LMS MODULAR

Cerberus Dati Data Link

Data Link Protocol Description

Document No.	**e1152**
Edition	07.93
Supersedes	CDI-135-017-E
Manual	**-**
Section	-

Copy No. _____

Table of contents

1 Document History

Date	Notes
3.2.92	Original version based on previous document STD-DOC.WPS, version 11.7.91
21.2.92	Document update: time-out values added
21.8.92	English translation
19.4.93	Title changes, document number added and DTPL 128 byte limit added

2 Scope of the Document

This document describes the communication protocol called CDDL(Cerberus Dati Data Link). The protocol operates at physical and data link levels. It is a proprietary protocol of Cerberus Dati S.p.A. - Italy .

CDDL is an asynchronous asymmetrical (in master to slave configuration) point-to-point protocol to be used in security system applications.

At the present, the CDDL protocol is used in the communications between the Gateway and Standard Subsystem, the Gateway and third-party host (FHI Pad Applications) and the Gateway and Cerberus Dati proprietary host.

3 CDDL Protocol - physical level

The physical connection (ISO-OSI level "physical layer") between two devices is a RS-232-C serial line configured at 8-bit characters and 2 stop bits. The parity bit is disabled (set to none). The standard transmission rate is 9600 baud (other available transmission rates are 150, 300, 600, 1200, 2400, 4800 baud).

4 CDDL Protocol - Data link level

The CDDL protocol at the data link level belongs to the "polling-selecting" type, with "stop-and-wait" transmission and positive acknowledgement. This protocol needs a master station. Usually the master is the Gateway, that starts the data transmission (polling) or asks the permission to transmit (calling) to the slave station, usually a sub-system. The slave station is normally in a receiving status and it transmits only if it is requested to do so.

4.1 Transmitted data format

The data transmitted among the stations are collected in frames. A frame has the structure shown in Fig.: 1

The frame is composed by:

- start frame sequence [DLE, SOP] (start of packet)
- frame header DTPL
- packet body DTP (Data Transmission packet)
- CRC-2 checksum (Cycling Redundancy Check 2nd order)
- end frame sequence [DLE, EOP] (end of packet)

The DTPL header (Data Transmission Package Length) is composed by two bytes containing the packet body DTP length in characters. The byte 0 is the most significant, while the byte 1 is the less significant. The length includes the packet body stuffing, when used (refer to par. 4.1.2 for a description of the byte stuffing technique).

The maximum theoretical value foreseen for the DTPL (DTP length) is 512 characters, but memory reasons reduce the practical limit to 128 byte. The DTP is composed by the sequence of bytes to be transmitted.

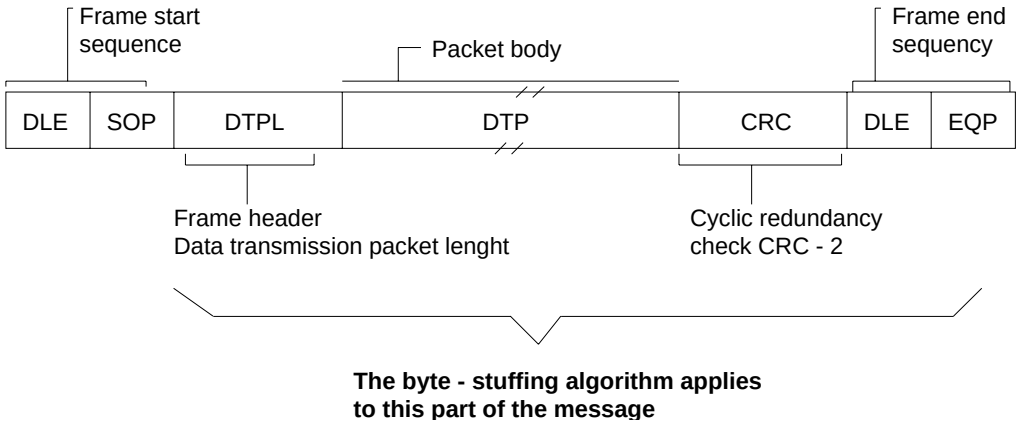

Fig.: 1

4.1.1 CRC

The CRC is composed by two bytes and it is computed using the CRC-2 polynomial including all the characters: starting from the start frame sequence, included (DLE + SOP + DTPL), to the whole packet length (DTP).

The CRC-2 algorithm performs a XOR of all the characters including the starting frame sequence. The algorithm result is inserted in the message putting the most significant byte in front of the less significant byte.

4.1.2 Control bytes masking

To avoid that bit masks equivalent to protocol control characters possibly present in the DTPL, DTP or CRC fields could generate malfunctioning, the CDDL protocol uses a byte-stuffing technique. The control characters to be masked and the masking sequences are as follows:

Control character	Masking sequence
EOP	DLE, NUL
DLE	DLE, DLE

The masking algorithm applies to the complete frame, with the exclusion of start and end frame sequences [DLE + SOP, DLE + EOP].

