LonTalk

LonTalk communication protocol

Supported device types and versions
Communication line configuration
Communication line protocol parameters
Communication station configuration
Communication station parameters
I/O tag configuration
Literature
Changes and modifications
Document revisions

Supported device types and versions

The LonTalk communication protocol supports communication with 3150 and 3120 "neuron" chips of the LonWorks technology from the Echelon company by means of reading/writing values of network variables. The communication is performed by means of the RNI interface (SLTA, PCLATA, iLON-10, ...) and the standard Win NT system driver, which is installed after installing the "Echelon OpenLDV". This driver can be obtained from http://www.echelon.com/. Versions known to work are "Echelon OpenLDV 4.0" and "Echelon OpenLDV 5.0". Communication has been verified these interfaces:

- SLTA/10
- PCLTA/10
- iLON-10
- SmartServer 2.0 Controller
- U10 USB Network Interface

Data exchange is performed by the SNVT set (Standard Network Variable Types) and by several user types.

Communication line configuration

- Communication line category: LONWORKS

Communication line parameters

Communication line configuration dialog box - tab **Protocol parameters**.
They influence some optional protocol parameters. The following line parameters can be set:

Tab 1

Parameter	Meaning	Unit	Default value
Full Debug	High level of communication tracing. It shows the read values of I/O tags and other debug information.	YES/NO	NO
Open/Close Interval	A time interval. After it passes, a re-open action over the interface will be done. 0 (zero) disables this function.	min	0

Communication station configuration

Communication protocol: **Echelon LonTalk**.
The station address (node) consists of the required address parameters (i.e. "Subnet/Node Addressing"):

- **Domain** is a domain number (0 or 1).
- **Subnet** specifies a network ID (0 up to 255), as a decimal number.
- **Node** is a node number in the network (0 up to 127), as a decimal number.

Neuron ID is an optional address parameter. If it is defined, this address executes the polling of the network variables of the node. Neuron ID address is in the hexadecimal form.

If it is necessary to send a network variable value with another address type than "Subnet/Node Addressing", you can configure *Group* or *Broadcast* address parameters.

- "Group Addressing" parameters:
  - **ID** specifies a group ID in the domain (the decimal number ranging from 0 to 255).
  - **Member** is a member ID of a target node in this group (the decimal number ranging from 0 to 63).
  - **Size** is a group size (2 up to 64). The group size is unlimited when you set 0 (zero).
- "Broadcast Addressing" parameters:
  - **Subnet** defines a network ID to which a broadcast message is sent. Set a required value *Domain*.

- **Backlog** is an estimated number of ACKD responses or request/response messages.

  Parameter description from the literature 1: This field specifies an estimate of the channel backlog that would be created by an acknowledged or request/response message broadcast using this address. It should be set to the expected number of acknowledgments or responses. For example, this might be the worst-case number of nodes expected to respond to a message on a channel. If this is unknown, this field can be set to zero in which case a backlog of fifteen is assumed.

Station protocol parameters

They influence some optional protocol parameters. The following station protocol parameters can be defined:

Table 1

Full name	Meaning	Unit	Default value
Rpt_timer	Value of the parameter is written into the address array *SendAddrDtl* when the packets "Poll request" and "Write NV value" are sent. See Note 1. Parameter description from the literature 1: This field specifies the time interval between repetitions of an outgoing message when unacknowledged–repeated service is used.	-	3
Retry	Value of the parameter is written into the address array *SendAddrDtl* when packets "Poll request" and "Write NV value" are sent. See Note 1. Parameter description from the literature 1: This field specifies the number of retries for acknowledged, request/response, or unacknowledged–repeated service (0–15). The maximum number of messages sent is one more than this number.	-	1
Tx_timer	Value of the parameter is written into the address array *SendAddrDtl* when packets "Poll request" and "Write NV value" are sent. See Note 1. Parameter description from the literature 1: This field specifies the time interval between retries when acknowledged or request /response service is used. The transaction retry timer is restarted when each attempt is made, and also when any acknowledgment or response (except for the last one) is received. For the request/response service, the requesting node should take into account the delay necessary for the application to respond when setting the transaction timer.	-	3
Response Timeout	Timeout for receiving the response to the polling request. Total time is tx_timer * retry + Response timeout.	ms	500
Write ACK Timeout	Timeout for receiving the confirmation of the sent packet with a new value. Total time is tx_timer * retry + WAT.	ms	1500
Write ACK Immediately	If its value is YES, writing new values into the network does not wait for confirmation (with *Write ACK Timeout*). They are immediately acknowledged as received.	YES /NO	YES
Skip Reading after Failed Polling	If the value is YES, the reading of I/O tag values is stopped by the polling on the given station (node). It helps to speed up the value reading from the network variables when some nodes do not communicate (they are switched off). The communication is not delayed because of waiting for the responses of polling from a non-functional node. The polling always starts in a usual way, however, after the first error the reading of the remaining network variable is interrupted and the next node is read.	YES /NO	NO

