MOSCAD TECHNICAL NOTES

DESCRIPTION OF THE SEVEN LAYERS OF THE OPEN SYSTEMS INTERCONNECTION (OSI)

Introduction

This document describes the MDLC (Motorola Data Link Communication) protocol as implemented in MOSCAD family Remote Terminal Units (RTUs).

The MDLC protocol is designed for optimal operation in SCADA systems which use a variety of communications media such as radio, trunked radio, wire-lines, etc. All these constitute the *physical layer* of the protocol.
The MDLC protocol is characterized by the following main features:

* The communication protocol is based on the Open Systems Interconnection (OSI) model published by ISO. The protocol comprises the seven recommended layers adapted for SCADA, where, among others, every RTU is simultaneously a distributed control unit and a communication node serving itself as well as the other units.

* The protocol is efficient both for transferring small quantities of information (such as measurements and discrete statuses) and for transferring large quantities of information such as downloading applications software including database, process, etc.

The following table gives an overview of the main MDLC features:

Table 1: MDLC features

Feature	Description
Data Integrity	Sophisticated recovery procedures built into various network layers to ensure a high degree of end-to-end data transmission safety
Data Authentication	Smart procedures to ensure that a message came from a legitimate source or goes to a legitimate source (consult product group for this feature)
Access Control	A comprehensive multi-level password scheme results in high level of data privacy
Data Download and Upload	Each RTU can be downloaded from the center with application and configuration software to ensure minimal commissioning time and to ease database update and re-configuration procedures. Locally configured RTU database can be easily uploaded to update the MCC or the SCC
Store-and-Forward and Network Nodes	Each RTU can become an intelligent tandem processing node to provide optimum use of the existing communications network

The seven layers and their functions are summarized in the following table:

Table 2: MDLC seven layers summary

Layer	Function
Layer 1: Physical	The physical layer caters for communications over conventional radio, trunk radio, or telephone line. Since radio communication uses a shared channel, the radio physical layer is responsible for the channel access and collision control.
Layer 2: Link	The link layer is totally separated from the physical channels and therefore is independent of the layer 1. The link layer ensures proper communications over a physical link. To this end the link layer arranges the data in variable-length frames and attaches addresses, frame sequence numbers, and Cyclic Redundancy Code (CRC) to the frames.
Layer 3: Network	The network layer is responsible for the establishment of end-to-end communication path in a network. This is necessary since communications can take place via more than one link and message can travel via several transit nodes (repeaters, store-and-forward RTUs) until it reaches its final destination.
Layer 4: Transport	The transport layer ensures end-to-end integrity of the information flow between two nodes in the network. This is achieved by means of remote end acknowledgment that data has been received completely and transmitted in the correct order to the upper layer.
Layer 5: Session	The session layer allows the definition of any number of entities capable of conducting simultaneous sessions with a parallel entity in a remote unit. This enables transparent communications between multiprocessing machines without interference between the applications.
Layer 6: Presentation	The presentation layer checks the integrity of the information received from various applications. This layer also performs data compression, authentication and encryption, if required.
Layer 7: Application	The application layer performs the task of interfacing to the various applications such as data transfer, configuration downloading, application software monitoring, remote diagnostics, etc.

The importance of the MDLC protocol for the user is that MDLC is a transparent protocol which liberate the user from technical constraints and complexities of the network operation and allows him to concentrate on the application. This fundamental division between the network and the application usually occurs at the Network Layer. Thus the boundary is the Network Layer and the functionality provided by the three lower layers is known as the Network Service. MDLC's Transport Layer provides additional enhanced quality of Network Service suitable to SCADA applications.

The Network Service layers (1-3) are those communicating with the intermediary sites (relay systems) while the upper layers are not involved in the network function and are communicating directly, each layer with the appropriate layer, what is called Peer-to-Peer Communications. The following figure provides an overview of the communication between layers in different network locations.