The maximum overall length of an on-line frame is 1036 characters (512 x 2 DTP body packet characters + 2 x 2 DTPL header characters + 2 x 2 checksum characters + 2 start frame characters + 2 end frame characters).

4.2 Polling

The master station issues at regular intervals the polling sequence [DLE, ENQ].

The [DLE, ENQ] sequence reception enables the slave station to transmit.

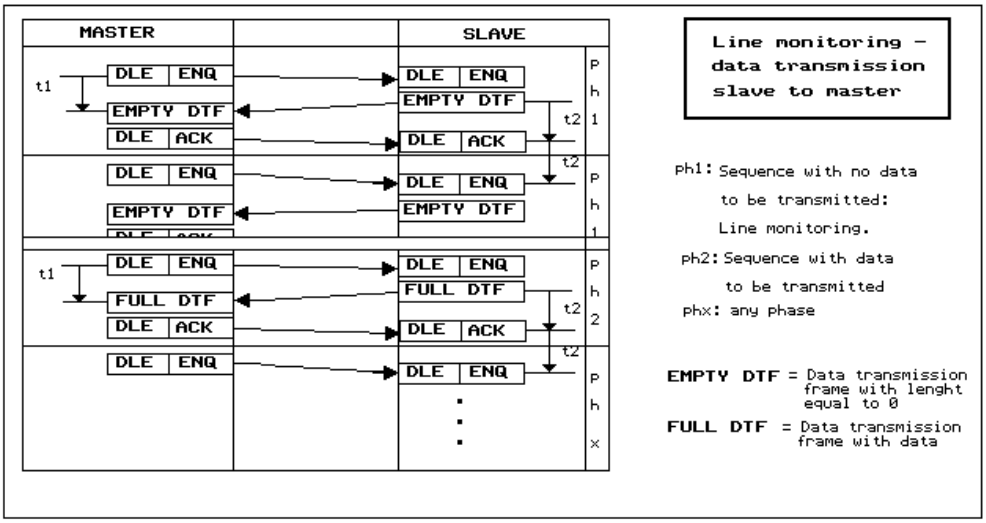

Fig.: 2

The slave station answers anyway to the selection sending a packet. If no data to be transmitted are available, the packet will be empty (empty DTF).

The polling sequence is shown in Fig.: 2

4.3 Calling

Using the [DLE, SYN] sequence the master station requests to the slave station to be enabled to transmit a packet.

If the slave station is ready to receive, it answers with the [DLE, ENQ] sequence, otherwise it sends the [DLE, EOT] (wait) sequence.

The calling sequence is shown in Fig.: 3 and Fig.: 4.

