17 Metering protocols

- DLMS/COSEM - IEC 62056-21 - MBus Serial/TCP - Elgama (Meters based on IEC 62056-21 / 31 protocols)

17.1 DLMS/COSEM

Introduction

IEC 62056 is a set of standards for electricity metering data exchange by International Electrotechnical Commission.

The IEC 62056 standards are the international standard versions of the DLMS/COSEM specification.

DLMS or **Device Language Message Specification** (originally Distribution Line Message Specification),[1] is the suite of standards developed and maintained by the DLMS User Association (DLMS UA) and has been adopted by the IEC TC13 WG14 into the IEC 62056 series of standards. The DLMS User Association maintains a D Type liaison with IEC TC13 WG14 responsible for international standards for meter data exchange and establishing the IEC 62056 series. In this role, the DLMS UA provides maintenance, registration and compliance certification services for IEC 62056 DLMS/COSEM.

COSEM or **Companion Specification for Energy Metering**, includes a set of specifications that defines the transport and application layers of the DLMS protocol. The DLMS User Association defines the protocols into a set of four specification documents namely Green Book, Yellow Book, Blue Book and White Book. The Blue Book describes the COSEM meter object model and the OBIS object identification system, the Green Book describes the architecture and protocols, the Yellow Book treats all the questions concerning conformance testing, the White Book contains the glossary of terms. If a product passes the conformance test specified in the Yellow Book, then a certification of DLMS/COSEM compliance is issued by the DLMS UA.

The IEC TC13 WG14 groups the DLMS specifications under the common heading: "Electricity metering data exchange - The DLMS/COSEM suite". DLMS/COSEM protocol is not specific to electricity metering, it is also used for gas, water and heat metering.

Source: https://en.wikipedia.org/wiki/IEC_62056

DLMS Master

Overview

DLMS (Device Language Message Specification) is a suite of standards developed and maintained by the DLMS User Association. COSEM (Companion Specification for Energy Metering) includes a set of specifications that define the transport and application layers of the DLMS protocol.

In DLMS/COSEM all the data in electronic utility meters and devices are represented by means of mapping them to appropriate classes and related attribute values.

Objects are identified with the help of OBIS (Object Identification System) codes (as per IEC 62056-61).

The DLMS driver allows only for readout and displaying only numeric values of DLMS object data fields. Connection via TCP (HDLC or WRAPPER) or serial (RS232/RS485) port are supported.

The setup of the DLMS driver consists of communication and tag configuration. Protocol specific parameters (except for DLMS/IEC handshake mode) apply for both serial and IP connections.

Configuration

Devices section

serial_number, **physical_address** and **logical_address** define the meter addressing parameters. Either **serial_number** (meter serial number) or a combination of **physical_address** and **logical_address** is used. If a serial number is provided, physical and logical server addresses are ignored.

⚠ Before configuring the Device section it is best to first check the connection parameters with a 3rd party DLMS utility.

client_address is defined in hex and usually depends on the authentication used. Most meters support hex 11 for no authentication.

type defines the object referencing. SN should be used for short name referencing and LN for logical name referencing.

mode defines the communications mode. If IEC is used along with comms settings for serial readout, the connection is initiated as per IEC 62056-21, at the default initial baud rate (300 7E1). DLMS-HDLC shall be used for HDLC

connections via IP. DLMS-WRAPPER is also supported for IP connections. The default setting is DLMS-HDLC.

timeout_ms defines the reply timeout for telegrams both via serial and TCP.

auth and **password** define the authentication mode and password. This can be set to None, or other authentication variant (see table below), depending on the mode configured and supported by the particular meter.
ip and **port** define the IP address and TCP port for DLMS communication via IP.

> ⚠ Connection parameters are device specific and can differ between makes, models and utility companies. For initial connection settings please refer to the configuration of the particular meter.

> ℹ When ip and port are configured, any serial port settings are ignored and connection is initiated only via IP.

