FREE Studio
LON Communication Protocol

LonWorks Application Notes

09/2016

Application Notes

9MA10266.00

www.eliwell.com

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Eliwell nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein.
If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Eliwell.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Eliwell software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

TABLE OF CONTENTS

CHAPTER 1. Introduction 9

1.1. Definitions 9

1.2. Requirements 9

1.3. LONWorks Introduction 10

Bus termination and bus type 10

Bus Topology Segment 10

Doubly Terminated Bus Topology Segment 10

Free Topology Segment 11

1.4. LONWorks Communication Module 12

Communication Module EVS LONWorks Overview 12

Node not answering 13

CHAPTER 2. LON Profiles 14

2.1. First Start-up 14

2.2. LON profiles 15

2.2.1. LonWorks® Profiles Overview 15

2.2.2. Profiles Management 16

Add a LON profile to an application 16

Chiller profile example 17

Edit Mode command 17

Creating a custom profile 17

2.2.3. Roof top unit custom profile example 19

List of custom profile variable types 20

2.3. FREE Studio Device 21

List of variable types 22

2.4. Generate the LonMark® XIF file 23

Download XIF file from the Controller to USH-Host 25

CHAPTER 3. LON_IEC Library 26

List of target blocks library functions *26*

Control Group *26*

Init Group *28*

Update Group *31*

List of USB target variables *33*

CHAPTER 4. Appendix - Testing 34

4.1. Website 34

4.1.1. LON Profiles and Standard Program Identifier (SPID) Master List 34

4.1.2. NodeUtil 35

4.2. LonWorks® Interface and Communication Module 37

4.3. Testing the Lon Communication Module with NodeUtil 38

4.3.1. NodeUtil commands 39

Useful NodeUtil commands *41*

4.4. Commissioning 43

SAFETY INFORMATION

Important Information

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to inform of potential hazards or to call attention to information that clarifies or simplifies a procedure.

⚡ The addition of this symbol to a Danger safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

⚠ This is the safety alert symbol. It is used to alert you to potential personal injury hazards.
Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result** in death or serious injury**.**

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result** in death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result** in minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel.
No responsibility is assumed by Eliwell for any consequences arising out of the use of this material.
A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

BEFORE YOU BEGIN

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠ WARNING

UNGUARDED EQUIPMENT

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

START-UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

Failure to follow these instructions can result in injury or equipment damage.

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from the NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop, power outage and restart.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link.
- Observe all accident prevention regulations and local safety guidelines.[1]
- Each implementation of this equipment must be individually and thoroughly tested for proper operation before being placed into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" and to NEMA ICS 7.1 (latest edition), "Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems" or their equivalent governing your particular location.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software approved by Eliwell for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

ABOUT THE BOOK

Document Scope

This document is aimed at designers and developers, requires a knowledge of one or more IEC61131-3 standard programming languages and is designed to provide a first-level overview of the installation, functions and use of **FREE Studio.**

Validity Note

This document is valid for **FREE Studio (v.3.5 or greater)**.

The technical characteristics of the devices described in this manual also appear online.

The characteristics that are presented in this manual should be the same as those characteristics that appear online. In line with our policy of constant improvement, we may revise content over time to improve clarity and accuracy. If you see a difference between the manual and online information, use the online information as your reference.

Related Documents

Title of Documentation	Reference Document Code
FREE Advance Logic Controllers, Hardware User Manual	9MA10252 (EN) 9MA00252 (IT)
FREE Evolution - FREE Panel Logic Controllers, Hardware User Manual	9MA10252 (EN) 9MA00252 (IT)
FREE Studio software, Quick Start	9MA10255 (EN) 9MA00255 (IT)
FREE Studio software, Programming Guide	9MA10256 (EN)

You can download these technical publications and other technical information from our website at:

www.eliwell.com

CHAPTER 1

Introduction

1.1. Definitions

- **AVD/AVC**: FREE Advance
- **EVD/EVC**: FREE Evolution
- **EVS LONWorks**: LON Communication Module
- **target**: AVD/AVC - EVD/EVC controller.

