**A**

***SYNOPSIS REPORT***

*on*

**Device Language Message Specification**

*Submitted in partial fulfilment of the requirements for the degree of*

**BACHELOR OF TECHNOLOGY**



**Session: - Jan-June 2023**

Submitted by

Lokesh Bhoi (19ETCEC005)

8th Sem , ECE

Under Guidance of

Deepak Kumar Jain

Manager

Secure Meters Ltd.

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001**

**MAY - 2023**



Department of Electronics and Communication Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**Certificate**

This is to certify that project work titled **Device Language Message Specification**

by **Lokesh Bhoi** was successfully carried out in the Department of Electronics and Communication Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Electronics and Communication.

Deepak Kumar Jain Dr. Vivek Jain

Manager HOD, ECE Dept.

Secure Meters Ltd. Dept. of E.C.E TINJRIT, Udaipur

Date 25/05/2023 Date 25/052023



Department of Electronics and Communication Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**Examiner Certificate**

This is to certify that the following student

**Lokesh Bhoi**

of final year B.Tech. (Electronics & Communication Engineering), was examined for the project work titled

**Device Language Message Specification**

during the academic year 2022 – 2023 at Techno India NJR Institute of Technology, Udaipur

**Remarks:**

**Date:**

 Signature Signature

(**Internal Examiner**) (**External Examiner**)

Name :- ……………………… Name :- ………………………

Designation:- ……………….. Designation:- ………………..

Department: - ………………. Department: - ……………….

Organization:- ……………… Organization:- ………………

**Preface**

DLMS stands for Distribution Line Message Specification. It is an international standards established by IEC TC 57 and published as IEC61334-4-41.

The concept was driven forward later to become Device Language Message Specification with the objective to provide an interoperable environment for structured modeling and meter data exchange. Applications like remote meter reading, remote control and value added services for metering any kind of energy, like electricity, water, gas or heat are supported “Device Language Message specification” it is a generalized concept for structured modeling of meter data.

In Chapter 1, we give an overview of the Interoperability**,** DLMS and COSEM**,** Why we need of DLMS and Benefits of DLMS and about DLMS User Association

Chapter 2 discusses the Association objects**,** object identification system(OBIS) Code, Basic Structure of DLMS**,** Timeouts implementation

Chapter 3 About COSEM layer Architecture, Connection establishment takes place between client and server

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001

**ACKNOWLEDGMENT**

We take this opportunity to record our sincere thanks to all who helped us to successfully complete this work. Firstly, We are grateful to our **Manager Deepak Kumar Jain** for his invaluable guidance and constant encouragement, support and most importantly for giving us the opportunity to carry out this work.

We would like to express our deepest sense of gratitude and humble regards to our

**Head of Department Dr. vivek Jain** for giving invariable encouragement in our endeavors and providing necessary facility for the same. Also a sincere thanks to all faculty members of ECE, TINJRIT for their help in the project directly or indirectly.

Finally, We would like to thank my friends for their support and discussions that have proved very valuable for us. We are indebted to our parents for providing constant support, love and encouragement. We thank them for the sacrifices they made so that we could grow up in a learning environment. They have always stood by us in everything we have done, providing constant support, encouragement and love

**Lokesh Bhoi (19ETCEC005)**

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001

**CONTENTS**

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |
| **Abstract** ……………………………………………………………… | 8 |
| **List of Abbreviations Used**………………………………………………. | 9 |
| **Chapter 1**  | Overview Of DLMS | 10 |
|  | 1. Protocol Requirements
 |  |
|  | 1. Different types of protocol
 |  |
|  | 1. Why we need DLMS
 |  |
|  | 1. What is DLMS?
2. DLMS User Assosication
3. COSEM (Companion Specification for Energy Metering)
4. Function of layers
5. Data Access Security
 |  |
| **Chapter 2** | COSEM LAYER ARCHITECTURE………………………. | 14 |
|  | 1.COSEM Object Identification System (OBIS) |  |
|  | 2.COSEM Interface Classes |  |
|  | **3.**Authentication procedures |  |
|  |  |  |

**Chapter 3** COMMUNICARION BETWEEN CLIENT AND SERVER 25

 1 Communication in DLMS

 2 Communication Over Data Link and Application Layer

 3 HDLC frame format

 4 Frame Interpretation

 5 Reference for further

**ABSTRACT**

The DLMS (Device Language Message Specification) protocol has emerged as a key standard in the realm of smart energy management, facilitating efficient communication between utility companies and their consumers. This report provides a comprehensive overview of DLMS, highlighting its fundamental principles, architecture, and applications.