Device configuration parameters for DLMS meters acquisition:

Parameter	Type	Description	Required		Default Value (when not specified)	Range	
			TCP	RTU		Min	Max
name	string	User-friendly name for a device	Yes	Yes			
description	string	Description of a device	No	No			
device_alias	string	Alphanumeric string to identify a device	Yes	Yes			
enable	boolean	Enabling/disabling of a device	No	No	1	0	1
protocol	string	Protocol to be used	Yes	Yes		DLMS	
serial_number	integer	Meter serial number	No	No	0		
physical_address	integer	Meter physical server address	No	No	1600		
logical_address	integer	Meter logical server address	No	No	0		
address_size	integer	Meter address size in bytes	No	No	1	1	4
client_address	integer	Client address	Yes	Yes			
type	string	Meter object referencing: SN - short referencing, LN - logical referencing	No	No	SN	SN, LN	
mode	string	Initial handshake mode.	Yes	Yes	DLMS--HDLC	IEC, DLMS, DLMS-HDLC or DLMS-WRAPPER	
timeout_ms	integer	Timeout in milliseconds	No	No	2500		
auth	string	Authentication.	No	No	None	None, Low, High, HighMd5, HighSha1, HighSha256, HighGmac, HighEcdsa	
password	string	Password for authentication	No (when auth is None)	No (when auth is None)			

ip	string	IP address	Yes	-			
port	integer	TCP port	Yes	-			
device		Communication port	-	Yes		PORT1	PORT2
baudrate	integer	Communication speed (bauds/s)	-	No	9600	300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200	
databits	integer	Data bit count for communication	-	No	8	6	9
stopbits	integer	Stop bit count for communication	-	No	1	1	2
parity	string	Communication parity option	-	No	none	none, even, odd	
flowcontrol	string	Communication device flow control option.	-	No	none	none	
retry_counter	integer	Number of requests, before link is considered lost (device status signals are changed) and reconnect attempt will be issued	No	No	3		
scan_rate_ms	integer	If provided and positive all reads and writes will be executed within the timeframe in milliseconds	No	No	10000		
reconnect_time	integer	Defines how often (in milliseconds) the client will try to reestablish communication with the meter after an unsuccessful attempt.	No	No	1000		

Signals section

DLMS configuration parameters creating signals:

Parameter	Type	Description	Required		Default Value (when not specified)	Range	
			TCP	RTU		Min	Max
signal_name	string	User-friendly signal name	Yes	Yes			
device_alias	string	Device alias from a Devices tab	Yes	Yes			
signal_alias	string	Unique alphanumeric name of the signal to be used	Yes	Yes			
enable	boolean	Enabling/disabling of an individual signal	No	No	1	0	1
log	boolean	Enable logging in event log	No	No	0	0	1
short_name	integer	Address of value to read (Short name).	No	No	0		

obis_job	string	OBIS codes can be accompanied by an attribute index, eg.: 1.0.1.8.0.255:2. Objects of register and extended register types do not require indexes and the scalers are applied to values automatically (though they can still be used if attributes other than the value need to be read out).	Yes	Yes			

Debugging the DLMS service

If the configuration for DLMS devices is set up, the handler for the protocol will start automatically. If the configuration is missing or contains errors, the protocol will not start. It is done intentionally to decrease unnecessary memory usage.

If DLMS does not work properly (e.g. no communication between devices, data is corrupted, etc.), a user can launch a debug session from the command-line interface and find out why the link is not functioning properly. To launch a debugging session, a user should stop dlms process and run dlms command with respective flags as in the table shown below.

Procedure for DLMS protocol service debugging:

- **Step 1**: Service must be stopped by entering the following command into the WCC Lite: **/etc/init.d/dlms stop**
- **Step 2**: After service is stopped it must be started with the preferred configuration file (JSON files found in /etc/dlms folder) and a debug level 7: **dlms -c /etc/dlms/dlms.json -d7 --dlms**
  Additional output forming options described in the table below.

- **Step 3**: Once the problem is diagnosed normal operations can be resumed with the following command: **/etc/init.d/dlms start**

DLMS command line debugging options

Option	Description
-h [–-help]	Display help information
-V [–-version]	Show version
-p [–-port]	Show output for one port only
-d <debug level>	Set debugging level
-c [–-config]	Config path

17.2 IEC 62056-21

Introduction

IEC 61107 or currently IEC 62056-21, was an international standard for a computerprotocol to read utility meters. It is designed to operate over any media, including the Internet. A meter sends ASCII (in modes A..D) or HDLC (mode E) data to a nearby hand-held unit (HHU) using a serial port. The physical media are usually either modulated light, sent with an LED and received with a photodiode, or a pair of wires, usually modulated by a 20mAcurrent loop. The protocol is usually half-duplex.

