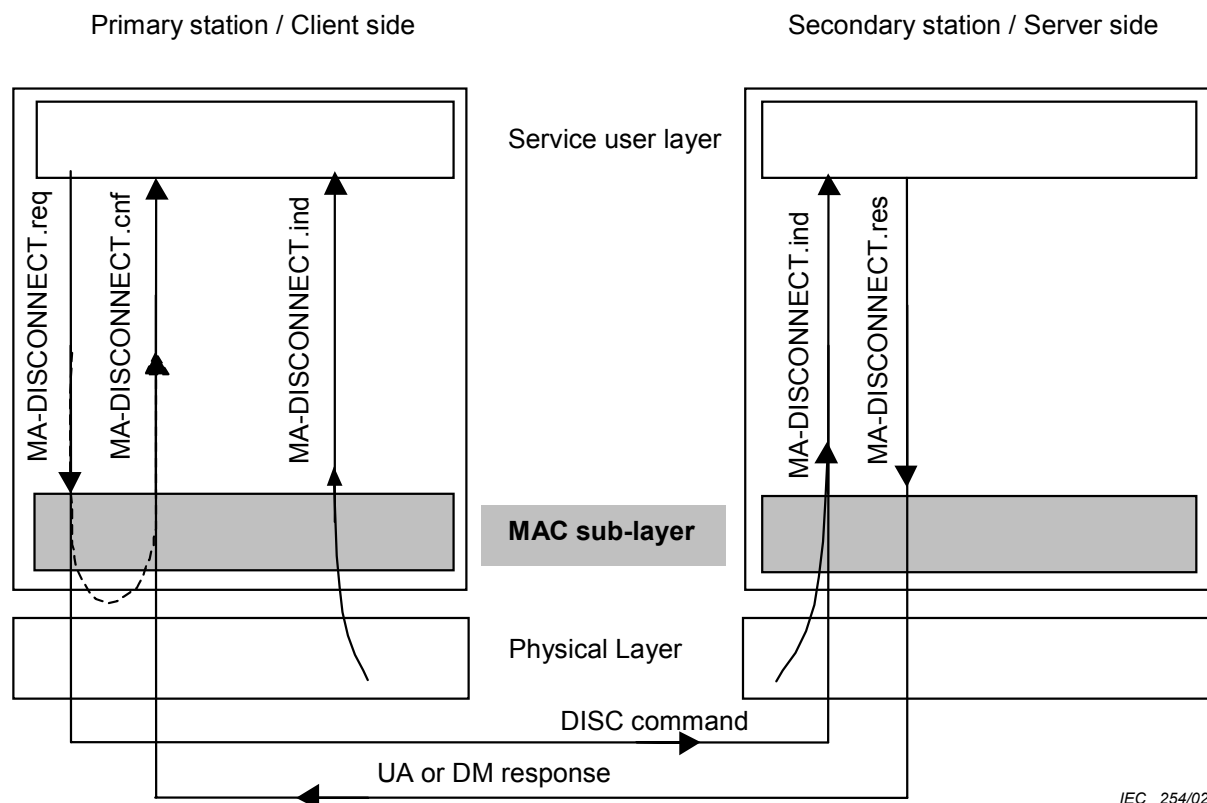
**Figure 7 – MAC sub-layer services for disconnecting the MAC (DL) connection
at the client and server sides**

As MAC disconnection can only be requested by the client device, the MA-DISCONNECT.request and .confirm services are provided only at the client side. On the other hand, the remotely initiated (by the client) MA-DISCONNECT.indication and the MA.DISCONNECT.response services are provided only at the server side.

Both the client and server side MAC sub-layers provide a locally initiated MA-DISCONNECT.indication service, to signal a non-solicited disconnection, due to an unexpected loss of the MAC or physical connection. Physical connection loss is indicated locally by the PH-ABORT.indication service and MAC connection loss is locally detected at the MAC layer.

The MA-DISCONNECT.request service primitive, in case of a locally detected error, can also be locally confirmed.

6.2.2.2   MA-DISCONNECT.request

Function

This service primitive is invoked by the client side Service User layer to request the disconnection of an existing MAC connection.

Service parameters

The semantics of the primitive is as follows:

    MA-DISCONNECT.request
    (
        Destination_MSAP,
        Source_MSAP,
        User_Information
)

The Destination_MSAP and Source_MSAP parameters indicate the MAC connection to be disconnected. The User_Information parameter, if present, shall be inserted in the User data subfield of the Information field of the DISC frame to be sent. The specification of the contents of this parameter is not within the scope of this standard.

Use

The client side Service User layer invokes the MA-DISCONNECT.request primitive to request for disconnection of the MAC connection.

6.2.2.3   MA-DISCONNECT.indication

Function

The server side MAC sub-layer generates this primitive to indicate to the service user layer that the peer MAC sub-layer requests the disconnection of an MAC connection. On both the server and client sides, this primitive is also used to indicate that the MAC or physical connection abort occurred in a non-solicited manner (e.g. the physical line has been disconnected).

Service parameters

The semantics of the primitive is as follows:

    MA-DISCONNECT.indication
    (
        Destination_MSAP,
        Source_MSAP,
        Reason,
        Unnumbered Send Status,
        User_Information
)

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAPs of the connection to be terminated.

The Reason parameter indicates whether the originator of the MA-DISCONNECT.indication invocation is a received DISC frame (Reason == REMOTE), or a PH –DISCONNECT .indication service (Reason == LOCAL_PH)., or a local data link error (Reason == LOCAL_DL). The first case (Reason == REMOTE) may happen only at the server side.

The value of the USS parameter indicates whether at the moment of the MA-DISCONNECT.indication service invocation the MAC sub-layer has (USS==TRUE) or does not have (USS==FALSE) pending UI message(s).

NOTE The number of pending UI frames is a layer parameter, which can be accessed using the layer management services.

The User_Information field may be present only when Reason == REMOTE and in this case it shall carry the contents of the User data subfield of the Information field of the DISC frame received.

The specification of the contents of this parameter is not within the scope of this standard.

Use

When the server side MAC sub-layer receives a correctly formatted DISC frame, it shall invoke this service primitive with Reason == REMOTE. After invoking the service, the server MAC layer is waiting for an MA_DISCONNECT.response from the Service User layer indicating that the Service User layer has accepted the request.

NOTE   The Service User layer cannot refuse this request, but it may indicate that no response should be sent.

Both the server and client MAC sub-layers shall invoke this primitive with Reason == LOCAL_PH after receiving a Ph-ABORT.indication service primitive from the physical layer, meaning that the physical connection has been interrupted. In this case, the User_ Information parameter is not present.

6.2.2.4   MA-DISCONNECT.response

Function

This primitive is provided only at the server side. The Service User layer invokes this service primitive in order to indicate to the MAC layer whether the previously proposed MAC disconnection can be accepted by the Service User layer or not. As in this environment, the server has no right to refuse the disconnection, the response depends on only whether the referenced connection is existing (UA)[1] or non-existing (DM).

Service parameters

The semantics of the primitive is as follows:

```
MA-DISCONNECT.response
(
    Destination_MSAP,
    Source_MSAP,
    Result
)
```

The Destination_MSAP and Source_MSAP specify the remote and local MSAPs involved in the connection being disconnected. Depending on the value of the Result parameter (OK, NOK or NO_RESPONSE) the MAC layer sends a UA or a DM message or nothing to the remote client. The following cases are possible:

- Result == OK. This means that the received disconnect request refers to an existing higher layer connection, which should be disconnected. In this case, the server MAC sub-layer shall enter into the disconnected state and send a UA message to its peer sub-layer.

- Result == NOK. This means that the received disconnect request has attempted to disconnect an MAC connection which does not correspond to an existing higher layer connection. In this case, the server MAC sub-layer, depending on the fact that the data link connection is existing or not, shall enter into (remain in) a disconnected state and shall send a UA or a DM message to its peer as appropriate.

- RESULT == NO_RESPONSE: This means that no response to the MA-DISCONNECT.indication shall be sent.

---

[1]   UA and DM are the appropriate HDLC frames to be sent following the invocation of the DL-DISCONNECT.res (MA-DISCONNECT.res) service primitive.

Use

The server side Service User layer entity invokes the MA-DISCONNECT.response primitive to indicate the result of a previously received request for disconnection.

6.2.2.5   MA-DISCONNECT.confirm

Function

The client side MAC sub-layer generates this primitive in order to indicate to the service user protocol layer (via the LLC sub-layer) the result of a previously received MA-DISCONNECT.request service.

Service parameters

The semantics of the primitive is as follows:

```
MA-DISCONNECT.confirm
(
    Destination_MSAP,
    Source_MSAP,
    Result
)
```

The Destination_MSAP and Source_MSAP parameters, specify the local and remote MSAPs of the terminated connection. The Result parameter (OK, NOK, NO_RESPONSE) indicates the result of the attempt to close the MAC connection.

- Result == OK. This means that the disconnect request was accepted by the remote station.

- Result == NOK. This means that the disconnect request was not accepted by the remote station.

- Result == NO_RESPONSE. This means that there was no response from the remote station to the disconnect request.
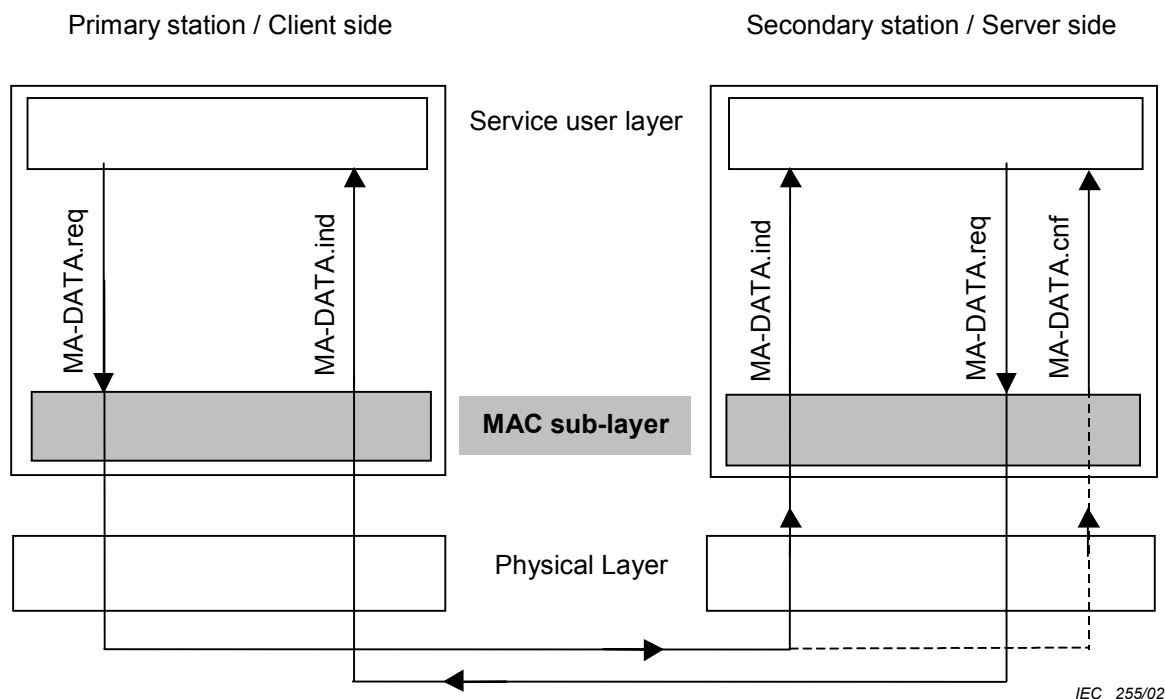
Use

This primitive is generated by the client side MAC sub-layer entity in order to indicate to the service user protocol layer the result of a previously received MA-DISCONNECT.request service.

6.2.3   Data communication

6.2.3.1   Overview

Figure 8 shows data communication services provided by the MAC sub-layer to the Service User layer to exchange data with the peer, using I frames or in a disconnected manner (UI frames).

Primary station / Client side   Secondary station / Server side

Service user layer

MA-DATA.req   MA-DATA.ind   MA-DATA.ind   MA-DATA.req   MA-DATA.cnf

MAC sub-layer

Physical Layer

IEC 255/02

Figure 8 – MAC sub-layer data communication services

Basically, for data exchange services the client and the server sides provide the same service set. However, in addition to the two standard .request and .indication services, an MA-DATA.confirm service is also provided at the server side. This service is necessary for transparent long message transfers. See also 6.4.4.5.

The client and server MAC sub-layers behave differently with regard to the MA-DATA.request primitive: In the Normal Response Mode used, the server station may initiate transmission only as the result of receiving an explicit permission to do so from the client station. After giving the permission to talk to the server, the client has also to wait (with a specific Time-out, TO_WAIT_RESP, see 6.4.4.10.1.) for the server to give back explicitly this permission before initiating a new transmission. If no response is received however, the client shall re-gain the permission to talk at the end of this time-out period.