The report begins by outlining the significance of DLMS in the context of smart energy grids, emphasizing its role in enabling advanced metering infrastructure, demand response systems, and energy data management. It proceeds to delve into the core elements of the DLMS protocol, including its messaging structure, data model, and transport mechanisms. A detailed examination of the DLMS/COSEM (Companion Specification for Energy Metering) framework sheds light on the standardized object-oriented model for energy-related devices.

Moreover, the report addresses the security aspects of DLMS, discussing the authentication, confidentiality, and integrity mechanisms employed to protect sensitive energy data during transmission. It explores the integration of DLMS with various communication technologies such as PLC (Power Line Communication), RF (Radio Frequency), and IP (Internet Protocol), elucidating the flexibility and adaptability of the protocol across diverse network infrastructures.

Furthermore, the report presents a wide range of applications facilitated by DLMS, ranging from basic metering functionalities to advanced grid management operations. It highlights the benefits of DLMS in supporting remote meter reading, load profiling, tariff management, and real-time monitoring. Additionally, the report examines the interoperability of DLMS with other industry standards, such as IEC 61850, to foster seamless integration of energy management systems.

Overall, this report provides a comprehensive and insightful overview of the DLMS protocol, highlighting its significance, architecture, security features, applications, and future developments. It serves as a valuable resource for researchers, energy professionals, and stakeholders involved in the design, implementation, and management of smart energy systems.

Abbreviation

|  |  |
| --- | --- |
| **AA** | Application Association |
| **AARE** | Application Association Response |
| **AARQ** | Application Association ReQuest |
| **ACSE** | Application Control Service Element |
| **AL** | Application layer |
| **APDU** | Application Protocol Data Unit |
| **base\_name** | The short\_name corresponding to the first attribute (“logical\_name”) of a COSEM object. |
| **Class\_id** | Interface class identification code |
| **COSEM** | Companion Specification for Energy Metering |
| **COSEM****object** | An instance of an interface class |
| **DLMS** | Device Language Message Specification |
| **HLS** | High Level Security |
| **IC** | Interface Class |
| **IEC** | International Electrotechnical Commission |
| **IP** | Internet Protocol |
| **LLS** | Low Level Security |
| **LN** | Logical Name |
| **OBIS** | OBject Identification System |

**Chapter 1**

**Over View of DLMS/COSEM**

1. **Protocol Requirements**

Protocol Identifier

Data size and commands negotiation

Framing or segmentation.

Source and flexible destination address.

Synchronization byte

Transmission error check.

Full duplex system support-change of priority

Selectivity

Meter should be able to communicate its Feature set.

1. **Different types of protocol**
	1. **Proprietary Protocol**

Proprietary protocols are developed and maintained by manufacturer. It may or may not be shared with other.

**2.a.1 Pros**

* Improved performance and less development time.
* No Compliance or certifications

**2.a.2 Cons**

* Different interfaces with billing software
* Various kinds of reports for data analysis
* Utilities demands for specific reports and interfaces for their billing system.
* Proprietary protocols mean complete solution from one manufacturer including MRI, communication modems for reading and configuration tools.
	1. **Open Protocol**

Open protocols maintained by specified organization, manufacturer publish needed information with product.

 Example – Modbus Protocol

**2.b.1 Pros**

* Open standard, provide flexibility to incorporate with proprietary solution.
* Independent of manufacture

**2.b.2 Cons**

* Suitable for limited application only.
	1. **Interoperable Protocol**

Interoperability means that any meter data management system can communicate to any meter, independent of the manufacturer, the type, the energy type measured and the communication media. This also assumes, that on both sides a compatible set of features are correctly implemented.

Example – DLMS Protocol

**2.c.1 Types of interoperability**

* Reading Interoperability
* Interpretation Interoperability
1. **Why we need DLMS**
	1. Utilities need System Integration and Interoperability

b. DLMS/COSEM has been selected by both the electricity and the water/gas/heat metering community as the standard of choice, which is particularly favorable for Multi –Energy Providers gaining ground

c. DLMS/COSEM defines an interface model, valid for all kind of energy types, like electricity, gas, water, heat etc.

d. It supports innovation and future evolution by allowing manufacturer specific instances, attributes, methods, and the possibility to add new interface classes and versions without changing the services to access the objects, thus maintaining interoperability.

e. As the interface model is completely independent from the communication media,a wide choice of media can be used, without ever changing the model and the data management application of the data collecting system.

f. Unlike with older protocols, where a specific device driver in the data collection system was necessary for every new metering equipment, DLMS/COSEM facilitates to build generic drivers, able to communicate various meter types from different manufacturers.