The following exchange usually takes a second or two, and occurs when a person from the utility company presses a meter-reading gun against a transparent faceplate on the meter, or plugs into the metering bus at the mailbox of an apartment building.

The general protocol consists of a "sign on" sequence, in which a handheld unit identifies itself to the metering unit. During sign-on, the handheld unit addresses a particular meter by number. The meter and hand-held unit negotiate various parameters such as the maximum frame length during transmission and reception, whether multiple frames can be sent without acknowledging individual frames (windowing), the fastest communication rate that they can both manage (only in case of mode E switching to HDLC) etc.

Next, the meter informs the handheld unit about the various parameters that are available with it in various security settings viz. the 'no-security logical group', ' the low-security logical groups' and ' the high-security logical groups'.

If the parameter required is in the no-security group, just a get.request will provide the HHU with the desired response. If the parameter required is in the low-security group, a password authentication of the HHU is required before information can be read.

In case of high-security parameters, the meter challenges the handheld unit with a cryptographic password. The handheld unit must return an encrypted password. If the password exchange is correct, the meter accepts the handheld unit: it is "signed on."

After signing on, the handheld unit generally reads a meter description. This describes some registers that describe the current count of metered units (i.e. kilowatt hours, megajoules, liters of gas or water) and the metering unit's reliability (is it still operating correctly?). Occasionally a manufacturer will define a new quantity to measure, and in this case, a new or different data type will appear in the meter definition. Most metering units have special modes for calibration and resetting meter registers. These modes are usually protected by anti-tampering features such as switches that sense if the meter enclosure has been opened.

The HHU may also be given limited rights to set or reset certain parameters in the meter.

The handheld unit then sends a sign-off message. If no sign-off message is sent, the meter automatically signs off after a previously negotiated time interval after the last message.

Source: https://en.wikipedia.org/wiki/IEC_62056#IEC_62056-21

Overview

The IEC 62056-21 standard defines protocol specifications for local meter data exchange.
Data is read out via serial port in modes A, B or C. The default initial serial port settings are 300 bps 7E1, as per standard, but can be user configured.

The driver implementation additionally allows for communication via TCP/IP, which is not described in the standard. In this case, baud rate acknowledgement is allowed however actual switchover between baud rates is not possible.

Mode A: data is requested and read out at the configured baud rate.

Mode B: data is requested at the configured baud rate and mutually switched to the baud rate proposed by the meter. Baud rate confirmation is absent.

Mode C: data is requested at the configured baud rate, new baud rate is proposed by the meter and, if acknowledged, data is read out at the proposed baud rate.

Currently data readout is supported in modes A, B and C.

For data readout it is necessary to know the port settings and the format of OBIS code representation as they can slightly differ (see table) depending on the configuration of the meter.

Configuration

Device section

The serialnumber defines the serial number of the meter. 0 (zero) will result in a '/?!' handshake string and may cause issues if more than one meter is wired to the serial port.

The baudrate defines the initial connection baud rate. In modes B and C this will be switched to what ever baud rate is proposed by the meter.

The meter_model defines the meter profile. This is reserved for future use and should be set to 1. type defines the connection mode. Modes A, B and C are supported.

⚠ If **ip** or **port** parameters are configured, any serial port settings are ignored and connections are initiated via TCP.

