

“Implementing DLMS Protocol in Meters”

A White Paper



Kalki Communication Technologies Limited

#147, 5th Main, HSR Layout, Sector 7

Bangalore 560102, India

E-mail: info@kalkitech.com

Phone: 91-80-4052 7900

Web: www.kalkitech.com

ABSTRACT - Today, DLMS is the most popular communication protocol used in metering arena across the world. It has been accepted by most energy utilities in the wake of liberalized energy market. Moreover several countries including India have framed metering specification based on DLMS which will act as a guideline for utilities to specify their metering requirements and ultimately improve consumer data collection, billing and revenue. In this wider scheme of things, one vital component is DLMS powered inter operable meters. This paper examines the various phases and design challenges in implementing DLMS protocol in meters.

1. INTRODUCTION

Electronic energy meters with communication interface are not a rarity in this age. All large scale meter manufacturers and most medium scale manufacturers have this already in place with the only exception being that the communication protocol is proprietary or loose standards. It is therefore deducted that meters already use an embedded microcontroller platform with peripherals such as UART, EEPROM. In this backdrop the migration from proprietary protocol to DLMS is mainly a matter of proper planning in using right hardware and software components.

2. PLANNING

As per Indian companion specification there are three categories of meters:

1. Energy account and Audit meters
2. Boundary (ABT) meters
3. Consumer meters
 - 3.1. Single phase meters
 - 3.2. Three phase meters

Meter manufacturer shall first decide the categories in which they are going to develop DLMS meters and then prepare detailed list of parameters and functionality for each category.

3. FORM THE PARAMETER, AND FUNCTIONALITY LIST

Indian companion specification has well documented the complete list of parameters for each meter category and it also explains the basic minimum functionalities to be supported. This could be treated as a guideline for generating parameter list and functionality list. While doing this exercise meter manufacturer should have a fundamental knowledge of DLMS data models (especially Interface class and OBIS), since Indian companion specification widely uses DLMS terminologies. If sufficient DLMS expertise is not available with meter manufacturer, it is advised to develop the expertise by undergoing training or seek consultancy service from external industry experts.

3.1 Challenges while finalizing parameter, functionality list:

Meter manufacturers who already have non-DLMS meters with fair enough parameters and functionality may face the challenge of finding new parameters in the companion specification which is not part of his existing meter implementation. In this situation, the meter manufacturer shall analyze if the newly found parameters are entirely not supported in his existing meter or if they are already implemented, but in a different way.

- a. **Parameter not supported-** In this case meter manufacturer shall start implementing those parameters in the metering IC parallel to DLMS implementation efforts. An example for this could be "Logical device name".
- b. **Parameter supported in a different way -** In this case meter manufacturers shall tweak his existing implementation of one or more parameters to perfectly align with Indian companion specification. An example for this situation could be "Activity calendar object" specified under "TOU metering" section in Indian companion specification. Meter manufacturer might have already implemented TOU in his meter, but the time zone information and corresponding tariff IDs may be stored in a different way from the DLMS model. In this case meter manufacturer only needs to modify his existing TOU data storage to DLMS's Activity calendar format (with season profile, week profile and day profile).

The end result of this exercise is to generate one/ more parameter list tables as shown in Table 1. The table shall be comprehensive capturing the finest details regarding parameters:

Parameter Name	OBIS code	IC	Attribute ID
L1 voltage	1.0.32.7.0.25 5	3	2
L1 current	1.0.31.7.0.25 5	3	2
Frequency	1.0.14.7.0.25 5	3	2

Table 1-Parameter List

Functionality list

Meter manufacturer shall now decide the DLMS services and other protocol configurations to be used. The choices offered by DLMS are:

- i. Application context
- ii. Security level
- iii. Services
- iv. Communication profile

i. Application Context - DLMS offers two referencing schemes viz., Logical name and Short name. Indian companion specification has advocated the mandatory usage of Logical name.

ii. Security Level – DLMS offers three levels of sign on authentication viz., lowest level, low level and high level.

Lowest level has in fact no authentication. Low level involves client sending a plain text password to meter, based on which meter authenticates client and accordingly accept or reject Client's associate request.

High level involves both parties (Client and Server) exchanging challenges in first round and then exchanging processed challenges in second round. In this method both parties authenticates each other and also the risk of security being hacked is eliminated as password or processing key is not exchanged through communication media.