6.2.3.2   MA-DATA.request

Function

The Service User layer invokes this primitive to initiate a data transfer to its peer layer.

Service parameters

The semantics of the primitive is as follows:

    MA-DATA.request
    (
        Destination_MSAP,
        Source_MSAP,
        Frame_type,
        Data
)

The Destination_MSAP and Source_MSAP parameters specify the remote and local MSAPs involved in the data unit transmission. The Destination_MSAP can be an individual address, a group address, or a special HDLC address (ALL_STATION, NO_STATION, etc., see 6.4.2.4).

The Frame_Type parameter indicates for the MAC sub-layer which type of HDLC frame shall be sent. Valid frame types are different for the client and server sides. Client side valid frame types are I_COMPLETE and UI. On the server side, valid frame types are I_COMPLETE, I_FIRST_FRAGMENT, I_FRAGMENT, I_LAST_FRAGMENT, and UI. See also 6.4.3.

The Data parameter contains the protocol data unit (MSDU) to be transferred to the peer layer. This parameter may be empty (e.g. when Frame_type == UI, but the UI frame contains an empty Information Field).

Use

The MA-DATA.request service primitive is invoked by the Service User layer entity whenever data need to be transmitted to a single peer entity or, in the case of multicasting and broad-casting, to multiple peer entities.

Procedures for Data exchange are detailed in 6.4.4.4.3.4.

6.2.3.3   MA-DATA.indication

Function

This primitive is used to transfer the received data from the MAC sub-layer to its Service User layer.

Service parameters

The semantics of the primitive is as follows:

    MA-DATA.indication
    (
        Destination_MSAP,
        Source_MSAP,
        Frame_type,
        Data
)

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAPs involved in the data unit transmission. The Destination_MSAP can be an individual address, a group address, or a special HDLC address (ALL_STATION, NO_STATION, etc., see 6.4.2.4).

The Frame_Type parameter indicates to the Service User layer the type of the HDLC frame received. Valid frame types are at both the client and the server sides: I_COMPLETE and UI. An I_COMPLETE frame shall be reported only if it has been received within an established MAC connection.

The Data parameter contains the Service User layer protocol data unit (MSDU) received from the peer layer. This parameter may be empty (e.g. when trame_Type == UI, but the UI frame contains an empty Information Field).

Use

The MA-DATA.indication is passed from the MAC sub-layer entity to the Service User layer entity or entities to indicate the arrival of an MPDU from a remote MAC entity to the local MAC entity.

Procedures for Data exchange are detailed in 6.4.4.4.3.4 and in annex A.

6.2.3.4   MA-DATA.confirm

Function

This service primitive is provided only at the server side. The server side MAC sub-layer generates this primitive in order to indicate to the Service User layer the result of a previously received MA-DATA.request service, when this .request service has been invoked with Frame_type = I_FIRST_FRAGMENT, I_FRAGMENT or I_LAST_FRAGMENT. The MA-DATA.confirm service primitive is generated when the previously requested MSDU has been successfully sent (following the receipt of the positive acknowledgement after sending out the last HDLC frame) to the peer MAC sub-layer.

Service parameters

The semantics of the primitive is as follows:

```
MA-DATA.confirm
(
    Destination_MSAP,
    Source_MSAP,
    Frame_type,
    Result
)
```

The Destination_MSAP and Source_MSAP parameters specify the local and remote MSAPs involved in the data unit transmission. The Destination_MSAP shall be an individual address, a group address, or a special HDLC address (ALL_STATION, NO_STATION, etc., see 6.4.2.4).

The Frame_Type parameter indicates the type of the HDLC frame, which is confirmed. Valid frame types are I_FIRST_FRAGMENT, I_FRAGMENT and I_LAST_FRAGMENT.

The value of the Result parameter indicates the result of the transmission of the received MSDU. Possible values are OK and NOK.

Use

The MAC sub-layer indicates the result of a previously received MA-DATA.request service, when this .request service has been invoked with Frame_type = I_FIRST_FRAGMENT, I_FRAGMENT or I_LAST_FRAGMENT. These frame types are used with a special procedure specified for transferring long messages from the server to the client. This procedure is described in 6.4.4.5.

6.3   Physical layer services used by the MAC sub-layer

6.3.1   Overview

Figure 9 shows services provided by the physical layer to the MAC sub-layer. The same service set is used both at the client and the server sides.

IEC   256/02
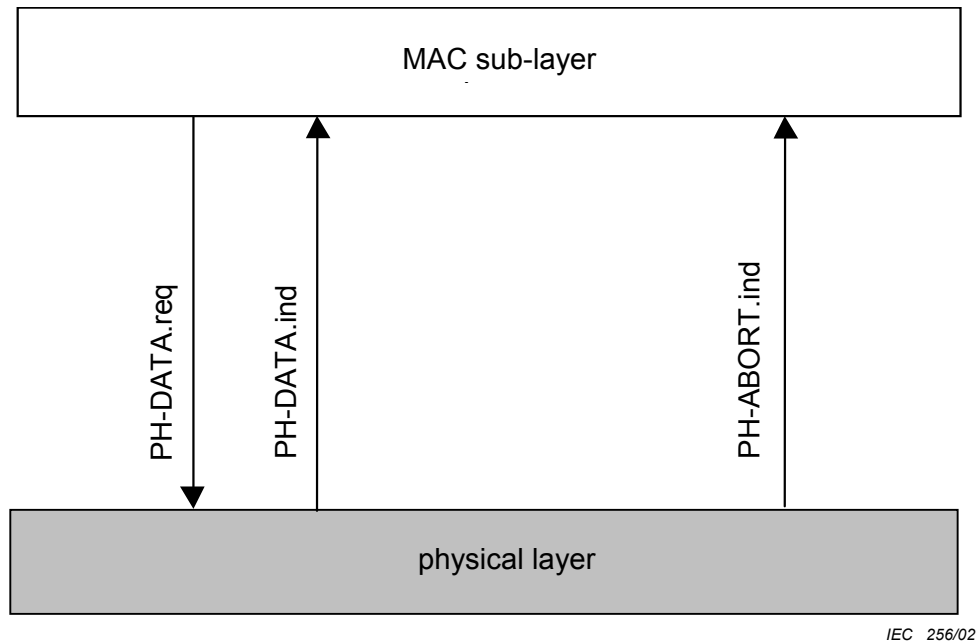
Figure 9 – Physical layer services used by the MAC sub-layer

6.3.2   Setting up a physical link

Setting up the physical link does not utilize services of protocol layers above the physical layer.

6.3.3   Disconnecting the physical link

Disconnecting the physical link does not utilize services of protocol layers above the physical layer. The only service used by another protocol layer is the PH-ABORT.indication service (IEC 62056-42), to initiate an MA-DISCONNECT.indication service indication with Reason == LOCAL_PH.

6.3.4   Data communication

The MAC sub-layer uses the PH-DATA.request and PH-DATA.indication service primitives of the physical layer for exchanging data with the remote device.

6.4   Protocol specification for the MAC sub-layer

This subclause specifies the protocol of the MAC sub-layer based on ISO/IEC 13239.

6.4.1   The MAC PDU and the HDLC frame

6.4.1.1   Overview

The MAC sub-layer uses the HDLC frame format type 3 as defined in Clause H.4 of ISO/IEC 13239: 2002.

HDLC frame format type 3 is as follows:

Flag	Frame format	Dest.address	Src. address	Control	HCS	Information	FCS	Flag

IEC   257/02

Figure 10 – MAC sub-layer frame format (HDLC frame format type 3)

This frame format is used in those environments where additional error protection, identification of both the source and the destination, and/or longer frame sizes are needed. Type 3 requires the use of the segmentation subfield, thus reducing the length field to 11 bits. Frames that do not have an information field, for example as with some supervisory frames, or an information field of zero length do not contain an HCS and an FCS, only an FCS. The HCS and FCS polynomials will be the same. The HCS shall be 2 octets in length.

The elements of the frame are described in the following subclauses.

6.4.1.2   Flag field

The length of the flag field is one byte and its value is $7E_H$. When two or more frames are transmitted continuously, a single flag is used as both the closing flag of one frame and the opening flag of the next frame, as it is shown in Figure 11.

NOTE   Frames are transmitted continuously when the period of time between two transmitted characters does not exceed the specified max. inter-octet time. See 6.4.4.3.3.

Flag	Frame I	Flag	Frame I+1	Flag	Frame I+2	Flag

IEC   258/02

Figure 11 – Multiple frames

6.4.1.3   Frame format field

The length of the frame format field is two bytes. It consists of three sub-fields referred to as the format type sub-field (4 bit), the segmentation bit (S, 1 bit) and the frame length sub-field (11 bit), as it is shown in Figure 12:

MSB   LSB

1	0	1	0	S	L	L	L	L	L	L	L	L	L	L	L

Format type   Frame length sub-field
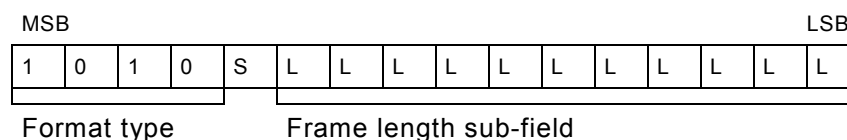
IEC   259/02

Figure 12 – The frame format field

The value of the format type sub-field shall be 1010 (binary), which identifies a frame format type 3 as defined in 6.4.1.1.

The value of the frame length subfield is the count of octets in the frame excluding the opening and closing frame flag sequences.

Rules of using the segmentation bit are defined in 6.4.4.4.3.6.

6.4.1.4   Destination and source address fields

There are exactly two address fields in this frame: a destination and a source address field. The HDLC address extension mechanism described in 4.7.1. of the HDLC standard ISO/IEC 13239 shall be applied to both address fields.

6.4.1.5   Control field

The length of the control field is one byte. The control field indicates the type of commands or responses, and contains sequence numbers, where appropriate (frames I, RR and RNR). See also 6.4.3.2.

6.4.1.6   Header check sequence (HCS) field

The length of the HCS field is two bytes.

HCS is calculated for the bytes of the header, excluding the opening flag and the HCS itself. The HCS is calculated in the same way as the frame check sequence (FCS). Frames that do not have an information field contain only an FCS (the HCS in this case is considered as FCS). Guidelines to calculate HCS (and FCS) are given in annex A.

6.4.1.7   Information field

The information field may be any sequence of bytes. In the case of data frames (I and UI frames), it carries the MSDU .

6.4.1.8   Frame checking sequence (FCS) field

The length of the FCS field is two bytes. FCS is calculated for the entire length of the frame, excluding the opening flag and the FCS itself. Unless otherwise noted, the frame checking sequence is calculated for the entire length of the frame, excluding the opening flag, the FCS and any start and stop elements. Guidelines to calculate HCS (and FCS) are given in annex A.

6.4.2   MAC addressing

6.4.2.1   Use of extended addressing

Extended addressing according to 4.7.1 of ISO/IEC 13239 shall be used.

The address field can be extended by reserving the first transmitted bit (low-order) of each address octet which would then be set to binary zero to indicate that the following octet is an extension of the address field. The format of the extended octet(s) shall be the same as that of the first octet. Thus, the address field may be recursively extended. The last octet of an address field is indicated by setting the low-order bit to binary one.

When an extension is used, the presence of a binary "1" in the first transmitted bit of the first address octet indicates that only one address octet is being used. The use of an address extension thus restricts the range of the single address octets to 128.

6.4.2.2   Address field structure

The address fields of the HDLC frame format type 3 (see 6.4.1.1) contain two addresses: a destination HDLC address and a source HDLC address. Depending on the direction of the data exchange, both the client and the server addresses can be destination or source addresses.

Client addresses shall always be expressed on one byte. The use of address extension restricts the range of client addresses to 128.
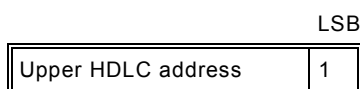
On the server side, to enable addressing more than one logical device within a single physical device and to support the multi-drop configuration, the HDLC address may be divided into two parts.[1] One part – the so-called **upper HDLC address** – shall be used to address a Logical Device (a separately addressable entity within a physical device), and the second part – the **lower HDLC address** – shall be used to address a Physical Device (a physical device on the multi-drop). Although the upper HDLC address shall always be present, the lower HDLC address can be left out when it is not required.

The HDLC extended addressing mechanism shall be applied to both address fields. This address extension specifies variable length address fields, but for the purpose of this protocol, the length of a complete HDLC address field is restricted to be one, two or four bytes long, as follows:
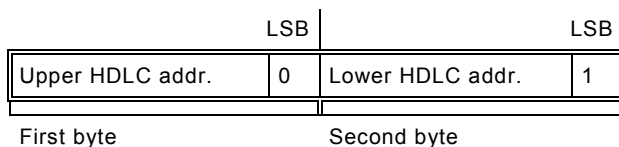
- One byte: only the upper HDLC address is present.