1. **What is DLMS?**

DLMS stands for Distribution Line Message Specification.

It is an international standards established by IEC TC 57 and published as IEC

61334-4-41.

The concept was driven forward later to become Device Language Message

Specification with the objective to provide an interoperable environment for structured modeling and meter data exchange. Applications like remote meter reading, remote control and value added services for metering any kind of energy,like electricity, water, gas or heat are supported “Device Language Message specification” it is a generalized concept for structured modeling of meter data.

1. **DLMS User Assosication**

The DLMS Use Association is a non-profit organization, located in Geneva, Switzerland. Its mission is to develop, promote and maintain the DLMS/COSEM specification. It provides an information exchange forum for users, manufacturers and system providers, test houses and standardization bodies. It also provides a conformance testing and certification scheme for metering equipment implementing the specification. The DLMS UA is formally liaised with IEC TC 13 WG 14.

1. **COSEM (Companion Specification for Energy Metering)**

COSEM stands for Companion Specification for Energy Metering defines a framework for data exchange for the energy metering industry. Data exchange between data collection systems and metering equipment using the COSEM interface object model is based on the client/server paradigm

1. **Function of layers**
	1. COSEM Application Layer
* Prepares data and manages it.
* Controls access to data i.e. Password clearance
* Type of data i.e. Load survey, event, daily load profile, etc.
	1. Data Link Layer
* Manages fragmentation of data.
* Data synchronization and Error correction.
* It doesn’t know about data.
	1. Physical Layer
* This layer transmits data in signal form using a H/W protocol e.g. RS232, RS485 and GSM.
1. **Data Access Security**
	1. DLMS Provide Three authentication security levels that are specified
* No authentication (lowest level) security.
* Low level, password based authentication security (LLS) identifying only the client.
* High level, four-pass authentication security (HLS) identifying both the client and the server.

**Chapter 2**

**COSEM LAYER ARCHITECTURE**

1. **COSEM Object Identification System (OBIS)**

**1.a) Introduction**

The competitive electricity market requires an ever-increasing amount of timely information concerning the usage of electrical energy. Recent technology developments enable to build intelligent static metering equipment, which are capable of capturing, processing and communicating this information to all parties involved. For further analysis of this information, for the purposes of billing, load-, customer- and contract management, it is necessary to uniquely identify all data in a manufacturer independent way collected manually or automatically, via local or remote data exchange.

**1.b) OBIS structure**

 **1.b.1) General**

OBIS codes identify data items used in energy metering equipment, in a hierarchical structure using six value groups A to F

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** | **E** | **F** |

Table 1: OBIS General

**1.b.2) Value group A**

The value group A defines the media (energy type) to which the metering is related. Non-media related information is handled as abstract data

The range for value group A is 0 to 15

|  |  |
| --- | --- |
| **0** | Abstract objects |
| **1** | Electricity related objects |
|  |  |
|  |  |
| **4** | Heat cost allocator related objects |
| **5** | Cooling related objects |
| **6** | Heat related objects |
| **7** | Gas related objects |
| **8** | Cold water related objects |
| **9** | Hot water related objects |
| **All Others** | Reserved |

**Table 2 : OBIS value A**

**1.b.3) Value Group B**

The value group B defines the channel number, i.e. the number of the input of a metering equipment having several inputs for the measurement of energy of the same or different types (for example in data concentrators, registration units). Data from different sources can thus be identified. The definitions for this value group are independent from the value group A.

|  |  |
| --- | --- |
| **0** | No channel specified |
| **1** | Channel 1 |
| **….** |  |
| **64** | Channel 64 |
| **65 … 127** | Utility specific codes |
| **128 …. 199** | Manufacturer specific codes |
| **200 … 255** | Reserved |

Table 3 – Value group B codes

If channel information is not essential, the value 0 shall be assigned.

The range 65…127 is available for utility specific use. If the value of group B is in this range, the whole OBIS code shall be considered as utility specific and the value of other groups does not necessarily carry a meaning defined neither in Clause 4 nor Clause 5.