- **LON** stands for Local Operating Network.
- **LonWorks**® communications protocol is an accepted standard (IEEE, ANSI, and ISO standard) for building automation and control networks.

- **IEC/PLC:** IEC/PLC application present in the **target** that implements the physical resources defined by the **LONWorks Profiles**.
- **BIOS:** the part of the **target** firmware where the management of the **LON** protocol is implemented.

1.2. Requirements

The LON functionality is available in **FREE Studio 3.5 (or greater).**
The development environment:

→ for programming, where the LON objects are available, is Application
→ for setting of the BIOS parameters is Device.

The targets that support the LON functionality are:

- **FREE EVS LON Communication Module**

- **FREE Advance** updated to version FW 596.1 or later
- **FREE Evolution EVD** (with display) updated to version FW 423.22 or later
- **FREE Evolution EVC** (without display) updated to version FW 477.22 or later

NOTE. All references shall be connected to EVS LON Communication Module.

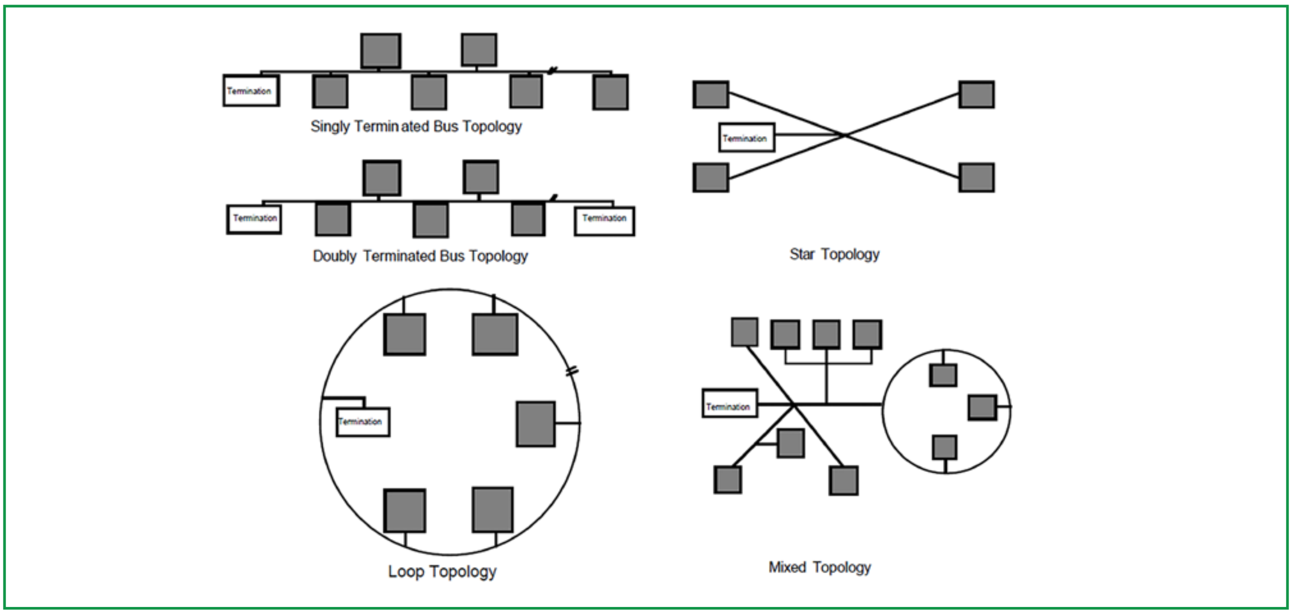
1.3. LONWorks Introduction

Bus termination and bus type

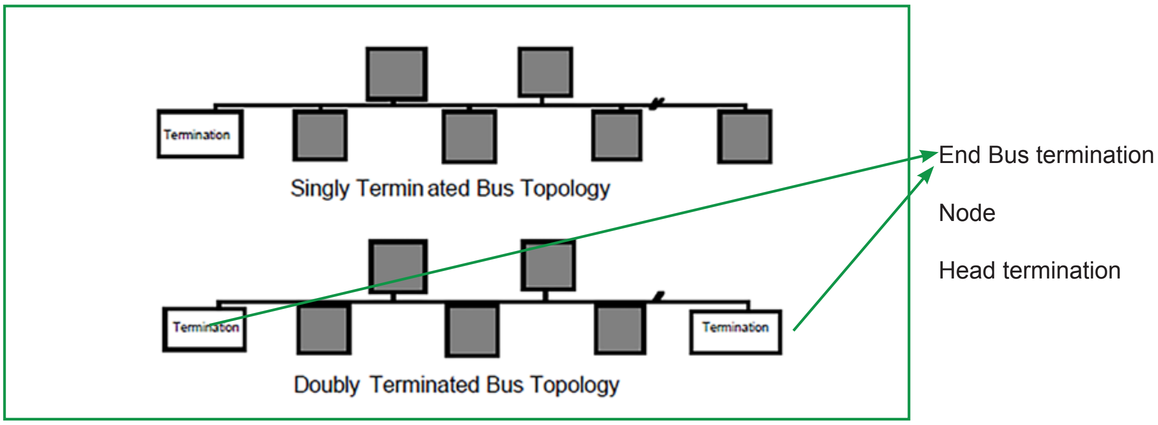
LONWORKS provides support for multiple communications channels.
A channel is a physical transport medium for data telegrams (packets) and can contain up to 32.385 nodes. A network may consist of one or more channels.
Packets are transferred from one channel to another by using routers.

LONWORKS supports bus, loop and star topologies.
The design of the transceiver determines the number of nodes in one channel as well as the maximum transmission distance between nodes in one channel.

Bus Topology Segment

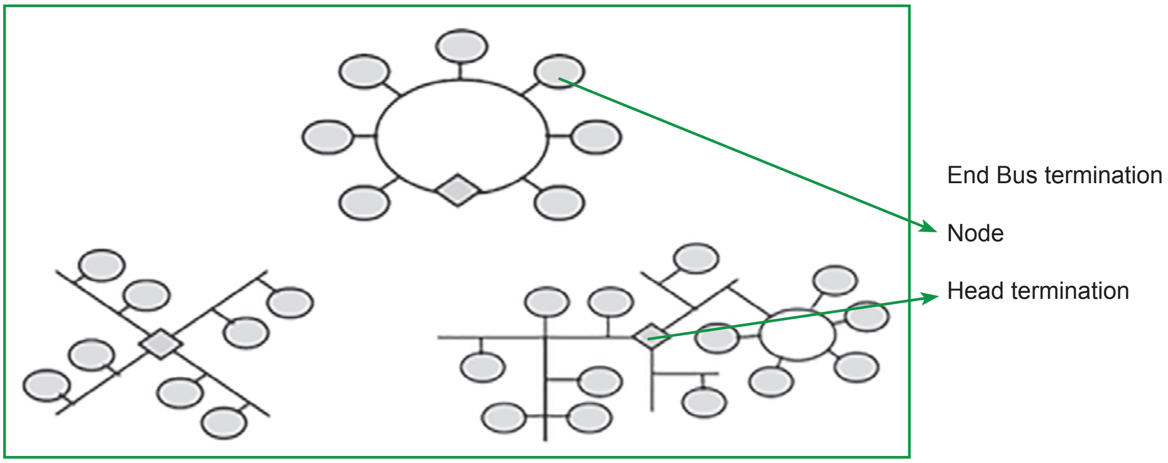


Doubly Terminated Bus Topology Segment



In a doubly terminated bus topology, two terminations are required, one at each end of the bus.
Termination features: RC network with R1 = 105Ω ± 1%, 1/8 W
At this time, only one LPI-10 Interface is supported per segment, which provides one of the two end terminations required. The other terminator must be an RC-type.