- Two bytes: one byte upper HDLC address and one byte lower HDLC address is present.

- Four bytes: two bytes of upper HDLC address and two bytes of lower HDLC address are present.

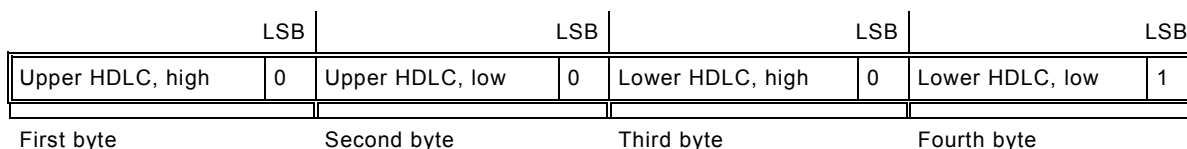These three cases are illustrated in the following figures.

Address structure for the 1-byte case:

	LSB
Upper HDLC address	1

Address structure for the 2-byte case:

	LSB		LSB
Upper HDLC addr.	0	Lower HDLC addr.	1

First byte   Second byte

Address structure for the 4-byte case:

	LSB		LSB		LSB		LSB
Upper HDLC, high	0	Upper HDLC, low	0	Lower HDLC, high	0	Lower HDLC, low	1

First byte   Second byte   Third byte   Fourth byte

The variable length HDLC address structure reserves one bit per byte to signal whether the given byte is the last one or another byte follows. This means that the address range for one byte addresses is 0..0x7F and for two byte addresses is 0..0x3FFF.

Individual, multicast and broadcast addressing facilities are provided for both the upper and the lower HDLC address level.

---

[1] As the address of the Server may be either the source or the destination address, this division may apply to both the source and the destination address fields.

6.4.2.3   Reserved special HDLC addresses

The following special HDLC addresses are reserved:

Table 3 – Table of reserved client addresses

Reserved HDLC addresses	
0x00	NO_STATION Address
0x01	Client Management Process
0x10	public client (lowest security level)
0x7F	ALL_STATION (Broadcast) Address

Table 4 – Table of reserved server addresses

Reserved upper HDLC addresses		
One byte address	Two byte address	
0x00	0x0000	NO_STATION Address
0x01	0x0001	Management Logical Device Address
0x02..0x0F	0x0002..0x000F	Reserved for future use
0x7F	0x3FFF	ALL_STATION (Broadcast) Address
Reserved lower HDLC addresses		
0x00	0x0000	NO_STATION Address
0x01...0x0F	0x0001..0x000F	Reserved for future use
0x7E	0x3FFE	CALLING[1] Physical Device Address

In the table above, the LSBs reserved for the address extension mechanism are not taken into account. For example, an HDLC frame, sent from a client to a server with the following addresses:

Client HDLC Address = $3A_H$ = 00111010_B

Server HDLC Address (using four bytes addressing)

   lower HDLC Address = $3FFF_H$ = 0011111111111111_B ALL_STATION (Broadcast) Address

   upper HDLC Address = 1234_H = 0001001000110100_B

---

[1]   The meaning of the CALLING Physical Device is discussed in 6.4.2.4

The address fields of the message shall contain the following octets:

Server address				Client address
Upper HDLC high	Upper HDLC low	Lower HDLC high	Lower HDLC low	HDLC address
LSB	LSB	LSB	LSB	LSB
0 1 0 0 1 0 0 0	0 1 1 0 1 0 0 0	1 1 1 1 1 1 1 0	1 1 1 1 1 1 1 1	0 1 1 1 0 1 0 1
First byte	Second byte	Third byte	Fourth byte	Fifth byte
Destination address				Source address

6.4.2.4   Handling special addresses

The following MAC address types and specific MAC addresses are specified:

- individual Addresses;
- group Addresses;
- the CALLING station Address;
- the ALL_STATION Address;
- the NO_STATION Address;
- the Management Logical Device Address (the presence of this device is mandatory).

The following rules apply:

- group Address management is not within the scope of this specification;
- the Source Address field of a valid HDLC frame may not contain either the ALL_STATION or the NO_STATION Address. If an HDLC frame is received with it, it shall be considered as an invalid frame;
- only HDLC frames transmitted from the primary station towards the secondary station(s) may contain the ALL_STATION or the NO_STATION in the Destination Address Field;
- broadcast and multicast I frames shall be discarded;
- the P/F bit of messages with ALL_STATION, NO_STATION or Group address in the Destination Address field shall be set to FALSE. UI frames addressed to the ALL_STATION, NO_STATION or to a Group address with P==TRUE shall be discarded;
- the CALLING station address is a special physical device address, to support event reporting; see 6.4.4.7. It shall be reserved to reference the server station initiating a physical connection to the client station. It is not that station's own physical address, therefore no station shall be configured to have the CALLING address as its own physical address.

6.4.2.5   Handling inopportune address lengths

Frames received at the server side may contain addresses which are expressed in different length than the addressing scheme used within the receiving device. In such cases, the following rules apply:

- As client addresses in this protocol shall be expressed in one byte, if the incoming frame at the server side contains more than one byte in the Source Address field, the frame shall be discarded.
- Destination addresses (DA) shall be handled according to Table 5.

Table 5 – Handling inopportune address lengths

Length of the received DA field	Length of the own address	Behaviour
1 byte	2 bytes	The received message shall be discarded
1 byte	4 bytes	The received message shall be discarded
2 bytes	1 byte	The received message is not discarded if the received lower MAC Address is equal to the ALL_STATION address only. In this case, the message shall be sent to the Logical Device(s) designated by the upper MAC Address field
2 bytes	4 bytes	In this case, the value of the received one-byte lower and upper MAC addresses shall be converted in the receiver into a two+two byte address, and the received message shall be taken into account as if it was received in using a 4-byte length Destination Address field
4 bytes	1 byte	The received message is not discarded if the received lower and upper MAC Addresses are both equal to the ALL_STATION address only.
4 bytes	2 bytes	The received message may not be discarded if the received lower MAC Address is equal to the ALL_STATION or to the CALLING physical device address only. In the first case, the frame shall be accepted only if the upper MAC Address is equal to the ALL_STATION address, in the second case the message shall be taken into account only if the upper MAC Address is equal to the Management Logical Device Address and the CALLING DEVICE layer parameter is set to TRUE. In any other case, the received frame shall be discarded
Three bytes or more than 4 bytes	N.A.	The frame shall be discarded
NOTE The server may support more than one addressing scheme.		

6.4.3   Command and response frames

6.4.3.1   Selected repertoire

This standard uses the UNC basic repertoire of commands and responses, extended with the UI commands and responses, as defined in ISO/IEC 13239.

Table 6 – Command and response frames

Commands	Responses
I	I
RR	RR
RNR	RNR
SNRM	UA
DISC	DM
UI	UI
	FRMR

6.4.3.2   Control field format

The encoding of the command/response frame control fields shall be modulo 8, as specified in 5.5 of ISO/IEC 13239) and in the table below:

Table 7 – Control field format

	MSB							LSB
I	R	R	R	P/F	S	S	S	0
RR	R	R	R	P/F	0	0	0	1
RNR	R	R	R	P/F	0	1	0	1
SNRM	1	0	0	P	0	0	1	1
DISC	0	1	0	P	0	0	1	1
UA	0	1	1	F	0	0	1	1
DM	0	0	0	F	1	1	1	1
FRMR	1	0	0	F	0	1	1	1
UI	0	0	0	P/F	0	0	1	1

Whereas **RRR** is the receive sequence number N(R), **SSS** is the send sequence number N(S) and **P/F** is the poll/final bit.

NOTE In this notation, the bit order is inverted compared to the notation in ISO/IEC 13239. However, in both notations, the LSB is the first bit transmitted. (For transmission order see 6.4.4.2.2.)

6.4.3.3   Information transfer command and response

The function of the information, I command and response is to transfer sequentially numbered frames, each containing an information field.

The I frame control field shall contain two sequence numbers:

a) N(S), which shall indicate the sequence number associated with the I frame; and

b) N(R), which shall indicate the sequence number (as of the time of transmission) of the next expected I frame to be received, and consequently shall indicate that the I frames numbered up to N(R) – 1 inclusive have been received correctly.

For data integrity reasons, in this profile, the default value of the maximum information field length – receive and maximum information field length – transmit HDLC parameters is 128 bytes. Other values may be negotiated at connection establishment time, see 6.4.4.4.3.2.

NOTE 1 In order to ensure a minimal performance, the master station should offer at least a max_info_ field_length_receive of 128 bytes.

NOTE 2 The maximum value of the information field length is 2030 bytes.

6.4.3.4   Receive ready command and response

The RR frame shall be used by a data station to

a) indicate that it is ready to receive an I frame(s); and

b) acknowledge previously received I frames numbered up to N(R) – 1 inclusive.

When transmitted, the RR frame shall indicate the clearance of any busy condition that was initiated by the earlier transmission of an RNR frame by the same data station.

6.4.3.5   Receive not ready command and response

The RNR, frame shall be used by a data station to indicate a busy condition, i.e. temporary inability to accept subsequent I frames. I frames numbered up to N(R) − 1 inclusive shall be considered as acknowledged. The I frame numbered N(R) and any subsequent I frames received, if any, shall not be considered as acknowledged; the acceptance status of these frames shall be indicated in subsequent exchanges.

6.4.3.6   Set normal response mode (SNRM) command

The SNRM command shall be used to place the addressed secondary station in the normal response mode (NRM) where all control fields shall be one octet in length. The secondary station shall confirm acceptance of the SNRM command by transmission of a UA response at the first respond opportunity. Upon acceptance of this command, the secondary station send and receive state variables shall be set to zero.

When this command is actioned, the responsibility for all unacknowledged I frames assigned to data link control reverts to a higher layer. Whether the content of the information field of such unacknowledged I frames is reassigned to data link control for transmission or not is decided at a higher layer.

The SNRM command may contain an optional information field that is used for negotiation of data link parameters (see 6.4.4.4.3.1) and to carry user information transported transparently across the link layer to the user of the data link.

6.4.3.7   Disconnect (DISC) command

The DISC command shall be used to terminate an operational or initialization mode previously set by a command. In both switched and non-switched networks, it shall be used to inform the addressed secondary station(s) that the primary station is suspending operation and that the secondary station(s) should assume a logically disconnected mode. Prior to actioning the command, the secondary station shall confirm the acceptance of the DISC command by the transmission of a UA response.

When this command is actioned, the responsibility for all unacknowledged I frames assigned to data link control reverts to a higher layer. Whether the content of the information field of such unacknowledged I frames is reassigned to data link control for transmission or not is decided at a higher layer.

An information field may be present in the DISC command.

6.4.3.8   Unnumbered acknowledge (UA) response

The UA response shall be used by the secondary station to acknowledge the receipt and acceptance of SNRM and DISC commands.

The UA response may contain an optional information field that is used for negotiation of data link parameters (see 6.4.4.4.3.1).

6.4.3.9   Disconnected mode (DM) response

The DM response shall be used to report a status where the secondary station is logically disconnected from the data link, and is, by system definition, in NDM.

The DM response shall be sent by the secondary station in NDM to request the primary/other combined station to issue a mode setting command, or if sent in response to the reception of a mode setting command, to inform the primary station that it is still in NDM and cannot action the mode setting command. An information field may be present in the DM response.

A secondary station in NDM shall monitor received commands to detect a respond opportunity in order to (re)transmit the DM response, i.e. no commands (other than UI commands) are accepted until the disconnected mode is terminated by the receipt of a mode setting command (SNRM).

6.4.3.10   Frame reject (FRMR) response

The FRMR response shall be used by the secondary station in an operational mode to report that one of the following conditions which is not correctable by retransmission of the identical frame resulted from the receipt of a frame without FCS error from the primary station:

- the receipt of a command or a response that is undefined or not implemented,

- the receipt of an I/UI command or response, with an information field which exceeded the maximum information field length which can be accommodated by the secondary/ combined station,

- the receipt of an invalid N(R) from the primary/combined station, i.e. an N(R) which identifies an I frame which has previously been transmitted and acknowledged or an I frame which has not been transmitted and is not the next sequential I frame awaiting transmission; or

- the receipt of a frame containing an information field when no information field is permitted by the associated control field.

The secondary station shall transmit the FRMR response at the first respond opportunity.

An information field that provides the reason for the frame rejection shall be included (see 5.5.3.4.2 of ISO/IEC 13239).