IEC 62056-21 device configuration parameters:

Parameter	Type	Description	Required		Default Value (when not specified)	Range	
			TCP	RTU		Min	Max
name	string	User-friendly name for a device	Yes	Yes			
description	string	Description of a device	No	No			
device_alias	string	Alphanumeric string to identify a device	Yes	Yes			
enable	boolean	Enabling/disabling of a device	No	No	1	0	1
protocol	string	Protocol to be used	Yes	Yes		IEC 62056-21	
poll_delay_ms	integer	Minimum time delay in milliseconds to wait before sending any data on port.	No	No	200		
scan_rate_ms	integer		No	No	10000		
device	string	Communication port	-	No		PORT1	PORT2
baudrate	integer	Communication speed (bauds/s)	-	No	9600	300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200	
databits	integer	Data bit count for communication	-	No	8	6	9
stopbits	integer	Stop bit count for communication	-	No	1	1	2
parity	string	Communication parity option	-	No	NONE	NONE, EVEN, ODD	
flowcontrol	string	Communication device flow control option	-	No		NONE	
serialnumber	unsigned long	Meter serial number	Yes	Yes		1	
serial_close_delay	integer	Delay before closing serial port	-	No	400		

timeout_ms	integer	Timeout of waiting for incoming request	No	No	2500		
type	string	Defines a connection mode	No	No	C	A,B,C	
t2	integer	Time to wait before acknowledging the suggested baud rate in mode C	No	No	300	200	1500
ip	string	IP address for TCP connection	Yes	-			
port	integer	TCP port	Yes	-		0	65535

Signals section

tag_job_todo defines the job sub-job. This field should contain the exact representation of the OBIS code as it is configured in the meter. For E.g. if the parameter of interest is represented as

”1.8.0*24(0147238.4*kWh)”, the value of the configuration field should be ”1.8.0*24” (excluding quotation marks).

IEC 62056-21 tags configuration parameters:

Parameter	Type	Description	Required		Default Value (when not specified)	Range	
			TCP	RTU		Min	Max
signal_name	string	User-friendly signal name	Yes	Yes			
device_alias	string	Device alias from a Devices tab	Yes	Yes			
signal_alias	string	Unique alphanumeric name of the signal to be use	Yes	Yes			
enable	boolean		No	No	1	0	1
log	integer	Allow signal to be logged. If **log is 0** **signal** will not be logged. If **log is more than 0** signal will be logged	No	No	0	0	
number_type	string	Number format type	Yes	Yes			
tag_job_todo	string	Tag job in OBIS format	Yes	Yes			

For **tag_job_todo** configuration it is best to first manually read the meter via PC or HHU (hand-held unit) to determine the exact OBIS representation format of the parameter as they can differ between meter manufacturers and utility companies.

17.3 Elgama

Overview

Elgama protocol is used for communications with *Elgama elektronika* electricity meters.

Configuration

Available meter types (use number only):

- 0 EPQM/LZQM
- 1 EPQS
- 2 GAMA300
- 3 GAMA100
- 4 ITS cl

Elgama parameters for *Device* tab

Parameter	Type	Description	Required	Default value (when not specified)	Range	
					Min	Max
name	string	User-friendly device name	Yes			
description	string	Description of a device	No			
device_alias	string	Alphanumeric string to identify a device	Yes			
enable	boolean	Enabling/disabling of a device	No	1	0	1
protocol	string	Protocol to be used.	Yes		Elgama	
device	string	Communication port	Yes		PORT1	PORT2
baudrate	integer	Communication speed (bauds/s)	No	9600	300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600,115200	
serial_close_delay	integer	Delay before closing serial port in milliseconds	No	400		
timeout_ms	integer	Timeout of waiting for incoming request in milliseconds	Yes			
id	integer	Meter serial number	Yes			
meter_model	integer	Meter type	Yes		0	4
use_time	boolean	Use system/meter (0/1) time (Default: 0)	No	0	0	1

Elgama parameters for the *Signals* tab

Parameter	Type	Description	Required	Default value (when not specified)	Range

					Min	Max
signal_name	string	User-friendly signal name	Yes			
device_alias	string	Device alias from a Devices tab	Yes			
signal_alias	string	Unique alphanumeric name of the signal to be used	Yes			
enable	boolean	Enabling/disabling of an individual signal	No	1	0	1
log	integer	Enable logging in event log	No	0		
number_type	string	Type of number (FLOAT, DOUBLE, DIGITAL, etc.)	Yes			
job_todo	string	Elgama protocol specific data set values (see files below)	Yes			
tag_job_todo	string	Tag sub job	Yes			

Files

1. Example configuration file for Elgama protocol data mapping download