Free Topology Segment



In a free topology segment, only one termination is required and may be placed anywhere on the free topology segment. Termination shall be RC-type network with R1 = 52.3Ω ±1%, 1/8 W

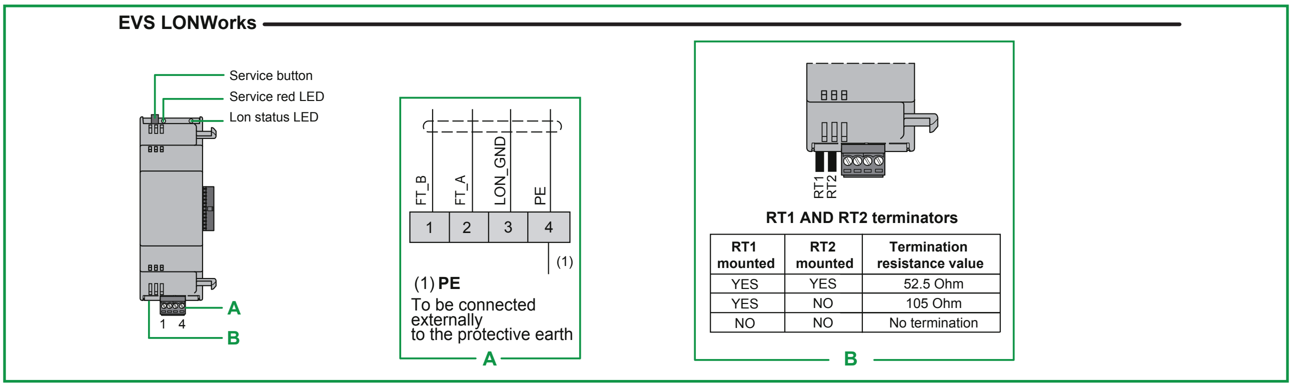
The transceiver type used should meet the physical performance specifications listed below:

Transceiver type	Bit rate	Topology	Nodes per channel	Max length	Neuron isolation
FTT10A	78 kbps	Bus	64 (63+ Master)	2700 m	transformer
FTT10A	78 kbps	Free	64 (63+ Master)	500 m	transformer

1.4. LONWorks Communication Module

A LON Communication Module (EVS0LON000000) is needed to setup the communication.
The Communication Module is powered by the controller.

EVS0LON000000 Communication Module HW description



(1) **PE**
To be connected externally
to the protective earth

RT1 AND RT2 terminators

RT1 mounted	RT2 mounted	Termination resistance value
YES	YES	52.5 Ohm
YES	NO	105 Ohm
NO	NO	No termination

For details about **EVS LON** Communication Modules, refer to the **FREE Evolution** Hardware User Manual.

Communication Module EVS LONWorks Overview

The transceiver type is the FTT-10 with a non-polarized twisted pair cable and a maximum speed of 78 Kb/s.

LED	Description
Status led (GREEN)	• ON: Communication Module configured and online • OFF: Communication Module NOT configured or error present
Service led (RED)	• ON: critical HW failure or applicationless state • Flashing: unconfigured from LonMaker® side or configured but in offline state • OFF: OK
Button	**Description**
Service Button	The service button is used to send a Program ID message

Red LED driven by Echelon hardware:

HW LED (red)	Description
ON	Critical error Application-less (non configured - IEC side)
Flashing	Unconfigured Lon network node not configured
OFF	Node configured, can communicate with the application

Green LED driven by AVD/AVC - EVD/EVC application:

HW LED (green)	Description
ON	Configured online
Flashes	Driven by Echelon hardware: profile NOT active
OFF	Depends on Red LED status.OFF when Red LED is ON or Flashing

NOTE: The 2 communication cables are not polarized.

NOTE: Neuron ID is associated to each EVS LONWorks Communication Module

Node not answering

HW LED (red)	Description
Application-less	Node configured but no application. Node answers only via NodeUtil.
Unconfigured	Application available but variable list not complete.
Init configuration	Hybrid status. Configuration ongoing.
Configured offline	Query on node can be made but NO events generated. Cannot send messages, can only receive messages.
Configured online	Node configured.