**1.b.4) Value group C**

The value group C defines the abstract or physical data items related to the information source concerned, for example current, voltage, power, volume, temperature. The definitions depend on the value of the value group A . Further processing, classification and storage methods are defined by value groups D, E and F. For abstract data, value groups D to F provide further classification of data identified by value groups A to C.

|  |  |
| --- | --- |
| **0 …. 89** | Context specific identifiers |
| **93** | Consortia specific identifiers |
| **94** | Country specific identifiers |
| **96** | General service entries |
| **97** | General error messages |
| **98** | General list objects |
| **99** | Abstract data profiles |
| **127** | Inactive objects |
| **128…199,240** | Manufacturer specific codes |
| **All Others** | Reserved |
| **0** | Abstract objects |

**1.b.5) Value group D**

The value group D defines types, or the result of the processing of physical quantities identified with the value groups A and C, according to various specific algorithms. The algorithms can deliver energy and demand quantities as well as other physical quantities.

specifies the use of value group D for consortia specific applications. In this table, there are no reserved ranges for manufacturer specific codes. The usage of value group E and F are defined in consortia specific documents.

|  |  |
| --- | --- |
| **01** | SELMA Consortium |
| **All Other** | Reserved |

**Table 5 – Value group D codes - Consortia specific identifiers**

|  |
| --- |
| **Value group D****Country specific identifiers a (A = any, C = 94)** |
| **00** | Finnish identifiers |
| **01** | USA identifiers |
| **02** | Canadian identifiers |
| **07** | Russian identifiers |
| **10** | Czech identifiers |
| **11** | Bulgarian identifiers |
| **12** | Croatian identifiers |
| **13** | Irish identifiers |
| **14** | Israeli identifiers |
| **15** | Ukraine identifiers |
| **16** | Yugoslavian identifiers |
| **27** | South African identifiers |
| **30** | Greek identifiers |
| **31** | Dutch identifiers |
| **32** | Belgian identifiers |
| **33** | French identifiers |
| **34** | Spanish identifiers |
| **35** | Portuguese identifiers |
| **Value group D****Country specific identifiers a (A = any, C = 94)** |
| **36** | Hungarian identifiers |
| **38** | Slovenian identifiers |
| **39** | Italian identifiers |
| **40** | Romanian identifiers |
| **41** | Swiss identifiers |
| **42** | Slovakian identifiers |
| **43** | Austrian identifiers |
| **44** | United Kingdom identifiers |
| **45** | Danish identifiers |
| **46** | Swedish identifiers |
| **47** | Norwegian identifiers |
| **48** | Polish identifiers |
| **49** | German identifiers |
| **55** | Brazilian identifiers |
| **61** | Australian identifiers |
| **62** | Indonesian identifiers |
| **64** | New Zealand identifiers |
| **65** | Singapore identifiers |
| **81** | Japanese identifiers |
| **86** | Chinese identifiers |
| **90** | Turkish identifiers |
| **91** | Indian identifiers |
| **All other** | Reserved |
| NOTE Objects that are already identified in this Recnical Report must not be re-identified by country specific identifiers. |

*Table 6 – Value group D codes – Country specific identifiers*

**1.b.6) Value group E**

The value group E defines further processing or classification of quantities identified by value groups A to D..

|  |
| --- |
| **Value group E** **Electricity related objects (A = 1)**  |
| **0**  | Total  |
| **1**  | Rate 1  |
| **2**  | Rate 2  |
| **3**  | Rate 3  |
| **…**  | ...  |
| **9**  | Rate 9  |
| **…**  | …  |
| **63**  | Rate 63  |
|  |   |
| **128…254**  | Manufacturer specific codes  |
| **All other**  | Reserved  |

*Table 7 – Value group E codes – Tariff rates*

**1.b.7) Value group F**

The value group F defines the storage of data, identified by value groups A to E, according to different billing periods. Where this is not relevant, this value group can be used for further classification.

Value group F specifies the allocation to different billing periods (sets of historical values) for the objects defined by value groups A to E, where storage of historical values is relevant.

A blling period scheme is identified with its billing period counter, number of available billing periods, time stamp of the billing period and billing period length.

The range for value group F is 0 to 255. In all cases, if value group F is not used, it is set to 255.

**1.b.8)** **Manufacturer specific codes**

In value groups B, C, D, E and F the following ranges are available for manufacturer-specific purposes:

* group B: 128…199;

group C: 128…199, 240;

* group D: 128…254;
* group E: 128…254;
* group F: 128…254.