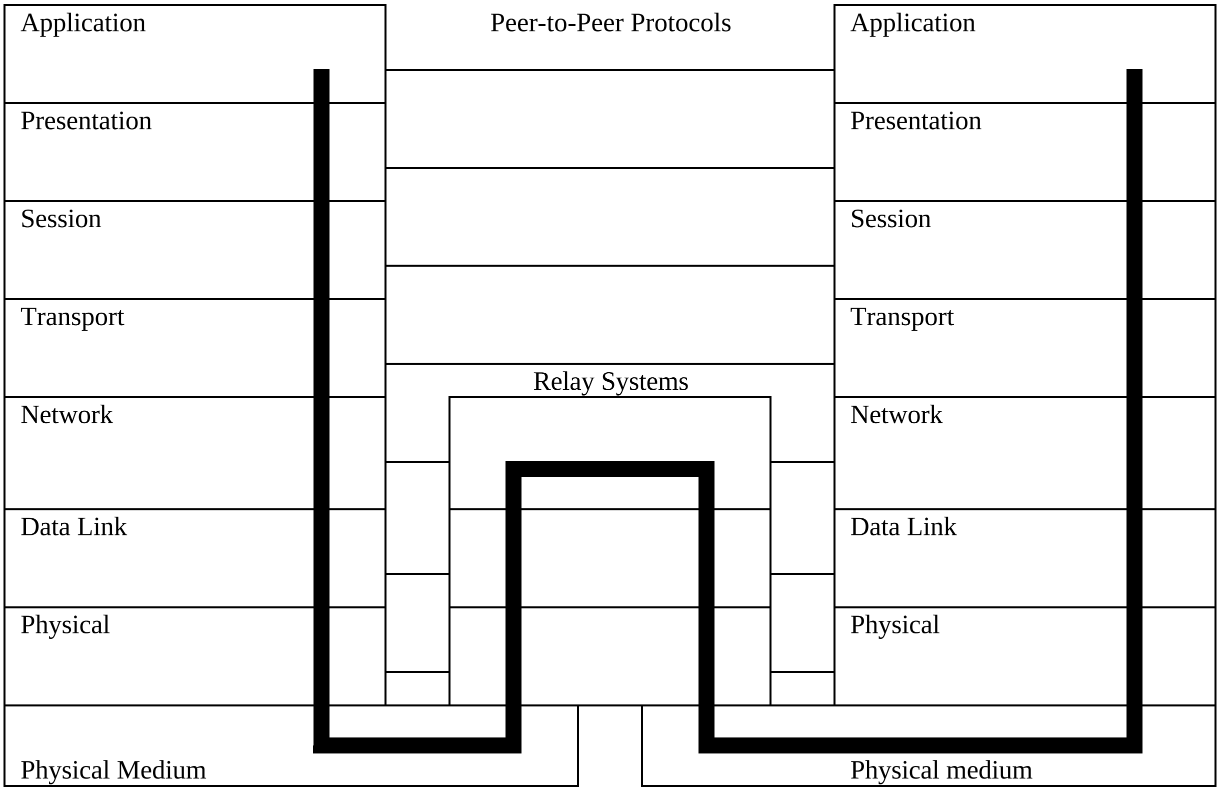

Figure 1: MDLC Layers Overview

In the following, a more detailed description of each layer is presented.

1. Physical Layer (OSI Layer 1)

* This is the lowest layer in the OSI model and comprises the various communication ports and the associated software. The software contains all the specific handling required by the port and provides an identical interface to the *link layer (OSI 2)*. In this way, the physical layer defines a standard entity for the link layer above it.

* The software is flexible and adapts itself to the number of ports and their various types as defined by the user. The possibilities of defining a large number of ports of various types does not impair the software's efficiency.
The software creates entities according to the types and the number of ports defined.

* The *physical layer* provides solutions for communication over *conventional radios* at speeds of at least up to 4800 bps, over *trunked radio* at speeds of up to 2400 bps, over *wirelines* at 1200/2400 bps and over *serial channels* (RS-232 and RS-485) at speeds of up to 9600 bps.

* The *physical layer* handles the procedure for accessing the communication channel as required by each and every channel. However, this done while according to the appropriate priorities (for DATA and ACK word types) in order to prevent transmission collisions. A mechanism is also provided for staggered channel access. The algorithm is such that, with each successive transmission retry to any RTU, the chance of collision decreases until it disappears.

* It is possible to define the physical ports of the system to interface with the higher protocol layers or to operate with an application to provide hard-copy printout, for operating terminals or for emulating external protocols.

2. Link Layer (OSI Layer 2)

* All the entities of this layer (each of which handles any of the physical channels) are identical. The flexibility of the system allows creation of a number of instances (entities) in accordance with the number of ports defined in the system.

* The function of this layer is to ensure proper communication over the communication channel. The information is stored in variable-length frames in which the *link layer* protocol contains the following fields per each DATA frame:

- The address of the unit to which the DATA is transmitted
- The address of the transmitting unit
- The number of the frame
- CRC for error detection

* Dual addressing is used to enable RTU-to-RTU transmission without control center intervention or transmission to several centrals or to another RTU which may serve as a subcenter for a lower hierarchy of RTUs.