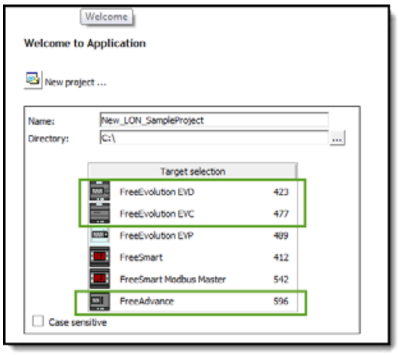
CHAPTER 2

LON Profiles

2.1. First Start-up

1. Click the Launch **FREE Studio** icon located on the Desktop.
2. Click the **FREE Studio Application** icon.

The program appears as in the figure (for example after clicking on **New Project**):



Welcome

Welcome to Application

New project ...

Name: New_LON_SampleProject
Directory: C:\

Target selection	
FreeEvolution EVD	423
FreeEvolution EVC	477
FreeEvolution EVP	489
FreeSmart	412
FreeSmart Modbus Master	542
FreeAdvance	596

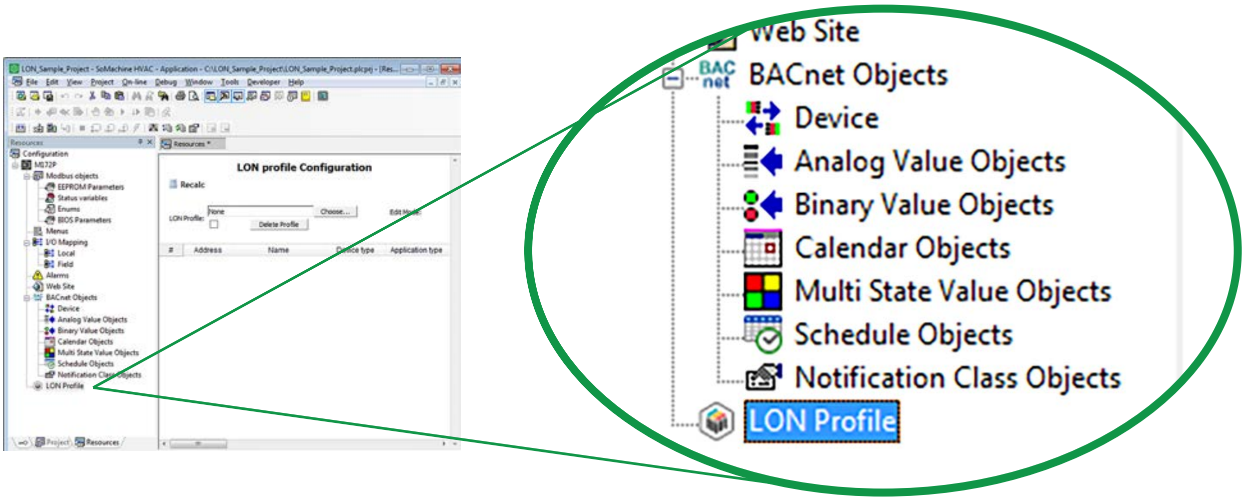
□ Case sensitive

You can now:

- Create a new project '**New Project ...**'
  - in this case choose a **AVD**[1] target
  - the numbers at the side indicate the firmware version (mask) (for example **596**)
- Open an existing project '**Open project …**'
- Open recent projects '**Open an existing project ...**' by selecting them from a list.

The projects keep the **.ppjs** extension, **as in the previous Evolution projects**[2]