If any of these value groups contain a value in the manufacturer specific range, then the whole OBIS code shall be considered as manufacturer specific.

1. **COSEM Interface Classes**

**2.a) Basic Principle**

This subclause describes the basic principles on which the COSEM interface classes are built. It also gives a short overview on how interface objects (instantiations of the interface classes) are used for communication purposes. Data collection systems and metering equipment from different vendors, following these specifications, can exchange data in an interoperable way.

Object modelling: for specification purposes this standard uses the technique of object modelling. An object is a collection of attributes and methods.

The information of an object is organized in attributes. They represent the characteristics of an object by means of attribute values. The value of an attribute may affect the behaviour of an object. The first attribute in any object is the “logical\_name”. It is one part of the identification of the object. An object may offer a number of methods to either examine or modify the values of the attributes.

Objects that share common characteristics are generalized as an interface class with a class\_id. Within a specific class, the common characteristics (attributes and methods) are described once for all objects. Instantiations of an interface class are called COSEM objects.

Manufacturers may add proprietary methods or attributes to any object, using negative numbers.

#### **2.b) Class description notation**

This subclause describes the notation used to define the interface classes. A short text describes the functionality and application of the class. A table gives an overview of

the class including the class name, the attributes, and the methods (class description template).

**Class name**

**Cardinality**

**class\_id, version**

***Attribute***

*(*

***Max.***

***Def.***

***s)***

***Data type***

***Min.***

(

logical\_name

.

1

static)

octet-string

2

.

…

…)

(

*…*

3

.

…

(...)

*…*

***Specific method***

(

***s) (if required)***

***m/o***

1

.

…

*…*

2

.

…

*…*

Each attribute and method must be described in detail.

|  |  |
| --- | --- |
| **Class name**  | Describes the class (for example “Register”, “Clock”, “Profile generic”,...)  |
| **Cardinality**  | Specifies the number of instances of the class within a logical device (see 4.1.6).   |
|   | *value*   | The class shall be instantiated exactly “value” times.   |
|   | *min...max.*   | The class shall be instantiated at least “min.” times and at most “max.” times. If min. is zero (0) then the class is optional, otherwise (min. > 0) "min." instantiations of the class are mandatory.  |
| **class\_id**  | Identification code of the class (range 0 to 65 535). The class\_id of each object is retrieved together with the logical name by reading the object\_list attribute of an “Association LN” / ”Association SN” object.  The class\_id-s from 0 to 8 191 are reserved to be specified by the DLMS UA. Class\_id-s from 8 192 to 32 767 are reserved for manufacturer specific interface classes. Class\_id-s from 32 768 to 65 535 are reserved for user group specific interface classes. DLMS UA reserves the right to assign ranges to individual manufacturers or user groups.  |
| **Version**  | Identification code of the version of the class. The version of each object is retrieved together with the logical name and the class\_id by reading the object\_list attribute of an “Association LN” / ”Association SN” object.  **Within one logical device, all instances of a certain class must be of the same version**   |

**Attribute(s)** Specifies the attribute(s) that belong to the class.

 (*dyn*.) Classifies an attribute that carries a process value, which is

updated by the meter itself.

 *(static)* Classifies an attribute, which is not updated by the meter

itself (for example configuration data).

|  |  |
| --- | --- |
| **logical\_name**  | octet-string The logical name is always the first attribute of a class. It identifies the instantiation (COSEM object) of this class. The value of the logical\_name conforms to OBIS (see Clause 5).  |
| **Data type**  | Defines the data type of an attribute (see 4.1.3).  |
| **Min.**    | Specifies if the attribute has a minimum value.  *x* The attribute has a minimum value.  <*empty*> The attribute has no minimum value.   |
| **Max.**    | Defines if the attribute has a maximum value.  *x* The attribute has a maximum value.  <*empty*> The attribute has no maximum value.   |
| **Def.**    | Specifies if the attribute has a default value. This is the value of the attribute after reset.  *x* The attribute has a default value.  <*empty*> The default value is not defined by the class definition.   |
| **Specific** **method(s)**   | Provides a list of the specific methods that belong to the object.  *Method Name ()* The method has to be described in the subsection "Method description".   |
| **m/o**    | Defines if the method is mandatory or optional.  *m (mandatory)* The method is mandatory.  *o (optional)* The method is optional.  |

**Attribute description**

Describes each attribute with its data type (if the data type is not simple), its data format and its properties (minimum, maximum and default values).