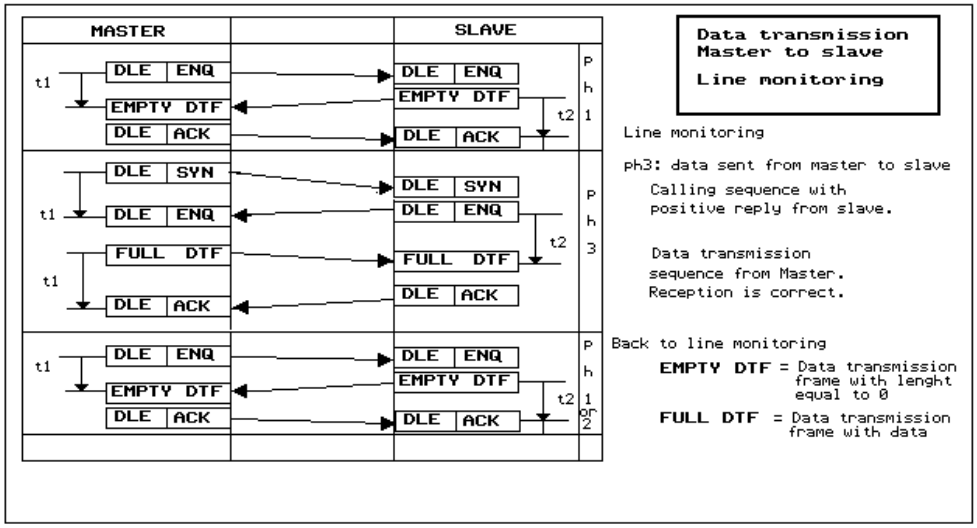

Fig.: 3

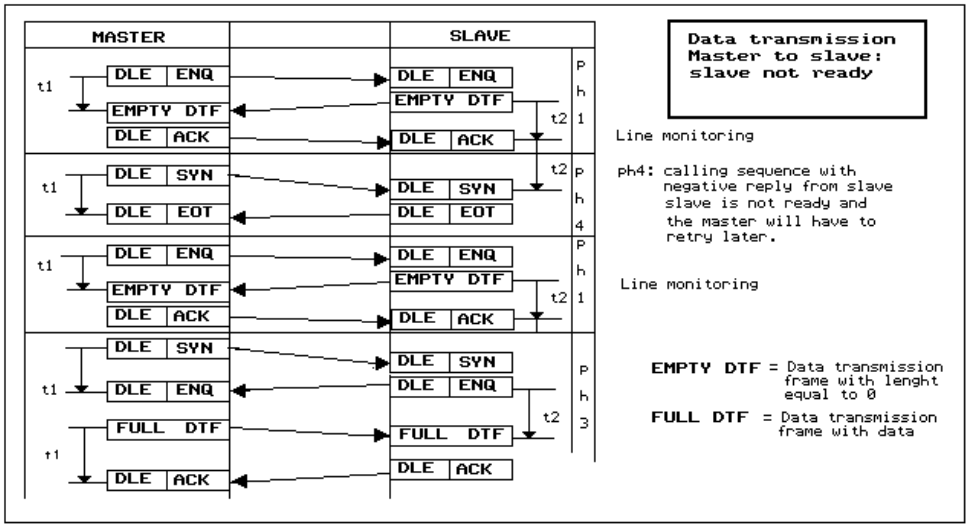

Fig.: 4

4.4 Waiting

The master station sends a packet without data instead of a polling sequence when it wants to signify it is no longer able to receive and manage information.

The slave station answer must be [DLE, ACK] if it correctly understood the message. Otherwise the answer must be [DLE, NAK].

The waiting sequence is shown in Fig.: 5.

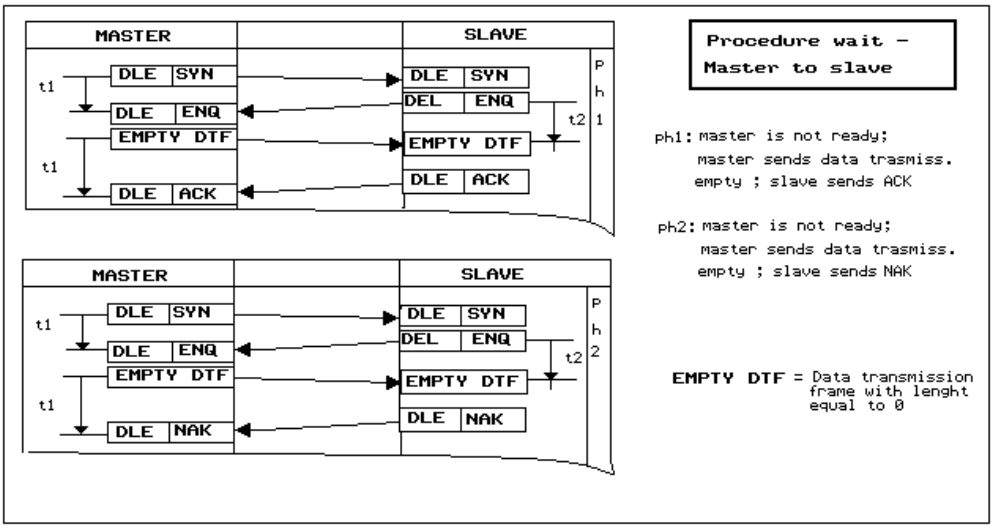

Fig.: 5

4.5 Errors management

The transmitting station enters an answer-waiting status after sending a data packet.

A successful transmission (i.e. the receiving station verifies that the CRC is correct) is signalled to the receiving station by the [DLE, ACK] sequence.

The packet is transmitted again if the [DLE, NAK] sequence or any other characters is received (even if before the transmission end), as well as if the answer is missing after the T1 or T2 time.

The Slave station of course cannot transmit the packet again if it has not been again selected .

The Fig.: 6 and Fig.: 7 show the possible sequences in both the two directions the data can flow.