6.4.3.11   Unnumbered information (UI) command and response

The UI command shall be used to send information to a secondary station(s) without affecting the V(S) or V(R) variables at any station. Reception of the UI command is not sequence number verified by the data link procedures; therefore, the UI frame may be lost if a data link exception occurs during transmission of the command, or duplicated if an exception condition occurs during any reply to the command. There is no specified secondary station response to the UI command. The UI command may be sent independently of the mode of the data link station (NDM or NRM).

6.4.4   Elements of the procedures

6.4.4.1   Overview

When the physical link is established between the primary and the secondary stations, but there is no active data link channel established, both the client and the server side MAC sub-layers are in NDM. In this mode, no information or numbered supervisory frames are transmitted or shall be accepted. In this mode, the secondary station capability is limited to:

- accepting and responding to SNRM commands;

- accepting a UI command;

- transmitting a UI response at a respond opportunity;

- responding with a DM response to a received disconnect (DISC) command.

When an MAC connection is established, the MAC layer operates in the NRM. The secondary station (the server) shall initiate data transmission only as a result of receiving explicit permission to do so from the primary station (the client). After receiving permission (POLL BIT == TRUE), the secondary station shall initiate a response transmission. The response transmission may consist of one or more frames, while maintaining active data link channel state (see 6.4.4.3). The last frame of the response transmission shall be explicitly indicated by the secondary station (FINAL BIT == TRUE). Following the indication of the last frame, the

secondary station shall stop transmitting until explicit permission is again received from the primary station.

6.4.4.2   Transmission considerations

6.4.4.2.1   Transparency

Subclause 4.3 of ISO/IEC 13239 specifies multiple transparency mechanisms.

For the purpose of this protocol, the non-basic frame format transparency mechanism has been selected, as it is specified in 4.3.4 of that standard.

When using the non-basic frame format with the frame format field, the length sub-field obviates the need for the bit or octet insertion methods to achieve transparency. Consequently, no control octet transparency (octet insertion) is used in this MAC sub-layer operation.

6.4.4.2.2   Order of bit and octet transmission

Addresses, commands, responses, sequence numbers and data link information within information fields shall be transmitted with the low-order bit first.

Fields, which carry values expressed in more than one octet, shall be transmitted with the most significant octet first. For example, if an address field is expressed in two octets with the value 0x1234, the most significant octet (0x12) will be transmitted first, followed by the least significant one (0x34). 16-bit HCS and FCS shall be transmitted to the line commencing with the coefficient of the highest term[1] (corresponding to x^{15}) also.

6.4.4.2.3   Invalid frame

An invalid frame is defined as one that is not properly bounded by two flags, or one that is too short (that is shorter than sevent octets between flags using the 16-bits FCS), or one in which octet framing is violated (e.g. an "0" bit occurs when the stop-bit is expected), or one containing either an ALL_STATION or NO_STATION address in its Source Address field.

Invalid frames shall be ignored.

6.4.4.3   HDLC channel states

6.4.4.3.1   Active HDLC channel state

A HDLC channel is in active state when the primary station or a secondary station is actively transmitting a byte of the frame or an inter-octet fill. In active state, the right to continue transmission shall be reserved.

6.4.4.3.2   Abort sequence

In this standard, the non-basic transparency is used, therefore the Abort Sequence HDLC feature cannot be used. The abort sequence is specified (see 5.1.1.2 of ISO/IEC 13239) as a two-octet sequence, starting with the control escape octet followed by a closing flag octet. Receipt of this sequence is interpreted as an abort and the receiving data station shall ignore the frame.

---

[1]   Annex A gives an example for FCS calculation.

6.4.4.3.3   Start/stop transmission inter-octet time-out

The inter-octet time-out (T_{io}) for start/stop transmission is an optional time-out used for recovering from situations in which excessive periods of time elapse between transmitted octets within a frame. This time-out function (or equivalent) only applies to a frame that is being received. The time-out function (or equivalent) is started once the stop bit of an octet is detected and stopped upon receipt of the start bit of the next octet or when the time-out function (or equivalent) runs out. Whenever this time-out occurs in the receiver, the end of the actually received frame shall be assumed and the data stream shall be scanned for the next opening flag sequence.

NOTE   The value of T_{io} depends on the media used. For PSTN connection T_{io} = 25 ms.

6.4.4.3.4   Idle HDLC channel state

A Data Link channel is in idle state when a continuous mark-hold condition persists for a system specific period of time (T_{idle}).

NOTE   The value of T_{idle} depends on the media used. For PSTN connection T_{idle} = 25 ms.

6.4.4.4   HDLC channel operation – Description of the procedures

6.4.4.4.1   General

For the purpose of this standard, unbalanced connection-mode data link operation has been selected. An unbalanced data link involves one primary station and one or more secondary station(s). The primary station shall be ultimately responsible for overall data link error recovery.

Subclause 5.2 of ISO/IEC 13239 defines three operational and three non-operational modes. For the purpose of this standard, the NRM (5.2.1.1 of ISO/IEC 13239) and the NDM (5.2.2.1 of ISO/IEC 13239) have been selected.

Each data station shall check for the correct receipt of the I frames it has sent to the remote data station by checking the N(R) of each numbered information and supervisory frames received.

6.4.4.4.2   Data station characteristics

The primary station is responsible for:

- setting up the data link and disconnecting the data link;
- sending information transfer, supervisory and unnumbered commands; and
- checking received responses.

The secondary station shall be responsible for:

- checking received commands;
- sending information transfer, supervisory and unnumbered responses as required by the received commands.

6.4.4.4.3   Definition of the procedures

6.4.4.4.3.1   Setting up the data link

The primary station shall initialize the HDLC link with the secondary station by sending an SNRM command and shall start a response time-out function (see 6.4.4.10.1). The addressed secondary station, upon receiving the SNRM command correctly, shall send the UA response at its first opportunity, and shall set its send and receive state variables to zero. If the UA response is received correctly, the HDLC link set up to the addressed secondary station is complete, and the primary station shall set its send and receive state variables relative to that secondary station to zero and shall stop the response time-out function.

If, upon receipt of an SNRM command the secondary station determines that it cannot enter the indicated mode, it shall send the DM response. If the DM response is received correctly, the primary station shall stop the response time-out function.

If the SNRM command, UA response or DM response is not received correctly, it shall be ignored. The result will be that the primary station's response time-out function will run out, and the primary station may re-send the SNRM command and restart the response time-out function.

This action may continue until an UA or DM response has been received correctly or until the SNRM frame is transmitted MAX_NB_OF_RETRIES times (see 6.4.4.10.2). Any further recovery action takes place at a higher layer.

6.4.4.4.3.2   HDLC parameter negotiation during the connection phase

The SNRM/UA message exchange allows not only the establishment of the connection, but also to negotiate some data link parameters. ISO/IEC 13239 specifies a set of negotiable parameters. For the purpose of this protocol, a subset of these parameters has been selected. This subset of the negotiable HDLC parameters contains two elements:

the WINDOW_SIZE parameter;

the MAXIMUM_INFORMATION_FIELD_LENGTH parameter.

The default values for these parameters are as follows:

default WINDOW_SIZE = 1;

default MAXIMUM_INFORMATION_FIELD_LENGTH = 128 (80_H).

Rules for the negotiation are as follows:

The negotiation starts with the SNRM frame. This frame may contain an Information Field. When the Information Field is present, it shall be formatted as follows (see 5.5.3.2 of ISO/IEC 13239):

Example of an information field encoding (the parameter values representing the default):

81	80	12	05	01	80	06	01	80	07	04	00	00	00	01	08	04	00	00	00	01

where

81_H	format identifier
80_H	group identifier
12_H	group length (18 octets)
05_H	parameter identifier (maximum information field length – transmit)
01_H	parameter length (1 octet)
80_H	parameter value

06$_H$	parameter identifier (maximum information field length – receive)
01$_H$	parameter length (1 octet)
80$_H$	parameter value
07$_H$	parameter identifier (window size, transmit)
04$_H$	parameter length (4 octets)
00$_H$	parameter value (high byte of value)
00$_H$	parameter value
00$_H$	parameter value
01$_H$	parameter value (low byte of value)
08$_H$	parameter identifier (window size, receive)
04$_H$	parameter length (4 octets)
00$_H$	parameter value (high byte of value)
00$_H$	parameter value
00$_H$	parameter value
01$_H$	parameter value (low byte of value)

In multi-octet fields (as in this example), the highest order bits are in the first octet transmitted (see 6.4.4.2.2).

When the Information Field is present, the first two bytes – Format Identifier (81$_H$) and Group Identifier (80$_H$) – shall also always be present. On the other hand, any (one or more) of the negotiable parameters may be absent. The absence of a particular parameter (PV=0 or PI/PL/PV missing, see 5.5.3.1.2 of ISO/IEC 13239) shall be interpreted to mean default values.

The absence of the Information Field shall be interpreted to mean default values for each parameter.

Besides these negotiable HDLC parameters, the Information Field of an SNRM frame may also contain a User data subfield.

Including other parameters in the Information Field implies the rejection of the SNRM frame. In this case, a DM frame (with *"Incorrect Parameter list within the Information Field of the SNRM frame"*) shall be sent to the primary station.

The presence of the user-defined parameter sub-field shall not imply the rejection of the frame.

If the secondary station receives a correct SNRM frame, and the requested connection can be accepted, it shall respond with a UA frame. This UA frame shall carry the result of the HDLC parameter negotiation.

The result shall be calculated by the secondary station by choosing the smaller value between the proposed value of a parameter (sent with the SNRM frame) and the value of the corresponding parameter at the secondary station, received with the UA frame.

In case, when the maximum information field length – transmit and maximum information field length – receive parameters are represented on two bytes – this is the case when version 1 of the IEC HDLC setup class is used, see IEC 62056-62 – the group length will be 14$_H$ (20 octets) and the parameter length of parameters 5 and 6 will be 02$_H$. An example is shown below:

81	80	14	05	02	00	80	06	02	00	80	07	04	00	00	00	01	08	04	00	00	00	01

Example

Table 8 – Example for parameter negotiation values with the SNRM/UA frames

Proposed parameter/*value* (by the received SNRM frame)	Value of the corresponding parameter, which can be potentially used by the secondary station	Result (transmission/reception direction from the point of view of the secondary station)
Max_info_field – transmit (05_H)/ 80_H (128_D)	Max_info_field – receive (06_H)/ 40_H (64_D)	Max_info_field – receive (06_H)/ 40_H (64_D)
Max_info_field – receive (06_H)/ 80_H (128_D)	Max_info_field – transmit (05_H)/ 80_H (128_D)	Max_info_field – transmit (05_H)/ 80_H (128_D)
Window size – transmit (07_H)/ 0001_H (1_D)	Window size – receive (08_H)/ 0007_H (7_D)	Window size – receive (08_H)/ 0001_H (1_D)
Window size – receive (08_H)/ 0007_H (7_D)	Window size – transmit (07_H)/ 0007_H (7_D)	Window size – transmit (07_H)/ 0007_H (7_D)
NOTE 1 The incoming SNRM frame in the above case shall include an Information Field, but in this field only the presence of the 'Window size – receive' parameter is mandatory, because the values for the other parameters are considered as default values.		
NOTE 2 For reasons of communication efficiency, the value of the parameter maximum information field length-receive proposed by the primary station should be at least as big as the value of the parameter of the maximum information field length–transmit supported by the secondary station.		

The response UA frame will also include an Information Field. This Information Field shall contain the 'Max_info_field – receive' (40_H), and the 'Window size – transmit' (07_H) parameters. The other two parameters are not necessarily present in the response frame, because they carry the default values (see 6.4.4.10.5 for Maximum information field length and 6.4.4.10.6 for Window size).

After the successful exchange of SNRM/UA frames with the above parameters, both sides shall consider that the HDLC parameters for the established HDLC connection are as follows:

- Max_info_field for transmission client (primary station) $\rightarrow$ server (secondary station): **40_H**
- Max_info_field for transmission Server (secondary station) $\rightarrow$ client (primary station): **80_H**
- Windows size for transmission client (primary station) $\rightarrow$ server (secondary station):   **01_H**
- Windows size for transmission server (secondary station) $\rightarrow$ client (primary station):   **07_H**

6.4.4.4.3.3   Disconnecting the MAC connection

The primary station shall disconnect the MAC link with secondary station(s) by sending a DISC command and shall start a response time-out function. The addressed secondary station(s), upon receiving the DISC command correctly, shall send a UA response at its first opportunity and shall enter the NDM. If, upon receipt of the DISC command, the addressed secondary station is already in the disconnected mode, it shall send a DM response. The primary station, upon receiving a UA or DM response to a DISC command, shall stop the response time-out function.

In the case of a multi-point configuration, the UA response from the secondary stations shall not interfere with one another. The mechanism to avoid overlapping responses to the DISC command using a group address or the all-station address is not within the scope of this standard.