**Method description**

Describes each method and the invoked behaviour of the instantiated COSEM object(s).

**Selective access**

The xDLMS services Read, Write, UnconfirmedWrite (used with SN referencing) and GET, SET (used with LN referencing) typically reference the entire attribute. However, for certain attributes selective access to just a part of the attribute may be provided. The part of the attribute is identified by specific selective access parameters. These are defined as part of the attribute specification.

#### **2.b) COSEM logical device**

The COSEM logical device is a set of COSEM objects. Each physical device shall contain a “Management logical device”.

The addressing of COSEM logical devices shall be provided by the addressing scheme of the lower layers of the protocol used.

###### **2.b.1) COSEM logical device name**

The COSEM logical device can be identified by its unique COSEM logical device name. This name can be retrieved from an instance of IC “SAP assignment” (see 4.2.14), or of a COSEM object “COSEM logical device name” (see 4.6.1.1.25).

This name is defined as an octet-string of up to 16 octets. The first three octets uniquely identify the manufacturer of the device [[1]](#footnote-1). The manufacturer is responsible for guaranteeing the uniqueness of the octets that follow (up to 13 octets).

###### **2.b.2**) **The “association view” of the logical device**

In order to access COSEM objects in the server, an application association shall first be established. This characterizes the context within which the associated applications will communicate. The major parts of this context are:

This information is contained in a special COSEM object, the “Association” object. There are two types of this association object defined. One for associations using short name referencing (“Association SN”) and one for using logical name referencing (“Association LN”).

Depending on the association established between the client and the server, different access rights may be granted by the server. Access rights concern a set of COSEM objects – the visible objects – that can be accessed (‘seen’) within the given association. In addition, access to attributes and methods of these COSEM objects may also be restricted within the association (for example a certain type of client can only read a particular attribute of a COSEM object).

The list of the visible COSEM objects – the “association view” – can be obtained by the client by reading the “*object\_list*” attribute of the appropriate association object. Additional information about the access rights (read only, write only, read and write) to the attributes and the availability of the methods (within the established association) can be found via specific attributes (logical name referencing, see 4.2.12) or special methods (short name referencing, see 4.2.13) provided by the association objects.

###### **2.b.3) Mandatory contents of a COSEM logical device**

The following objects shall be part of each COSEM logical device. They shall be accessible for GET/READ in all application associations with this logical device:

###### **2.b.4) Management logical device**

The management logical device is a mandatory element of any physical device, and it has a reserved address. As defined in Clause 9.2.3.4 of the Green Book, it must support an application association to a public client with the lowest security level. Its role is to support revealing the internal structure of the physical device and to support notification of events in the server.

In addition to the “Association” object modelling the association with the public client, the management logical device shall contain a “SAP assignment” object, giving its SAP and the SAP of all other logical devices within the physical device. The SAP assignment object must be readable at least by the public client.

#### **Authentication procedures**

###### **3.a) Low Level Security (LLS) authentication**

As described in Clause 9 of the Green Book, the ACSE provides the authentication services for low level security (LLS). Low level security authentication is typically used when the communication channel offers adequate security to avoid eavesdropping and message (password) replay.

For LLS, all the authentication services are provided by the ACSE. The association objects provide only the method/attribute to change the “secret” (for example password).

For LLS authentication the client transmits a “secret” (for example a password) to the server, by using the “Calling\_Authentication\_Value” parameter of the COSEM-OPEN.request service primitive of the client application layer. The server checks if the received “secret” corresponds to the client identification. If yes, the client is authenticated and the association can be established.

###### **3.b) High Level Security (HLS) authentication**

As described in Clause 9 of the Green Book, the ACSE provides part of the authentication services for high-level security (HLS). High-level security authentication is typically used when the communication channel offers no intrinsic security and precautions have to be taken against eavesdroppers and against message (password) replay. In this case, a 4-pass authentication protocol is foreseen. The 4-pass authentication allows the authentication of the client as well as of the server in the following way.

**Pass1:** The client transmits “challenge” CtoS (for example a random number) to the server.

**Pass2:** The server transmits “challenge” StoC (for example a random number) to the client.

 The length of the challenges shall be 8 to 64 octets.