* During data reception, the address is compared by hardware at the physical level, in order not to waste valuable software time on checking addresses not intended for that specific RTU. This preliminary screening enables the assignment of up to four different addresses per RTU:

1. An individual RTU address.
2. A base system address for broadcasting to a set of RTUs.
3. A local port address - this address (which is relative to the base address) is common to all RTUs and eliminates the need during programming or configuration to have knowledge of the individual RTU address.
4. A spare address for future communication modes.

* A link entity associated with a channel (such as a radio, for example) receives information from several RTUs (part of which is intended for that RTU and other parts passing through it). It transmits an acknowledgment (ACK) to each RTU according to the DATA received from it. Therefore, an ACK word must be used in addition to the DATA word, since the RTU to which the DATA is being transmitted is not necessarily the same RTU to which the ACK is being transmitted.

The ACK word enables the receiving site to identify the missing frames and request retransmitting **only those frames**, in order to save air time by not repeating correctly received frames.

* The CRC is 32 bits or 16 bits per CCITT standard definition. In radio communication channels it is preferred to use 32-bit CRC. The CRC calculation is performed by hardware in order to save computer time for the process.

* The frame synchronization (FLAG), at the beginning and at the end of each word, is transmitted in different ways for different physical ports. For radio or wire-line, the flag is similar to the one used in HDLC protocol. For RS-232 communication via UART, a certain combination (CHAR) of eight bits is provided together with an appropriate coding method to ensure that the FLAG combination does not appear within the DATA word itself.

3. Network Layer (OSI Layer 3)

* A SCADA network may use more than one communication link (wireline and/or various radios). It therefore comprises a network in which part of the communication is routed from RTU to RTU or from RTUs to the control center via a Store & Forward Repeater, which , for this purpose serves as a node in the network.

* The *network layer* is responsible for routing packets in the network via the various nodes in order to enable communication between RTUs in the network. This makes it possible to access the applications and data from any port in the system [at RTUs or at control center(s)]. The RS-232 ports of the various RTUs, are used for purposes of definition, monitoring, modification, diagnostics, error logging, etc.

* The MDLC protocol supports a network of a practically unlimited number of links (radio, lines, RS-232-C, etc.) and RTUs. The Store & Forward mode is a special case of a network in which the node may receive, store and retransmit to the same RF link. This mode of operation is useful when transmitting between sites that are out of the RF coverage.

4. Transport Layer (OSI Layer 4)

* The *transport layer* ensures the END-TO-END completeness of the information between the transmitting and the receiving points. This layer transfers the DATA in an orderly fashion to the *session layer* above it. This layer assigns sequential numbers to the packets (independent of the numbers assigned by the *link layer*) and also executes the transmission of an ACK word, making sure that the DATA is complete and all packets were transferred in the appropriate order to the layer above.

* The *transport layer* performs multiplexing, thus enabling several session entities (logical channels) to operate via either a single physical port or several physical ports. A possibility is provided to define any number of logical channels independently from the number of channels defined in the *physical layer*.

5. Session Layer (OSI Layer 5)

* The *session layer* enables the definition of any number of entities (instances), each of which is capable of conducting a "session" or conversation with a parallel entity in another point (RTU, center or subcenter). These entities allow the unit to simultaneously conduct several sessions between these sitest, i.e., to simultaneously run several tasks or applications such as data transfer, diagnostics, monitoring, etc., without causing interference. The *session layer* is handling the following:

- Start session
- Synchronization of message direction
- End session
- Abort session
- Re-synchronize session

* The *session layer* also supports transfer of **short** (single-frame) messages from certain applications at one site to the parallel application at another site without the need to start a session.

6. Presentation Layer (OSI Layer 6)

The *presentation layer* performs processing of the DATA words, received from the various applications within the packets. It performs the following:

- Checks that the information transferred to the application is complete
- Data compression

7. Application Layer (OSI Layer 7)

The *application layer* performs all functions required for maintaining a SCADA system:

* Enabling bi-directional data transfer upon request from other databases in the network.

* Handling downloading of the configuration of a site:

- I/O modules definition
- Communication ports definition.

* Handling the downloading and monitoring of the application software to the sites (as defined by the user in *ladder diagram* language) , including:

- Definition of the data structure (tables with rows and columns)
- Object code of the processes (compiled ladder rungs)
- Real-time symbolic monitoring of database and processes.

* Transmitting events and short messages related to the application.

* Broadcasts of data to several sites simultaneously.

* Remote diagnostics of the hardware and the software.

* Retrieval of error messages stored in error logs at each site.

* Calibration of I/Os in A/D and D/A modules.

* Communication analysis and accumulation of statistical data.