If the DISC command, UA response or DM response is not received correctly, it shall be ignored by the receiving station. This will result in the expiry of the primary station's response time-out function, and the primary station may re-send the DISC command and restart the response time-out function.

This action may continue until a UA or DM response has been received correctly or until the DISC frame is transmitted MAX_NB_OF_RETRIES times. Any further recovery action takes place at a higher layer.

6.4.4.4.3.4   Exchanging data

The receipt of MA-DATA.request primitive will cause the MAC sub-layer to transmit the received Data to the peer MAC sub-layer using HDLC protocol procedures. Transferring Information frames is only allowed when MAC Connection is already established and the MAC sub-layer is in operational mode (NRM). In the case of Frame_type == I_COMPLETE (Information) this Data will be transmitted in one or several information frames (I frames). Sequence numbering and acknowledgement procedures of HDLC shall be applied. If necessary, the Data will be divided into sub-frames (segmentation) prior to transmission. The peer MAC layer has to recombine these sub-frames after reception. A special procedure is specified to allow transmitting long messages from the server to the client. This procedure is using the frame types I_FIRST_FRAGMENT, I_FRAGMENT, I_LAST_FRAGMENT, and is described in 6.4.4.5.

In case of Frame_type == UI the received Data will be transmitted in a UI frame. In this case, the size of the Data shall fit in one HDLC frame. UI frames basically serve for broadcasting/multicasting, but may also be used for event reporting. UI frames may be sent both in operational mode (NRM) and in non-operational mode (NDM).

Exchange of information frames

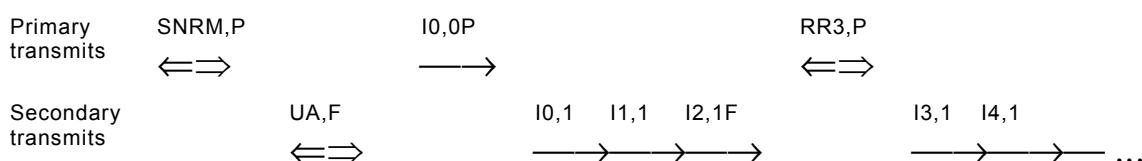*Information frame exchange is specified in detail in 6.11.4.2 of ISO/IEC 13239.*

In conclusion, as the NRM is used, the secondary station shall initiate transmission only as the result of receiving explicit permission to do so from the primary station; the primary station initiates each data exchange. The primary station shall set the poll bit to "1" in the last frame of a transmission to solicit response frames from the secondary station. The secondary station shall set the final bit to indicate the last frame of its response transmission (see 5.4.3 in ISO/IEC 13239).

I frames containing N(S) send and N(R) receive sequence numbers are used for confirmed information transfer. I frames shall be acknowledged by the receiver. Several I frames may be linked together, the poll/final bit indicates the last frame of such a sequence.

A time-out function shall be used in the primary station to monitor the responses of the secondary station.

The following examples show typical exchanges of frames in NRM with TWA transmission (see 6.11.4.5 of ISO/IEC 13239). More examples can be found in ISO/IEC 13239.

Example 1: Start-up procedure and information transfer without transmission errors (window size=3):

Primary transmits	SNRM,P ⇐⟹		I0,0P ⟶			RR3,P ⇐⟹		
Secondary transmits		UA,F ⇐⟹	I0,1 ⟶	I1,1 ⟶	I2,1F ⟶	I3,1 ⟶	I4,1 ⟶ ...

Example 2: Information transfer (window size 1) and closing procedure:

Primary transmits	I0,0P	I1,1P	I2,2P	RR3,P	DISC,P
	⟶	⟶	⟶	⟸⟹	⟸⟹

Secondary transmits	I0,1F	I1,2F	I2,3F	RR3,F	UA,F
	⟶	⟶	⟶	⟸⟹	⟸⟹

Example 3: Information transfer with intermediate UI frame:

Primary transmits	I0,0P	I1,1P	RR1,P	RR1,P	RR2,P
	⟶	⟶	⟸⟹	⟸⟹	⟸⟹

Secondary transmits	I0,1F	RR2,F	UI,F	I1,2F	RR2,F
	⟶	⟸⟹	⟶	⟶	⟸⟹

where

⟶ frame with info ⟸⟹ frame without info   P/F poll/final bit

6.4.4.4.3.5   Window size considerations

The HDLC standard allows transferring more than one frame in a sequence before an acknowledge is due. The send and receive sequence numbers allow acknowledgement with the information of how many frames have been correctly received. The maximum number of consecutive frames is referred to as the *window size*.

The maximum value of the window size depends on the range of the send/receive sequence numbers. In this protocol, 3-bit sequence numbering is used (range 0..7). Therefore, the maximum window size is 7.

The default value of the window size shall be 1 for both the primary and the secondary stations, unless another value has been negotiated at connection establishment time (see 6.4.4.4.3.1).

If window size > 1 is used, consecutive frames may be linked together. In this case, a single flag is used as both the closing flag for one frame and the opening flag for the next frame (see Figure 11).

If window size > 1 is used and frames are not linked together, a time-out between consecutive frames has to be observed (see 6.4.4.3.3).

6.4.4.4.3.6   Segmentation

In the frame format field, the segmentation subfield of 1 bit follows the format type subfield and if present, reduces the length subfield by one bit. The field is used as follows (see 4.9.3 of ISO/IEC 13239):

• all MSDUs transmitted shall have the segmentation algorithm applied. The algorithm shall be applied to MSDUs which fit in a single HDLC frame or those which must be transmitted as a sequence of HDLC frames;

• the final HDLC frame of an MSDU shall be sent with segmentation subfield = 0;

• when MSDUs must be transmitted in multiple HDLC frames, all except the final HDLC frame of the MSDU shall be sent with their segmentation sub-field = 1.

• The HDLC window sequence numbers guarantee that all segments are sent/received in order and that lost segments can be detected.

6.4.4.5 Transferring long MSDUs from the server to the client

As servers are often embedded systems with relatively poor memory resources, a special mechanism is specified to transmit 'long' MSDUs from the server to the client.

NOTE   From the implementation point of view, an MSDU is considered to be 'long' when the server does not have enough room to allocate a buffer, which may contain the complete MSDU.

In order to transmit transparently (from the point of view of the peer client MAC layer) these long Service User layer PDUs, the server Service User layer shall be able to segment the complete PDU and send the first segment of it by invoking the DL-DATA.request service with Frame_type == I_FIRST_FRAGMENT.

The server MAC sub-layer shall send the received MSDU within as many HDLC frames as necessary, but it shall send the last frame with the Segmentation bit set to 1. When this last frame is acknowledged by the client MAC sub-layer (with a RR frame), the server MAC sub-layer shall generate an MA-DATA.confirm primitive with Frame_type = I_FIRST_FRAGMENT.

When the server Service User layer receives the DL-DATA.confirm primitive, it shall send the next segment of the long PDU, by invoking the DL-DATA.request service with Frame_type == I_FRAGMENT. HDLC frames containing this segment shall be transmitted similarly to the frames of the first segment: even the last HDLC frame shall be sent with Segmentation bit = 1, and on the receipt of the acknowledgement of this last frame a DL-DATA.confirm primitive shall be generated with Frame_type == I_FRAGMENT.

The following segments of the long Service User layer PDU shall be transmitted in the same way – until the last one, which shall be transmitted using a DL_DATA.request service with Frame_type == I_LAST_FRAGMENT. The final HDLC frame of that last fragment shall be transmitted with the Segmentation bit = 0, meaning that the complete PDU has been sent – and received at the client side.

The client side shall generate an MA-DATA.indication primitive only when this final HDLC frame (with the Segmentation bit = 0) is received: the fragmentation procedure provided by the server is not at all visible at the client side. That is the reason that this procedure is considered to be transparent.

Figure 13 shows the corresponding message sequence chart for the GET.request application layer service. The READ.request service is treated in the same way.

IEC 260/02

Figure 13 – MSC for long MSDU transfer in a transparent manner

6.4.4.6 Multicasting and broadcasting

Multicasting and broadcasting are possible using UI frames. In the present environment, this is allowed only for the clients – servers are not allowed to send frames with broadcast or multicast address in the Destination_Address field.

UI frames shall be reported to the service user layer if the Destination Address of the frame designates one of the local MSAPs, if it is an existing local group address, or if it is the broadcast MAC address.

Only UI (and DISC)[1] messages may serve as broadcast or multicast messages; all other message types with any broadcast or multicast address in the Destination_Address field shall be discarded by the server MAC sub-layer. Broadcast and multicast UI messages shall always

---

[1] When the data link layer specified in this standard is used together with the COSEM application layer specified in IEC 62056-53, releasing of Application Association is done by disconnecting the supporting data link layer connection. To release non-pre-established multiple Application Associations, the DISC command may also be sent with a broadcast or multicast address. The mechanism used to avoid overlapping responses to the disconnect (DISC) command using a group address or the all-station address is not within the scope of this standard. SNRM commands using group or the all-station address must not be taken into account by the Server.

be sent with Poll bit == FALSE. Broadcast and multicast UI frames with Poll == TRUE shall be discarded.

Broadcast and multicast are allowed both to the logical devices within a physical device, using the upper HDLC address and to physical devices, using the lower HDLC address.
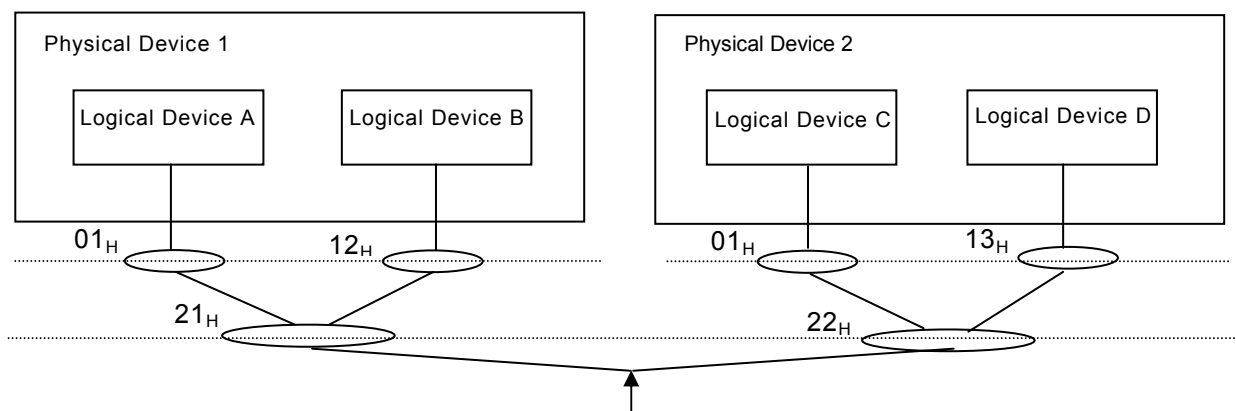
IEC 261/02

Figure 14 – Example configuration to illustrate broadcasting

Figure 14 shows an example with two Physical Devices, each of them including two Logical Devices. MAC Addresses for these devices are as follows:

Table 9 – Summary of MAC Addresses for the example

	Upper MAC Address (logical device)	Lower MAC Address (physical device)
Logical Device A	01_H	21_H
Logical Device B	12_H	21_H
Logical Device C	01_H	22_H
Logical Device D	13_H	22_H

Messages with a broadcast address at any MAC address level shall be handled as follows:

Table 10 – Broadcast UI frame handling

Lower MAC Address (Physical Device Address)	Upper MAC Address (Logical Device Address)	UI message handling
Individual (e.g. 21_H)	Individual (e.g. 01_H)	The incoming UI frame shall be sent to the individually addressed Logical Device within the individually addressed Physical Device. In the example, it is Logical Device A. The MAC sub-layer of Physical Device 2 shall discard the received frame
Individual (e.g. 22_H)	Broadcast ($7F_H$)	The incoming UI frame shall be sent to all Logical Devices within the individually addressed Physical Device. In the example, the message shall be sent to Logical Devices C and D. The MAC sub-layer of Physical Device 1 shall discard the received frame
Broadcast ($7F_H$)	Individual (e.g. 01_H)	The incoming UI frame shall be sent to all individually addressed Logical Devices in all Physical Devices. In the example, the message shall be sent to Logical Devices A and C
Broadcast ($7F_H$)	Broadcast ($7F_H$)	The incoming UI frame shall be sent to all Logical Devices in all Physical Devices. In the example, the message shall be sent to Logical Devices A, B, C, and D

6.4.4.7   Sending a UI frame from the server to the client

This subclause discusses a situation where a server MAC layer receives a DL-DATA.request service invocation with Frame_type = UI.

By its nature, this request may happen at any moment, in an asynchronous manner. (It is not the result of a previous DL-DATA.indication, thus it is non-solicited.) Therefore, when it happens, the server MAC layer generally has no right to send it out immediately ("no right to talk"). Consequently, this UI packet shall be stored in the MAC layer, while waiting for the opportunity to be sent out.