**Pass3:** The client processes StoC according to the rules of the HLS authentication mechanism negotiated. In case of HLS authentication\_mechanism\_id(2), the method of processing the challenge is secret (for example encrypting with a secret key), which is the HLS secret known by both the client and the server. In case of HLS authentication\_mechanism\_id(3), the client appends the HLS secret to the challenge StoC received during Pass2 and generates the digest using the MD5 algorithm (RFC 1321). In case of authentication\_mechanism\_id(4), the process is the same, but the SHA\_1 algorithm is used (FIPS PUB 180-1). The result – f(StoC) – is sent back to the server. The server checks if f(StoC) is the result of correct processing and – if correct – accepts the authentication of the client.

**Pass4:** If the client is authenticated, the server processes CtoS in the same way as described in Pass3. The result – f(CtoS) – is sent back to the client. The client checks if f(CtoS) is the result of the correct processing and – if correct – accepts the authentication of the server.

The HLS authentication service, supporting Pass1 is provided by the COSEM-OPEN.request service primitive of the client application layer. The parameter "Security\_Mechanism\_Name" carries the identifier of the HLS mechanism, and the parameter "Calling\_Authentication\_Value" carries the challenge CtoS.

The HLS authentication service, supporting Pass2 is provided by the COSEM-OPEN.response service primitive of the server application layer. The parameter "Security\_Mechanism\_Name" carries the identifier of the HLS mechanism, and the parameter "Responding\_ Authentication\_Value" carries the challenge StoC.

After Pass2, the association is formally established, but the access of the client is restricted to the method "reply\_to\_HLS\_authentication" of the current "association" object.

Pass3 and Pass4 are supported by the method **reply\_to\_HLS\_authentication** of the association object(s), (see 4.2.12, 4.2.13). If both passes are successfully executed, then full access is granted according to the current association. Otherwise, either the client or the server aborts the association.

**Chapter 3**

**COMMUNICARION BETWEEN CLIENT AND SERVER**

1. **Communication in DLMS**



1. **Communication Over Data Link and Application Layer**

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

Table 8 – Command and response frames

|  |  |
| --- | --- |
| **Commands**  | **Responses**  |
| I  | I  |
| RR  | RR  |
| RNR  | RNR  |
| SNRM  | UA  |
| DISC  | DM  |
| UI  | UI  |
|   | FRMR  |

**2.a) 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 – 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  |

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

**2.a.1) 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:

1. N(S), which shall indicate the sequence number associated with the I frame; and
2. 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 8.4.4.4.4.3.12.

**2.a.2) Receive ready (RR) command and response**

The RR frame shall be used by a data station to:

1. indicate that it is ready to receive an I frame(s); and
2. 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.

**2.a.3)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.

**2.a.4) 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 8.4.4.4.4.3.1) and to carry user information transported transparently across the link layer to the user of the data link.

**2.a.5) 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.

**2.a.6) 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 8.4.4.4.4.3.1).

**2.a.7) 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).

**2.a.8) 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 ISO/IEC 13239).

**2.a.9) 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).

1. **HDLC Frame format**

HDLC frame format type 3 as defined in Annex H.4 of ISO/IEC 13239.

HDLC frame format type 3 is as follows:

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

Figure 53 – 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.

**3.a) Flag field**

The length of the flag field is one byte and its value is 7EH. 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 54.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Flag  | Frame I  | Flag  | Frame I+1  | Flag  | Frame I+2  | Flag  |

Figure 54 – Multiple frames

**3.b) 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 55:

MSB LSB

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1  | 0  | 1  | 0  | S  | L  | L  | L  | L  | L  | L  | L  | L  | L  | L  | L  |
|  |  |  |  |  |  |  |  |

Format type Frame length sub-field

Figure 55 – The frame format field

The value of the format type sub-field shall be 1010 (binary).

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

**3.c) 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 of the HDLC standard ISO/IEC 13239 shall be applied to both address fields.

**3.d) 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 .

**3.d) Header check sequence (HCS)**

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).