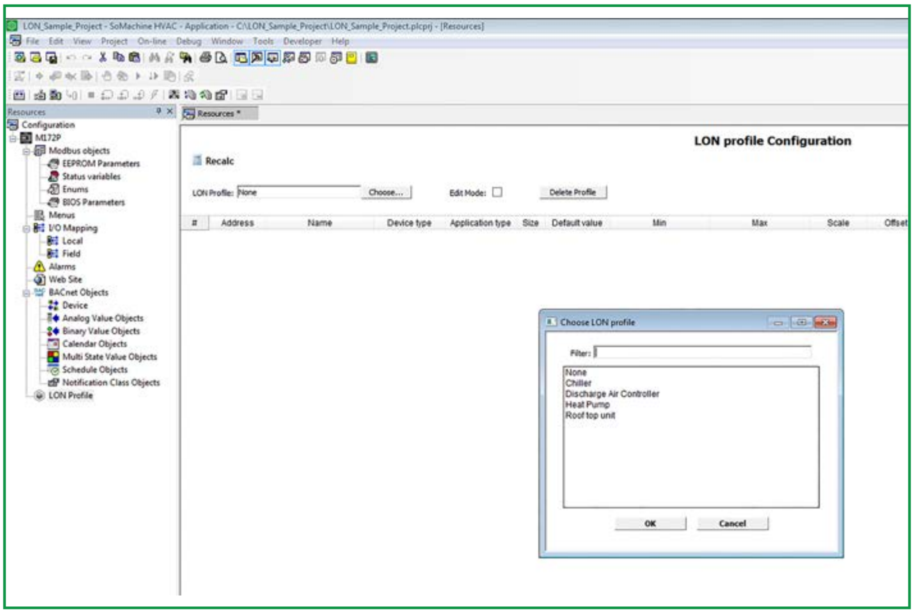
By default, the **LOn Profile configuration** will be listed in the **Resources** pane:



[1] The operation for AVC / EVD / EVC is completely identical
[2] For existing projects see corresponding section

2.2. LON profiles

Application allows you to choose from different HVAC profiles.
Click on the Choose button to select the appropriate one:



LON_Sample_Project - SoMachine HVAC - Application - C:\LON_Sample_Project\LON_Sample_Project.plcprj - [Resources]

LON profile Configuration

LON Profile: None

Edit Mode: □

#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset

Choose LON profile

Filter:

None
Chiller
Discharge Air Controller
Heat Pump
Roof top unit

2.2.1. LonWorks® Profiles Overview

The LonWorks® functional profiles supported in **FREE Studio** are:

Profile	Description	Functional Profile
None	LON protocol disabled. Not used	-
Roof top unit	**Roof Top Unit (RTU)** The RTU object can control heat, cool, economizer and fan outputs directly through hardware outputs.	8030
Chiller	**Chiller** The Chiller object for a generic Chiller controller object. Basically created around a model of a centrifugal Chiller.	8040
Heat Pump	**Heat Pump with Temperature Control** The heat pump with temperature control object provides the control of a heat pump unit. This object compares room temperature with heating and cooling set-points.	8051
Discharge Air Controller	**Discharge-Air Controller** Air controller is used to control duct static pressure and discharge air temperature, along with other auxiliary functions such as humidity and/or ventilation control.	8610

Profiles are available at
www.lonmark.org/profiles/

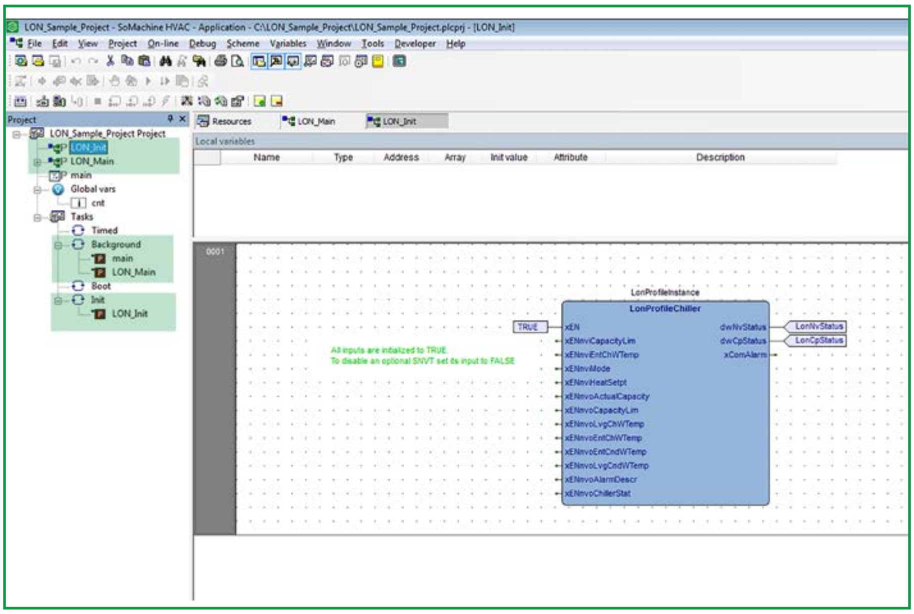
Contact an Eliwell support representative to check the available profiles.