Pending UI frames can be sent on the following conditions:

When a DL-DATA.request invocation with frame_type I_COMPLETE, I_FIRST_FRAGMENT, I_FRAGMENT, or I_LAST_FRAGMENT is received. The pending UI frame(s) shall be sent out just before the last window, which shall contain the closing Final=TRUE bit of the transmission.
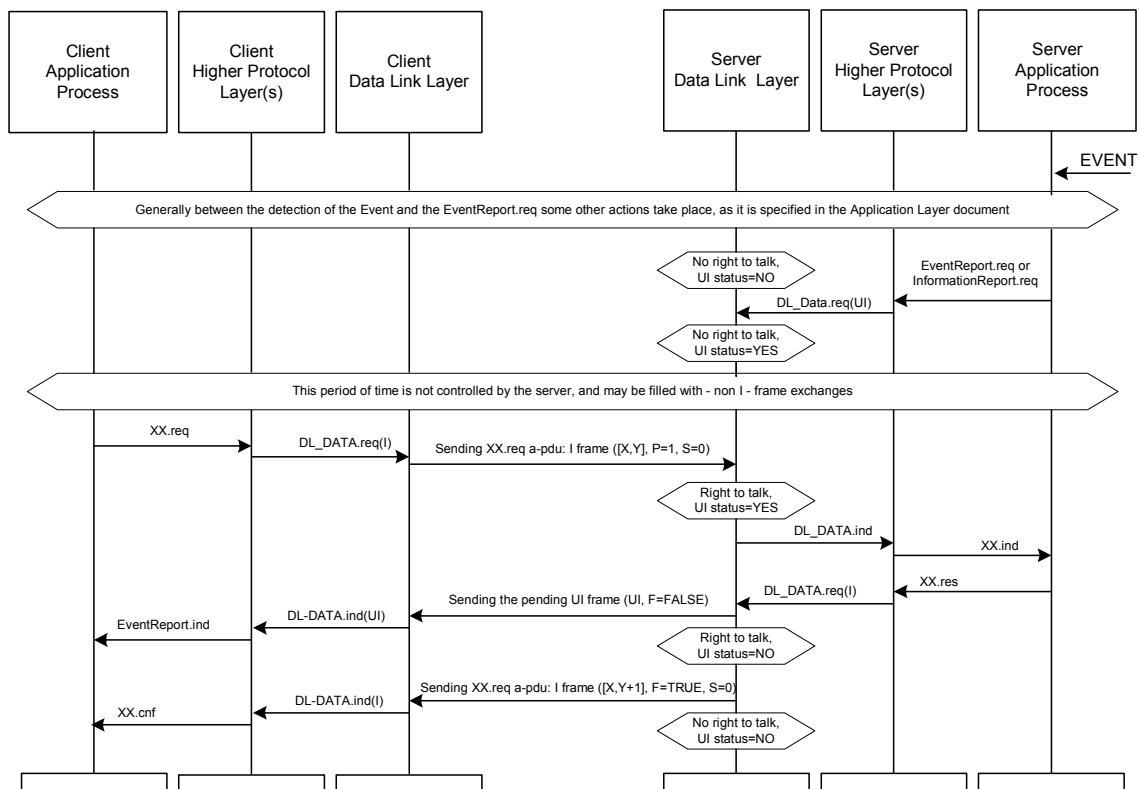
Figure 15 – Sending out a pending UI frame with a .response data

In receipt of a RR frame with P=1. If the server MAC layer has no pending I or UI frame when a RR frame is received, normally it shall respond with another RR frame, just to give back the "right to talk" to the HDLC primary station. When there is a pending UI frame, this UI frame shall be sent out before the normal response frame.
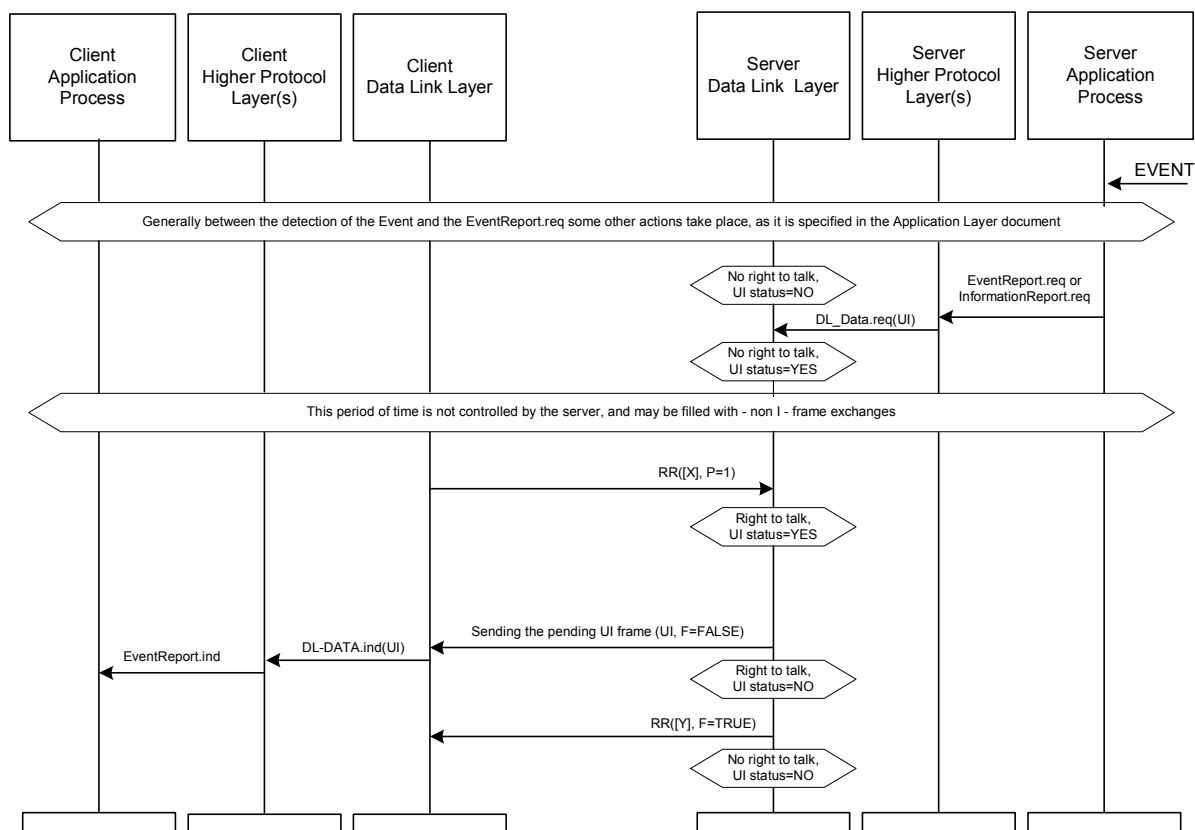
Figure 16 – Sending out a pending UI frame with a response to a RR frame

IEC   263/02

a) In receipt of a UI frame with P=1 and with zero length information field (empty UI frame). The receipt of this frame shall make the server data link layer send out all pending UI frames. The last UI frame shall be sent out with F=TRUE.
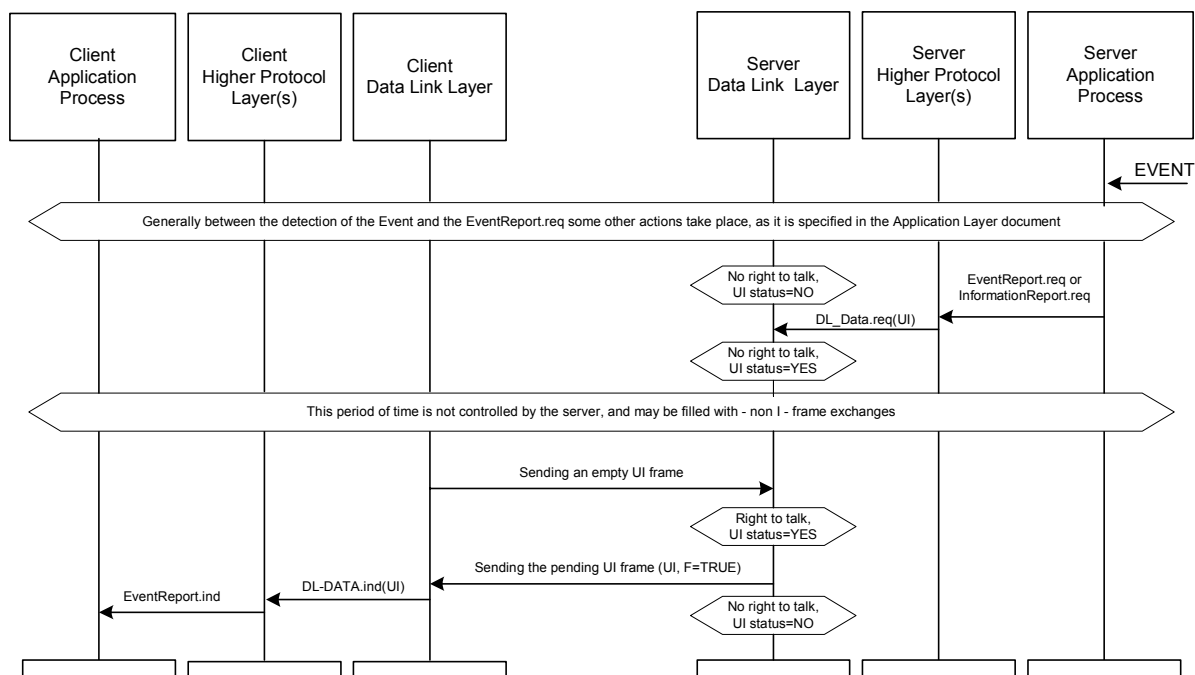
IEC   264/02

Figure 17 – Sending out a pending UI frame on receipt of an empty UI frame

6.4.4.8   Handling the CALLING device physical address

It is possible that the server initiates a physical connection establishment to the client device. Once this physical connection is established, the server can place a UI message in the MAC layer, which sends it out on one of the conditions discussed in 6.4.4.7.

As the client does not know whether the incoming call has come from a single meter or from a multi-drop configuration, it shall use CALLING Physical Device address in place of the MAC Physical Destination Address parameter. If the client wishes to send an SNRM frame to the server, the frame shall contain the following parameters:

Source Address:	Client Management Process HDLC Address (01_H)
Destination Address	
Lower MAC Address:	CALLING Physical Device Address ($7E_H$)
Upper MAC Address:	Management Logical Device Address (01_H)
Poll bit:	TRUE
Information Field:	Negotiable HDLC Layer Parameters (if any)

Only the initiator server device shall respond to this SNRM frame.

When the initiator is a single meter, it does not necessarily have its own proper physical device address. Nevertheless, it shall accept the incoming SNRM frame including the CALLING Physical Device Address (a special lower MAC Address, reserved for this purpose, see 6.4.2.4).

A problem might occur when the initiator of the call was a server of a multi-drop configuration: all servers of the multi-drop will receive the incoming SNRM frame, and all servers shall recognize that it is addressed to the CALLING Physical Device. In order to avoid potential conflicts, and to correctly handle the CALLING Physical Device Address, the lower MAC layer shall contain a CALLING DEVICE (Boolean) layer parameter, with default value = FALSE. Once the device's Application Process initiates a call, it shall set this layer parameter to TRUE.

NOTE   Single meters may also use the CALLING DEVICE (Boolean) layer parameter. In this case, it shall always be set to TRUE.

When the MAC layer receives a HDLC frame with the CALLING device address in the 'Physical MAC Address' portion of the Destination Address field, it shall check the value of the CALLING DEVICE layer variable, and if it is FALSE, it shall discard the incoming frame. On the other hand if CALLING DEVICE = TRUE, the MAC layer shall consider that it was addressed to that device, and shall pass the DL-CONNECT.indication to the upper layer. Using this layer variable ensures that only the originator server shall respond to the received SNRM frame.

When the MAC sub-layer of the initiator device recognizes that the incoming SNRM frame has been addressed to that device via the CALLING DEVICE address, it shall handle this SNRM exactly as if it was addressed to this device via its proper physical address: depending on the acceptability of the requested MAC connection it shall respond either with a UA or with a DM frame. The Lower MAC Address sub-field of the source address field of this UA or DM frame shall contain not the CALLING DEVICE address, but the proper Physical Address of the responding device.

When the current session terminates – PH-ABORT.indication is received – the MAC layer shall set the value of the CALLING DEVICE variable to FALSE.

6.4.4.9   Exception recovery

6.4.4.9.1   Response time-out

In the primary station, a time-out function is used to monitor the responses of the secondary station (see 6.11.4.3 of ISO/IEC 13239).