Note 1: The parameters *rpt_timer*, *retry* and *tx_timer* can get values ranging from 0 to 15. Conversion table for times in milliseconds is stated in the document LonWorks communication line configuration and in the literature nr.1.

I/O tag configuration

I/O tag address consists of:

- **Selector** - required parameter - basic identifier of the I/O tag within the range of 0..16383, as a decimal number or as a hexadecimal number with a prefix # (e.g. #3A).
- **Type** - value type of the network variable, required, options are: SNVT, ushort (1 byte, unsigned), short (1 byte, signed), ulong (2 bytes, unsigned), long (2 bytes, signed), float_type (4 bytes IEEE-754), analInp (a special format Weidmuller), long_4 (4 bytes, signed) or struct (1 and more bytes).
- **SNVT** - if you select the SVNT type in the parameter **Type**, then you must define a SVNT type number (0..255), as a decimal number.
- **Bit** - if you select the I/O tag type either Di or Dout, you can define a number of the bit within the frame of a 2-byte value word (0..15).
- **Poll** - allows the network variable polling. Polling is performed within the interval according to the station time parameters (polling parameters).
- **Priority** - indicates a higher priority for writing and polling (in terms of LonWorks network communication).
- **Authentication** - enables the authentication for the network variable writing and polling.
- **Service** - for output I/O tags, you must define a type of network communication for writing (ACKD, UNACKD_RPT, UNACKD); standard secured transmission type ACKD.

Value type **struct**

This type enables to split data of any length to the bytes or bits. You must set an index 'byte' (numbered from 1) and eventually the bit number (0 up to 7) for I/O tags of Di type. the separated bytes are interpreted as non-signed numbers from 0 up to 255.

Note: Using the *nodeutil* utility it is possible to:

- Connect to LON device:
  a) either by entering the hexadecimal 6-byte Neuron ID via the menu "*A -- (A)dd device to list.*"
  b) or by pressing the service pin on the LON device, which causes the message "*Received an ID message from device x*" to appear.
  Then select the appropriate device via the menu "*G -- (G)o to device menu....*".
- Read configuration information via the menu "*C -- Application (C)onfiguration structures*.
- Read the list of network variables via the menu "*N -- (N)etwork Variable configuration table.*".
  The list contains indexes of variables, their selectors, direction (in/out), and other parameters.
- Read values of individual network variables via menu "*P -- (P)oll network variable.*".
- Save the list of network variables in HC format to the file via the menu "*L – (L)ist network variables.*"
- Save the XIF file to the interface file via the menu "*X -- Create device interface ((X)IF) file.*"
  Additional information about the LON device and individual variables can be read from the interface file.

Literature

1. NEURON Chip Distributed Communications and Control Processors MC143150, MC143120. Motorola Inc. 1994, REV 3.
2. LonWorks Host Application Programmer's Guide, Rev. 2, Echelon corp. 078-0016-01B
3. LonWorks PC LonTalk Adapter User's Guide, Rev. 1.0, Echelon corp. 078-0122-01A
4. OpenLDV Programmer's Guide, Echelon corp. 078-0275-01D.

Changes and modifications

-

Document revisions

- Ver. 1.0 – October 27th, 2000 – Creating of the document.
- Ver. 1.1 - January 25th, 2011 - Updating of the document.
- Ver. 1.2 - October 25th, 2012 - Updating of the document.
- Ver. 1.3 - December 5th, 2019 - Hexadecimal notation of Selector, a note about *nodeutil*.

Related pages:

Communication protocols