2.2.2. Profiles Management

Add a LON profile to an application

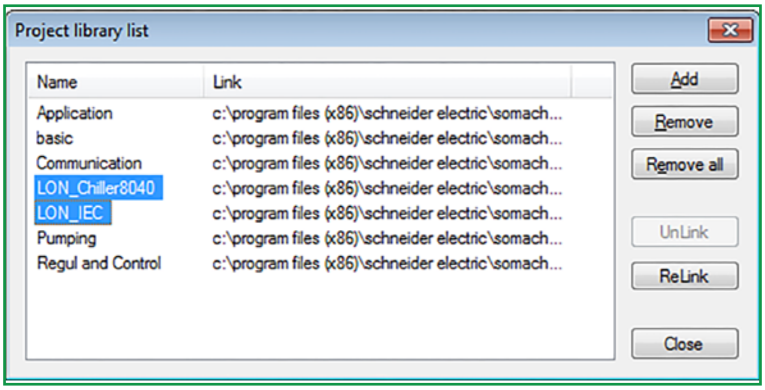
Once a LON profile has been added to your application, the follow programs will be automatically generated in the Project tab:

- LON_Init in Init task
- LON_Main in Background task



All inputs are initialized to TRUE.
To disable an optional SNVT set its input to FALSE

In addition, the LON_IEC and LON_Chiller 8040 libraries will be linked to the project:



Name	Link
Application	c:\program files (x86)\schneider electric\somach...
basic	c:\program files (x86)\schneider electric\somach...
Communication	c:\program files (x86)\schneider electric\somach...
LON_Chiller8040	c:\program files (x86)\schneider electric\somach...
LON_IEC	c:\program files (x86)\schneider electric\somach...
Pumping	c:\program files (x86)\schneider electric\somach...
Regul and Control	c:\program files (x86)\schneider electric\somach...

These programs are the interface between the LON protocol and the IEC Application variables.

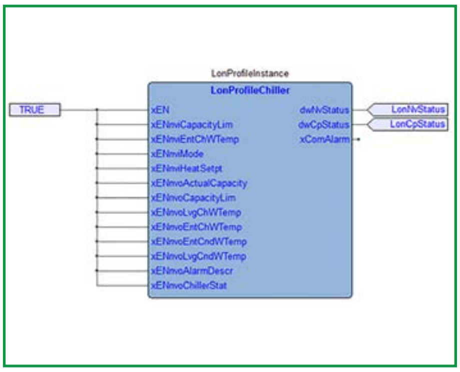
The developer should link the relevant variables including U.M.

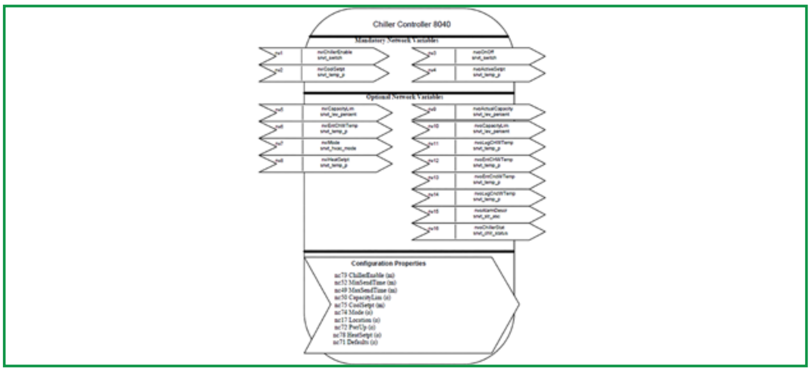
NOTE. Variables must be congruent with the LON profile specification selected.

Refer to the PDF file avilalable at www.lonmark.org/profiles/

- LON_IEC includes basic variables.
- LON Chiller includes mandatory and optional I/O interface variables.

All global variables are defined in the profile library.
Profile variables shall be instanced: the function block sets the global variables from the LON stack.
The function block will define which vvariables are optional or not and refresh them in both directions.