**3.d.1)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.

**3.e) Frame check 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 8.5.

1. **Frame Interpretation**

**SNRM (*Set normal response mode*)**

|  |  |
| --- | --- |
| **Flag** | **7E** |
| **Frame Format**  | **A0** |
| **Frame Length** | **20** |
| **Destination Add** | **03** |
| **Source Addr** | **21** |
| **Control Field For SNRM(Fixed)** | **93** |
| **HCS** | **7D D9** |
| **Format Identifier** | **81** |
| **Group Identifier** | **80** |
| **Group Length (20 Octets)** | **14** |
| **parameter identifier (maximum information field length – Tx)** | **05**  |
| **parameter length (2 octet);** | **02** |
| **02 00 is equivalent to 512 bytes** | **02 00** |
| **parameter identifier (maximum info field length – Rx)** | **06** |
| **parameter length (2 octet);** | **02** |
| **02 00 is equivalent to 512 bytes** | **02 00** |
| **parameter identifier (window size, transmit)** | **07** |
| **parameter length (4 octets)** | **04** |
| **Window Size 01 (Default)** | **00 00 00 01** |
| **parameter identifier (window size, receive)** | **08** |
| **parameter length (4 octets)** | **04** |
|  | **00 00 00 01** |
| **FCS** | **6F EF** |
| **Flag** | **7E** |

**UA (*Unnumbered acknowledge*)**

|  |  |
| --- | --- |
| **Flag** | **7E** |
| **Frame Format**  | **A0** |
| **Frame Length** | **1F** |
| **Destination Add** | **21** |
| **Source Addr** | **03** |
| **Control Field For SNRM(Fixed)** | **73** |
| **HCS** | **78 66** |
| **Format Identifier** | **81** |
| **Group Identifier** | **80** |
| **Group Length (20 Octets)** | **13** |
| **parameter identifier (maximum information field length – Tx)** | **05** |
| **parameter length (2 octet);** | **02** |
| **02 00 is equivalent to 512 bytes** | **02 00** |
| **parameter identifier (maximum info field length – Rx)** | **06** |
| **parameter length (2 octet);** | **01** |
| **02 00 is equivalent to 512 bytes** | **80** |
| **parameter identifier (window size, transmit)** | **07** |
| **parameter length (4 octets)** | **04** |
| **Window Size 01 (Default)** | **01** |
| **parameter identifier (window size, receive)** | **08** |
| **parameter length (4 octets)** | **04** |
| **Window Size 01 (Default)** | **01** |
| **FCS** | **EC CE** |
| **Flag** | **7E** |

**Reading Meter Serial Number Query:**

**7E A0 19 03 21 32 6F D8 E6 E6 00 C0 01 81 00 01 01 00 00 00 00 FF 02 00 C6 60 7E**

|  |  |
| --- | --- |
| **7E** | **Start flag** |
| **A0 19 ( A = HDLC frame format 3)** | **Frame format and length** |
| **03 (0000 0011)** | **Destination address (0x01)** |
| **21(0010 0001)** | **Source address(0x10)** |
| **32 (Sender & Receiver frame numbers), Info frame** | **Control field “I” frame** |
| **6F D8** | **HCS** |
| **E6 E6 00** | **LLC header (Logical Link control)** |
| **C0 01 81 00 01 01 00 00 00 00 FF 02 00** | **Information Field** |
| **C6 60** | **FCS** |

|  |  |
| --- | --- |
| **7E** | **End Flag** |

 **Reading Meter Serial NumberResponse:**

**7E A0 1A 21 03 52 A4 38 E6 E7 00 C4 01 81 00 09 08 55 53 4F 38 34 34 38 37 90 21 7E**

|  |  |
| --- | --- |
| **7E** | **Start flag** |
| **A0 1A** | **Frame format and length** |
| **21** | **Destination address** |
| **03** | **Source address** |
| **52** | **Control field “I” frame** |
| **A4 38** | **HCS** |
| **E6 E7 00** | **LLC header** |
| **C4 01 81 00 09 08 55 53 4F 38 34 34 38 37** | **Information Field** |
| **90 21** | **FCS** |
| **7E** | **End Flag** |

**Information Field:**

 **C4 01 81 00 09 08 55 53 4F 38 34 34 38 37**

|  |  |
| --- | --- |
| **C4 01** | **Response for get data** |
| **81** | **Invoke-Id & Priority** |
| **00** | **Tag for data, means after it we are getting the data requested** |
| **09** | **Data Type = Octet string** |
| **08** | **Length of data** |
| **55 53 4F 38 34 34 38 37** | **USO84487 (meter serial no.)** |

1. **References & Further Reading..**
	1. [www.dlms.com](http://www.dlms.com/)
	2. KB00068
	3. DLMS is covered in four colored books of the DLMS User Association:
		1. Blue Book (Identification System and Interface Classes)
		2. Green Book (Architecture and Protocols)
		3. Yellow Book (Conformance Test Process)
		4. White Book (Glossary of Terms)
1. [↑](#footnote-ref-1)