Indian companion specification requires all three authentication levels to be supported in meters; each level will be advocated by each type of client (Public client use lowest level, Meter reader use low level and Utility will use at High level).

iii. Services - DLMS offers variety of application layer services which is used by client to access server's (meter) data. Services mandatory as per Indian companion specification are:

- Get: Service used to read normal (small) data from meter.
- Get with block: Service used to read large data from meter.
- Set: Service used to write normal (small) data into meter.
- Set with block: Service used to write large data into meter.
- Action: Service used to execute methods inside meter.
- Selective Access: Service used to selectively access meter data (example – selective reading of load profile based on date-time)

iv. Communication profile – Decide on the communication interfaces for the meter. Mandatory choices as per Indian companion specification are:

- RS485 (allowing multidrop) – primarily for utility software to communicate. This could also be used for remote connection using modem.
- Optical – for meter reading instrument.

Other communication options possible with DLMS are:

- TCP IP (Ethernet/GPRS modem)
- PLC

These communication profiles may require their own stack (example – TCP IR stack in the meter's communication chip).

4. DECIDE IF YOU WANT A SMART METER?

Above parameter list and functionality list holds good as per Indian companion specification and to most Indian utilities. But in the long run, meter manufacturer may find their interest in producing “smarter” meters. Major additions in a smart meter are:

- Connect/ Disconnect** – The meter will have a Connect/ Disconnect unit that could be controlled electronically. Meter will give access to authorized clients (Example – Utility) to operate this Connect/ Disconnect unit remotely and change its states.
- Image Transfer** - This facility allows dynamic/ field download and activation of new firmware in meter. This feature comes with additional cost of more program memory. But in the long run, this could save cost and time for utility and meter manufacturer by allowing resolving any software issues in installed meters by upgrading firmware instead of replacing faulty meters.
- Limiter, Register Monitor** - Meter could be programmed to monitor certain parameters, values of which on crossing threshold could trigger pre-programmed control actions from within meter all without any external supervision.

5. DECIDING SOFTWARE STACK

Meter manufacturer at this stage can decide on how they wish to proceed about implementing the DLMS stack. The possible options are listed below:

- Develop stack by their own** – This option could be thought of if the meter manufacturer has a dedicated software resource team. By choosing this option, meter manufacturer should also foresee responsibility of maintenance of the code for the long run. Inevitably, the initial basic code will have to be modified periodically to cater to the different utility requirements, add new features and mainly cope up with constantly evolving DLMS standard. Time-to-market would be more in this approach.
- Purchase stack from third-party vendors** – The second option is to purchase DLMS library from third party vendors and integrate to metering platform. This is a widely followed practice across energy industry by OEMs. Time-to-market will be significantly less with this approach.
- Purchase DLMS enabled chips** – This option is suited for meter manufacturers interested in adding DLMS enabled micro-controllers to their metering platform there by reducing time to integrate third party library or develop stack by own. This approach would have the lowest time-to-market.

6. COMMUNICATE WITH SOLUTION PROVIDER

Depending on the approach, software provider here means software wing within metering company OR third party library vendor OR DLMS chip vendor. Metering team can enter into a dialogue with DLMS solution provider and reach into mutual agreement on the following:

- All the parameters and functionalities would be supported in the solution.
- Estimate of memory footprint (RAM and ROM) required for the software.

- Ways for interfacing communication software with metering code is smooth.

7. CHOOSING THE RIGHT ARCHITECTURE – SINGLE CHIP OR MULTI CHIP

Once approximate memory footprint information is obtained from communication software provider, meter manufacturer can choose between using single chip or multiple chip architectures.

i. Single Chip Solution

The Figure-1 architecture diagram could be considered if the memory footprint for communication software is low enough to fit in the metering micro-controller alongside metering code. While calculating memory availability, a small buffer has to be kept taking into account that memory footprint of communication code was approximate and not accurate. If the combined memory requirement (metering + communication) lies near the boundary of available memory, it is advised not to go with this architecture.

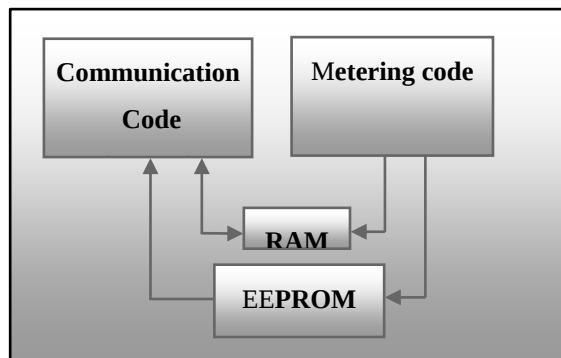
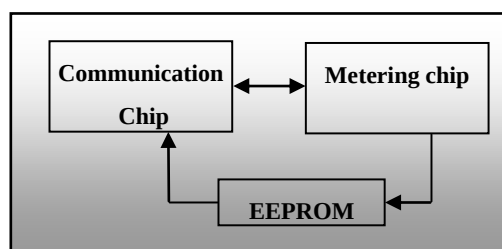