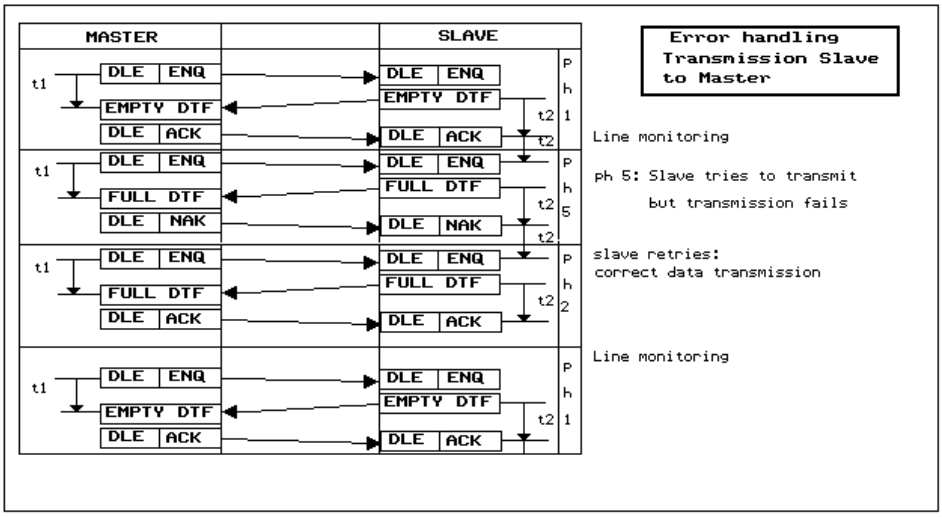

Fig.: 6

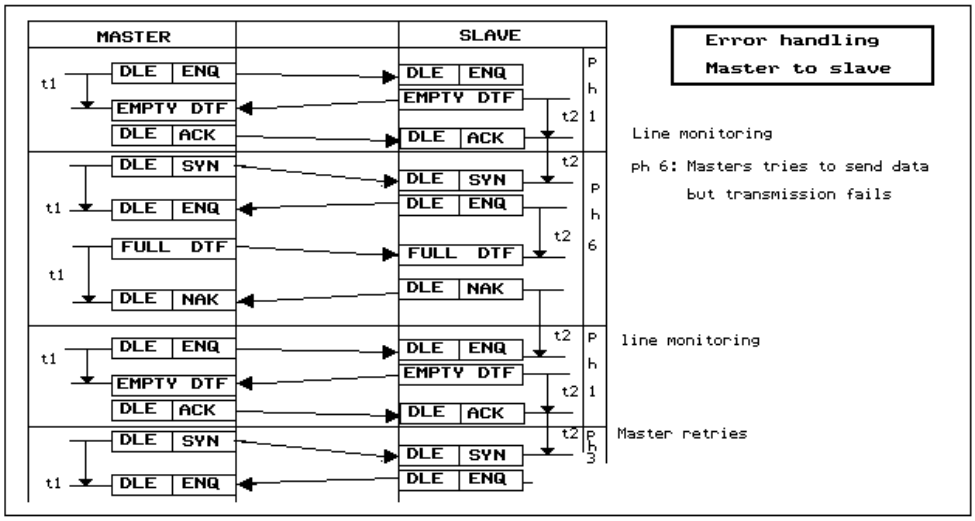

Fig.: 7

4.6 Time-out control and recovery procedure

These procedures are different in case of master or slave stations.

Below both cases are separately described.

4.6.1 Slave side

Two time counts are managed.

The first time-out, T2 (see the available range in Annex A), is used to verify the continuity of the master station polling activity. The T2 timer is reset and restarted each time a polling/calling/waiting selection is received or when a [DLE, ACK] sequence is waited after a transmission.

The second control, TC, is used during a packet reception, starting from the [DLE; SOP] sequence up to the checksum characters, to verify that the transmission is not interrupted. The maximum tolerable delay is 0.8 seconds between two consecutive characters.

When either T2 or TC expires, there are two results:

- the slave station gets back in a polling/calling wait status
- the error counter is incremented

This counter is reset after a successful data exchange; when it reaches the value 10, a line fault status is activated, for slave station internal use.

After a line fault, the communication gets back into a "no fault" status when a data packet sending toward the master station is successful (i.e. when an acknowledgement message is received).

4.6.2 Master side

Two time counts are managed.

The first time-out, T1 (see the available range in Annex A), is used to verify that meaningful answers are received from the slave station after a polling/calling/waiting sequence.

The T1 timer is reset and restarted each time a polling/calling/waiting selection is received or when a [DLE, ACK] sequence is waited after a transmission.

The second control, TC, is used during a packet reception, starting from the [DLE; SOP] sequence up to the checksum characters, to verify that the transmission is not interrupted.

During this phase, the maximum tolerable delay is 0.8 seconds between two consecutive characters.

The fault line treatment is similar to that described in Section 4.6.1. The communication gets back into a "no fault" status when an acknowledgement message is received.

4.7 Control characters

Here below you can find the hexadecimal control characters used by the standard protocol:

DLE	0DFH
SOP	007H
EOP	0EFH
ENQ	005H
ACK	006H
SYN	016H
EOT	004H
NAK	015H
NUL	000H

5 Annex A - Protocol time-out intervals

T1 and T2 are currently set at 1.50 seconds. This time can be changed by Cerberus Dati for specific applications.