The response timer is started after transmission of a frame with the poll bit set to "1". It is restarted after receipt of an error-free frame with the final bit set to "0". It is stopped after receipt of an error-free frame with the final bit set to "1".

If a time-out occurs, the primary station shall retransmit the last frame it has sent. If the last frame sent was an information frame, this I frame shall not be retransmitted but a RR frame shall be sent in order to poll the secondary station and resynchronize I frame numbering.

The duration of the time-out function and the maximum number of retries are defined below.

6.4.4.9.2   FCS and HCS error

Frames that have no HCS (only an FCS, frames without information field) shall be treated as described in 4.2.6 of ISO/IEC 13239.

Frames received with an HCS error are treated as described in 5.6.3 of ISO/IEC 13239. Furthermore, all consecutive frames that are directly concatenated have to be discarded (no frame resynchronization possible because the length field of the erroneous frame has to be ignored).

Frames received with an FCS error, but with an error-free header (no HCS error), shall not be accepted by the receiver, except for examination of the length field, the state of the P/F bit and the value of the N(R) field.

6.4.4.9.3   N(S) sequence error

Sequence errors shall be handled as defined in 5.6.2 of ISO/IEC 13239. Poll/final bit recovery shall be used (5.6.2.1 of ISO/IEC 13239).

6.4.4.9.4   Command/response frame rejection

A command/response rejection exception condition shall be handled as specified in 5.6.4 of ISO/IEC 13239. The secondary station shall send a FRMR response with appropriate diagnostics information. On receipt of an FRMR response, the primary station shall at least issue a SNRM command before continuing with information exchange.

6.4.4.9.5   Busy

Busy conditions shall be treated as specified in 5.6.1 of ISO/IEC 13239.

6.4.4.10   Time-outs and other MAC sub-layer parameters

This subclause contains the list of time-outs and other MAC sub-layer parameters associated with services for the MAC sub-layer.

6.4.4.10.1   Time-out 1: Response time-out (TO_WAIT_RESP)

This time-out function is provided only by the MAC sub-layer of the primary station (client) – and is the same for all command (SNRM, DISC) and numbered information (I) frames. The maximum time waited by the primary station for the return frame from the secondary station before retrial must be chosen for example as:

TO_WAIT_RESP > RespTime + 2*MaxTxTime

where RespTime represents the theoretical response time of the secondary station and MaxTxTime the maximum time for transmission of a frame.

6.4.4.10.2   Layer Parameter 1: Maximum number of retries (MAX_NB_OF_RETRIES)

If no response frame is received during TO_WAIT_RESP time-out, the client MAC sub-layer will repeat the transmission of the command frame MAX_NB_OF_RETRIES times. This parameter is provided only by the client MAC sub-layer. If no response is received after the last time-out, the MAC sub-layer will generate a .confirm service indicating the reason (NOK_MAX_NB_RETRIES). In this case, the user layer shall request shutting down of the MAC Connection by invoking the MA-DISCONNECT.request service.

NOTE 1   This parameter does not apply on the unnumbered information (UI) frames.

NOTE 2   Shutting down the MAC connection is not done by the client MAC sub-layer itself.

6.4.4.10.3   Time-out 2: Inactivity time-out

This time-out is re-started each time that an octet is sent or received to/from the physical layer. If the Inactivity Time-out runs out, the data link layer shall generate a DL-LM_EVENT.indication primitive, signalling that no character has been sent/received during that period, and re-start the inactivity time-out. The data link layer shall be disconnected.

6.4.4.10.4   Time-out 3: Inter-frame time-out

The maximum permitted time between the stop bit of a character (octet) and the start bit of the next character within a frame (T_{in}) shall be selected to meet the requirements of the physical medium used. Whenever this time-out occurs in the receiver, the end of the actually received frame shall be assumed.

NOTE   The inter-octet time out, defined in 6.4.4.3.3, is the same as the inter-frame time-out.

6.4.4.10.5   Maximum information field length

The maximum information field length is 128 bytes by default and may be negotiated at connection time. The maximum value depends on the quality of the physical channel.

6.4.4.10.6   Window size

The window size is 1 by default and may be negotiated at connection time. The maximum value of the window size is 7.

6.4.5   State transition diagram for the server MAC sub-layer

Figure 18 shows a simplified state transition diagram for the server MAC sub-layer.

IEC 265/02
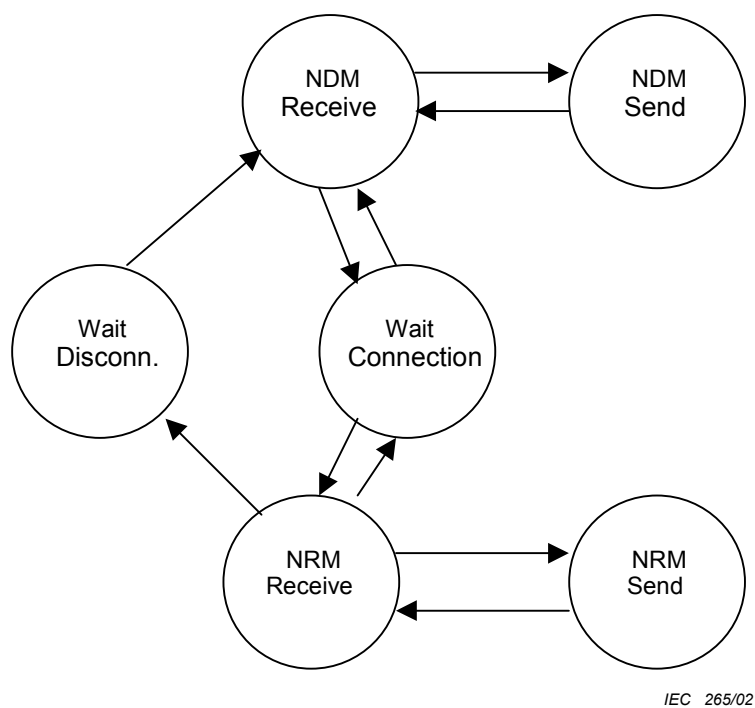
Figure 18 – State transition diagram for the server MAC sub-layer

Annex A
(informative)

FCS calculation

A.1   Test sequence for the FCS calculation [1]

The example presented here shows the proper FCS value for a two-byte frame consisting of 0x03 and 0x3F. The complete resulting frame is $7E_H$ 03_H $3F_H$ $5B_H$ EC_H $7E_H$.

```
\/ – first bit transmitted                              last bit transmitted – \/
0111 1110     1100 0000     1111 1100     1101 1010 0011 0111     0111 1110
   flag         address        control              FCS                  flag
```

In the test sequence, the following rules (according to ISO/IEC 13239) are considered:

* the FCS is calculated considering the bit order as transmitted on the channel;

* for the address field, the control field and all the other fields (including the data, except the FCS) the low order bit (of each byte) is transmitted first (this rule is automatically followed by the UART);

* for the FCS the coefficient of highest term (corresponding to x^{15}) is transmitted first.

A.2   Fast frame check sequence (FCS) implementation

The following example implementation of the 16-bit FCS calculation is derived from the internet Request for Comments 1662[2] that describes the PPP.

The FCS was originally designed with hardware implementations in mind. A serial bit stream is transmitted on the wire, the FCS is calculated over the serial data as it goes out and the complement of the resulting FCS is appended to the serial stream, followed by the Flag Sequence.

The receiver has no way of determining that it has finished calculating the received FCS until it detects the Flag Sequence. Therefore, the FCS was designed so that a particular pattern results when the FCS operation passes over the complemented FCS. A good frame is indicated by this "good FCS" value.

A.3   16-bit FCS computation method

The following code provides a table lookup computation for calculating the FCSequence as data arrives at the interface.

```
/*
 * u16 represents an unsigned 16-bit number. Adjust the typedef for
 * your hardware.
 *
 * Drew D. Perkins at Carnegie Mellon University.
 *
 * Code liberally borrowed from Mohsen Banan and D. Hugh Redelmeier.
 *
 */
```

---

[1] The test sequence presented here can be found in the 1988 CCITT Blue Book X.1 – X.32, Appendix I.

[2] RFC 1662, *PPP in HDLC-like Framing*, July 1994, W. Simpson.

```
    typedef unsigned short u16;
    /*
     *
     * FCS lookup table as calculated by the table generator.
     *
     */

static u16 fcstab[256] = {
      0x0000, 0x1189, 0x2312, 0x329b, 0x4624, 0x57ad, 0x6536, 0x74bf,
      0x8c48, 0x9dc1, 0xaf5a, 0xbed3, 0xca6c, 0xdbe5, 0xe97e, 0xf8f7,
      0x1081, 0x0108, 0x3393, 0x221a, 0x56a5, 0x472c, 0x75b7, 0x643e,
      0x9cc9, 0x8d40, 0xbfdb, 0xae52, 0xdaed, 0xcb64, 0xf9ff, 0xe876,
      0x2102, 0x308b, 0x0210, 0x1399, 0x6726, 0x76af, 0x4434, 0x55bd,
      0xad4a, 0xbcc3, 0x8e58, 0x9fd1, 0xeb6e, 0xfae7, 0xc87c, 0xd9f5,
      0x3183, 0x200a, 0x1291, 0x0318, 0x77a7, 0x662e, 0x54b5, 0x453c,
      0xbdcb, 0xac42, 0x9ed9, 0x8f50, 0xfbef, 0xea66, 0xd8fd, 0xc974,
      0x4204, 0x538d, 0x6116, 0x709f, 0x0420, 0x15a9, 0x2732, 0x36bb,
      0xce4c, 0xdfc5, 0xed5e, 0xfcd7, 0x8868, 0x99e1, 0xab7a, 0xbaf3,
      0x5285, 0x430c, 0x7197, 0x601e, 0x14a1, 0x0528, 0x37b3, 0x263a,
      0xdecd, 0xcf44, 0xfddf, 0xec56, 0x98e9, 0x8960, 0xbbfb, 0xaa72,
      0x6306, 0x728f, 0x4014, 0x519d, 0x2522, 0x34ab, 0x0630, 0x17b9,
      0xef4e, 0xfec7, 0xcc5c, 0xddd5, 0xa96a, 0xb8e3, 0x8a78, 0x9bf1,
      0x7387, 0x620e, 0x5095, 0x411c, 0x35a3, 0x242a, 0x16b1, 0x0738,
      0xffcf, 0xee46, 0xdcdd, 0xcd54, 0xb9eb, 0xa862, 0x9af9, 0x8b70,
      0x8408, 0x9581, 0xa71a, 0xb693, 0xc22c, 0xd3a5, 0xe13e, 0xf0b7,
      0x0840, 0x19c9, 0x2b52, 0x3adb, 0x4e64, 0x5fed, 0x6d76, 0x7cff,
      0x9489, 0x8500, 0xb79b, 0xa612, 0xd2ad, 0xc324, 0xf1bf, 0xe036,
      0x18c1, 0x0948, 0x3bd3, 0x2a5a, 0x5ee5, 0x4f6c, 0x7df7, 0x6c7e,
      0xa50a, 0xb483, 0x8618, 0x9791, 0xe32e, 0xf2a7, 0xc03c, 0xd1b5,
      0x2942, 0x38cb, 0x0a50, 0x1bd9, 0x6f66, 0x7eef, 0x4c74, 0x5dfd,
      0xb58b, 0xa402, 0x9699, 0x8710, 0xf3af, 0xe226, 0xd0bd, 0xc134,
      0x39c3, 0x284a, 0x1ad1, 0x0b58, 0x7fe7, 0x6e6e, 0x5cf5, 0x4d7c,
      0xc60c, 0xd785, 0xe51e, 0xf497, 0x8028, 0x91a1, 0xa33a, 0xb2b3,
      0x4a44, 0x5bcd, 0x6956, 0x78df, 0x0c60, 0x1de9, 0x2f72, 0x3efb,
      0xd68d, 0xc704, 0xf59f, 0xe416, 0x90a9, 0x8120, 0xb3bb, 0xa232,
      0x5ac5, 0x4b4c, 0x79d7, 0x685e, 0x1ce1, 0x0d68, 0x3ff3, 0x2e7a,
      0xe70e, 0xf687, 0xc41c, 0xd595, 0xa12a, 0xb0a3, 0x8238, 0x93b1,
      0x6b46, 0x7acf, 0x4854, 0x59dd, 0x2d62, 0x3ceb, 0x0e70, 0x1ff9,
      0xf78f, 0xe606, 0xd49d, 0xc514, 0xb1ab, 0xa022, 0x92b9, 0x8330,
      0x7bc7, 0x6a4e, 0x58d5, 0x495c, 0x3de3, 0x2c6a, 0x1ef1, 0x0f78
};

    #define PPPINITFCS16    0xffff  /* Initial FCS value */
    #define PPPGOODFCS16    0xf0b8  /* Good final FCS value */

    /*
     * Calculate a new fcs given the current fcs and the new data.
     */
    u16 pppfcs16(fcs, cp, len)
        register u16 fcs;
        register unsigned char *cp;
        register int len;
    {
        ASSERT(sizeof (u16) == 2);
        ASSERT(((u16) -1) > 0);
        while (len--)
            fcs = (fcs >> 8) ^ fcstab[(fcs ^ *cp++) & 0xff];

        return (fcs);
    }

    /*
     * How to use the fcs
     */
    tryfcs16(cp, len)
        register unsigned char *cp;
        register int len;
    {
        u16 trialfcs;

        /* add on output */
```

```
    trialfcs = pppfcs16( PPPINITFCS16, cp, len );
    trialfcs ^= 0xffff;                    /* complement */
    cp[len] = (trialfcs & 0x00ff);         /* least significant byte first */
    cp[len+1] = ((trialfcs >> 8) & 0x00ff);

    /* check on input */
    trialfcs = pppfcs16( PPPINITFCS16, cp, len + 2 );
    if ( trialfcs == PPPGOODFCS16 )
        printf("Good FCS\n");
}
```