Figure 1: Single Chip Solution

The advantages of this architecture includes lesser number of micro-controller resulting in lower cost and size of hardware, low power usage and ease of software integration of metering code and communication code. However care has to be taken while designing the software architecture so that one application does not hamper the other. Metering code is considered vital and has higher priority over communication. So the software design should contain a good switching mechanism on time sharing basis between metering code and communication code. The communication code also has to be specifically designed not to take longer time to complete its routines which results delay in calling metering code and potentially hampering metering functionalities.

ii. Multiple chip solution

The Figure-2 architecture diagram involves at least two microcontrollers – one for metering and other for communication (DLMS). The advantage of this architecture is mainly isolation of metering and communication there by eliminating risk of one affecting the performance of the other. Other advantages include more resources exclusively for communication code (Example – UART, Timer, Flash memory, RAM) without having the burden of sharing with metering code. Availability of more memory opens up the possibility to scale up the meter in terms of number of parameters/ functionalities for a different market in the long run without much change in metering part.



8. IMPLEMENTATION - TARGET FUNCTIONALITIES

After finalizing hardware architecture, meter manufacturer may now enter into implementation phase. The first step is to implement the target interface which includes the following:

- **Communication interface** - Initialize communication interfaces (Example - UART) and implement transmit and receive functions (or Interrupt Service Routines).
- **Timers** - Initialize timer (preferably with millisecond resolution).

In single chip solutions, these steps would be easier since meter manufacturer is well experienced in dealing with target resources.

9. IMPLEMENT – CONFIGURATION INTERFACE

Even though multiple utilities ask for DLMS enabled meters, the parameter list required for each utility will be different. If not foreseen and designed accordingly, meter manufacturer will end up developing separate code bases for each utility costing them time and valuable resource.

It has to be designed in such a way that the basic code remains the same irrespective of utility requirements. Utility specific requirements will be incorporated in one or more configuration files without having to change anything in the basic code base. This allows quick utility specific customization, testing and maintenance. Proper design of configuration interface could even open up the possibility to use a pre-compiled basic code which fetches configuration data in run time. This pre-compiled basic code could be used for all DLMS utilities.

10. IMPLEMENT – DATA INTERFACE

Last phase in implementing DLMS in meter is implementation of data interface. Data interface is a code which accepts request (Get, Set, and Action) from DLMS software, access the actual data storage location (RAM / EEPROM), perform the requested operation (read/ write) and return back the result to DLMS software. Data interface shall be designed for easy and fast memory access.

The challenge in data interface comes while handling bulk data such as profile generic (Example - Load survey, Billing Profile).

11. STORE PROFILE CAPTURE VALUES IN CONTIGUOUS LOCATIONS

It is recommended to store profile capture values in contiguous memory locations for ease of handling in code and better latency while reading. This approach may not be highly optimized for memory but is desirable from performance and coding point of view.

12. REDUCE NUMBER OF MEMORY WRITE OPERATIONS

If there are any issues regarding multiple writes to same EEPROM location damaging memory in the long run, then you may choose to switch the storage location. The actual location (which varies each time) shall be updated in a look up table which the data interface can use to resolve and get the actual address.

13. SELECTIVE ACCESS BY RANGE

While designing data interface, care has to be taken to implement selective access by range without much complexity. Assuming the Table-2 to be sample load profile data, a client may request for load profile data captured between 06:30 to 07:30. Such requests shall be easily and quickly resolved in data interface to get the appropriate start and end EEPROM addresses corresponding to 06:30 and 07:30 respectively.

Timestamp	Volt	Current	Frequency
6:00:00 AM	230	1	50
6:30:00 AM	231	1	51
7:00:00 AM	229	2	49.5
7:30:00 AM	230	1	50
8:00:00 AM	230	1	50

Table 2 - Load profile sample data

CONCLUSION

DLMS protocol will remain the choice of utility communication protocol in days to come. For meter manufacturers who have not yet adopted this standard, it is high time to kick start their efforts towards the same. Rather than seeing as a compulsion imposed by utilities or government bodies, meter manufacturer can see this as an opportunity to contribute to the standardization efforts going all over the world. These efforts will create a level playing field in the metering industry and also opens up the market to all. Special attention is drawn to the fact that one step by implementing DLMS will open up new markets in and outside the country for you.

REFERENCE

1. IEC 62056 – 42,46,47,53,61,62
2. Indian Standard - data exchange for electricity meter reading, tariff and load control-companion specification

ABBREVIATION

BIS	-	Bureau of Indian Standards
DLMS	-	Device Language Message Specification
EEPROM	-	Electrically Erasable Programmable Read Only Memory
GPRS	-	General Packet Radio Service
GSM	-	Global System for Mobile communication
HHU	-	Hand Held Unit
IC	-	Integrated Circuit
OEM	-	Original Equipment Manufacturer
RAM	-	Random Access Memory