A.4 FCS table generator

The following code creates the lookup table used to calculate the FCS-16.

```
/*
 * Generate a FCS-16 table.
 *
 * Drew D. Perkins at Carnegie Mellon University.
 *
 * Code liberally borrowed from Mohsen Banan and D. Hugh Redelmeier.
 */

/*
 * The FCS-16 generator polynomial: x**0 + x**5 + x**12 + x**16.
 */
#define P        0x8408

/*
 * NOTE The hex to "least significant bit" binary always causes
 * confusion, but it is used in all HDLC documents. Example: 03_H
 * translates to 1100 0000_B. The above defined polynomial value
 * (0x8408) is required by the algorithm to produce the results
 * corresponding to the given generator polynomial
 * (x**0 + x**5 + x**12 + x**16)
 */

main()
{
    register unsigned int b, v;
    register int i;

    printf("typedef unsigned short u16;\n");
    printf("static u16 fcstab[256] = {");
    for (b = 0; ; ) {
        if (b % 8 == 0)
            printf("\n");

        v = b;
        for (i = 8; i--; )
            v = v & 1 ? (v >> 1) ^ P : v >> 1;

        printf("\t0x%04x", v & 0xFFFF);
        if (++b == 256)
            break;
        printf(",");
    }
    printf("\n};\n");
}
```

<div align="center">

Annex B
(informative)

Data model and protocol

</div>

The data model uses generic building blocks to define the complex functionality of the metering equipment. It provides a view of this functionality of the meter as it is available at its interface(s). The model does not cover internal, implementation-specific issues.

The communication protocol defines how the data can be accessed and exchanged.

This is illustrated in the figure below:

<div align="center">
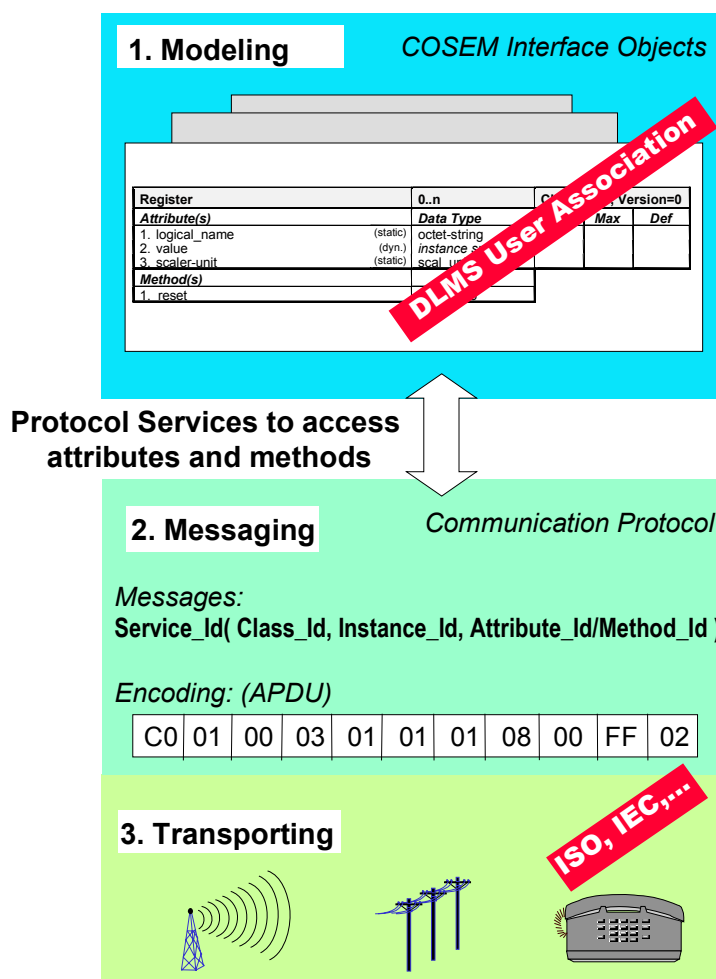
Figure B.1 – The three-step approach of COSEM

</div>

- The COSEM specification specifies metering domain-specific interface classes. The functionality of the meter is defined by the instances of these interface classes, called COSEM objects. This is defined in IEC 62056-62.

- Logical names, identifying the COSEM objects are defined in IEC 62056-61.

- The attributes and methods of these COSEM objects can be accessed and used via the messaging services of the application layer.

- The lower layers of the protocol transport the information.

NOTE In the three-layer connection oriented profile, the service user protocol layer is the COSEM application layer. Therefore, in this standard the meaning of the COSEM application layer is the same as the LLC service user protocol layer.

Annex C
(informative)

Data link layer management services

C.1   Data link layer management services

Figure C.1 shows management services provided by the data link layer to the system management process. The same service set is used both at the client and the server sides.

As these services are of local importance only, these clauses are included here only as guidance.
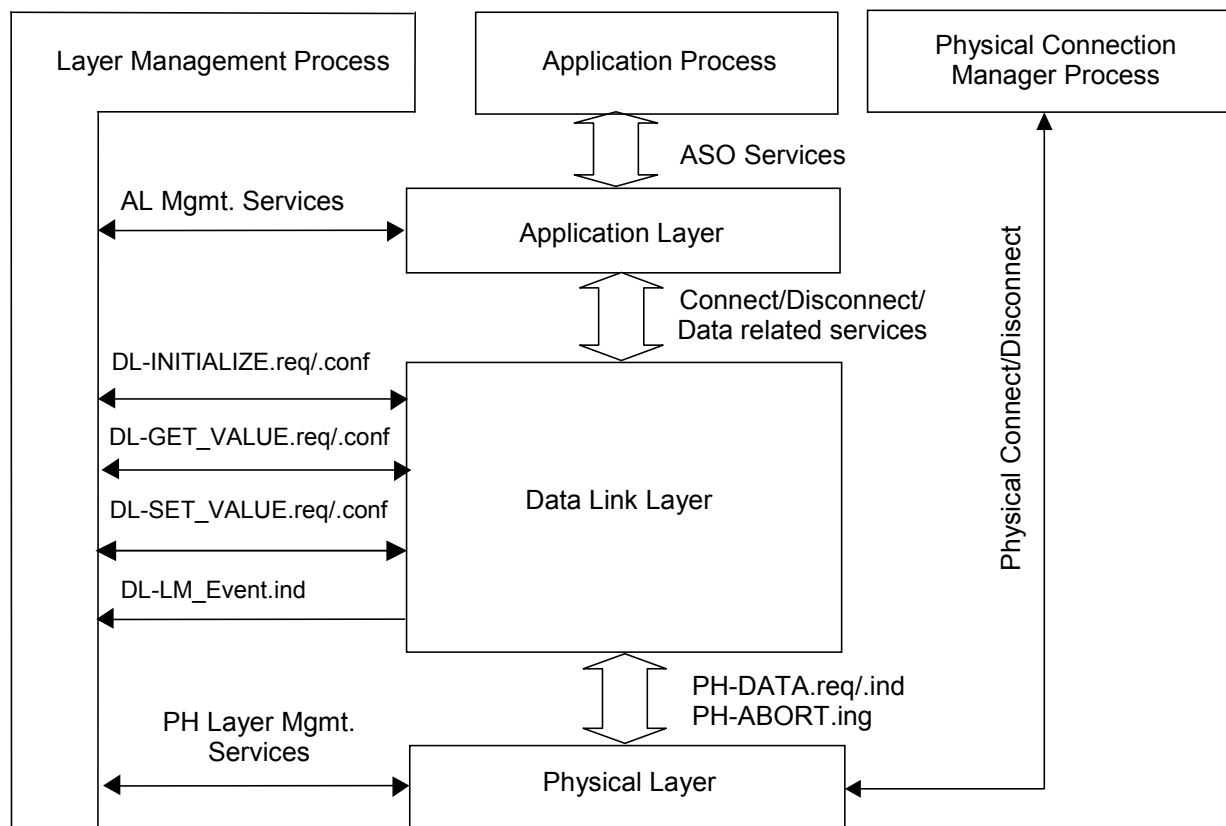
IEC   267/02

Figure C.1 – Layer management services

C.2   Data link layer management service definitions

C.2.1   DL-INITIALIZE.request

Function

This service primitive is used to initialize data link layer parameters (see 6.4.4.10) to their default values.

Service parameters

The semantics of this service primitive are as follows:

    DL-INITIALIZE.request

    (

)

Use

This primitive is used to initialize data link layer parameters to their default values.

C.2.2   DL-INITIALIZE.confirm

Function

This service primitive is used to locally confirm to the System Management process that the data link layer has executed the preceding DL-INITIALIZE.request service invocation.

Service parameters

The semantics of this service primitive are as follows:

    DL-INITIALIZE.confirm

    (

        Result

)

The value of the Result parameter indicates, whether the data link layer parameters have been successfully initialized or not.

Use

The data link layer generates this service primitive each time following the execution of a DL-INITIALIZE.request to indicate the result of the action.

C.2.3   DL-GET_VALUE.request

Function

This service primitive is used by the service user management process to obtain from the data link layer the value of one or more layer parameters.

Service parameters

The semantics of this primitive are as follows:

    DL-GET_VALUE.request

    (

        Layer_Parameter_Identifier_List

)

The Layer_Parameter_Identifier_List parameter indicates the required layer parameters.

Use

This service primitive is used to obtain from the data link layer the value of the layer parameters.

C.2.4 DL-GET_VALUE.confirm

Function

This service primitive is used to give back the value of layer parameters required by a preceding DL-GET_VALUE.request invocation.

Service parameters

The semantics of this primitive is as follows:

    DL-GET_VALUE.confirm

    (

        Layer_Parameter_GetResult_List

)

The Layer_Parameter_GetResult_List parameter carries the identifier(s) of the required parameter(s), the result of the GET operation applied to this parameter and in the case of a successful operation the value of the required layer parameters.

Use

The data link layer generates this service primitive each time following the execution of a DL-GET_VALUE.request to indicate the result of the action and to give back the layer parameters.

C.2.5 DL-SET_VALUE.request

Function

The service user management process invokes this service primitive to set the value of one or more layer parameters of the data link layer.

Service parameters

The semantics of this primitive is as follows:

    DL-SET_VALUE.request

    (

        Layer_Parameter_Value_List

)

The Layer_Parameter_Value_List parameter carries the identifier(s) and the value(s) of the required layer parameters to be set.

Use

This service primitive is used to set the value of one or more data link layer parameter.

C.2.6 DL-SET_VALUE.confirm

Function

This service primitive is used to indicate the result of a previously invoked DL-SET_VALUE.request.

Service parameters

The semantics of this primitive is as follows:

> DL-SET_VALUE.confirm
>
> (
>
> > Layer_Parameter_SetResult_List
>
>)

The Layer_Parameter_SetResult_List parameter carries the identifier(s) of the required parameter(s) and the result of the SET operation applied to this parameter.

Use

The data link layer generates this service primitive each time following the execution of a DL-SET_VALUE.request to indicate the result of the action.

C.2.7   DL-LM_EVENT.indication

Function

This service primitive is used to indicate the occurrence of a data link layer Event to the user of this service, the layer management process.

Service parameters

The semantics of this primitive is as follows:

> DL-LM_EVENT.indication
>
> (
>
> > Event_Identifier,
> > Event_Parameters
>
>)

The Event_Identifier parameter carries the identifier of the event(s) occurred, and the Event_Parameters parameter, if present, may give some additional information.

Use

The data link layer generates this service primitive each time when the occurrence of an event to be signalled is detected.

Bibliography

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 61334-6:2000, *Distribution automation using distribution line carrier systems – Part 6: A-XDR encoding rule*

Index

_____

ICS 91.140.50; 35.100.20