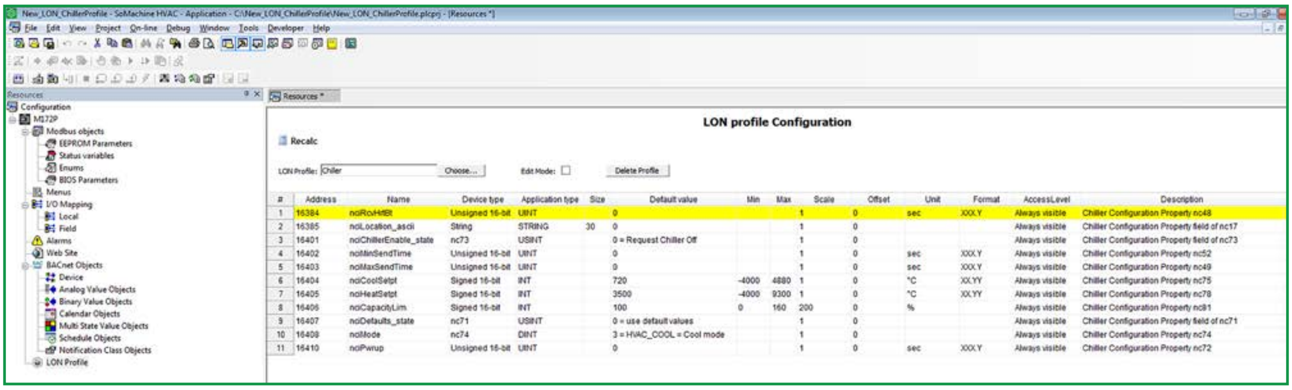


Profile Instance

Profile Specification

Chiller profile example

In the **Resources** area, choose a Chiller profile by clicking on the **Choose** button.
The Chiller Profile will be uploaded from the Library. Chiller configuration properties are available by default:



LON profile Configuration

LON Profile: Chiller

Edit Mode: □

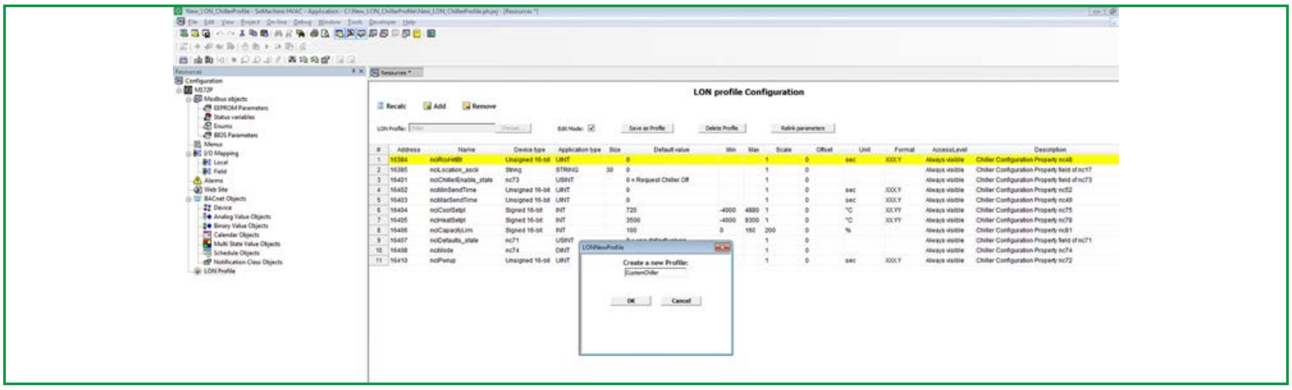
#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Description
1	16384	nciRcvHrtBt	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc48
2	16385	nciLocation_ascii	String	STRING	30	0			1	0			Always visible	Chiller Configuration Property field of nc17
3	16401	nciChillerEnable_state	nc73	USINT		0 = Request Chiller Off			1	0			Always visible	Chiller Configuration Property field of nc73
4	16402	nciMinSendTime	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc52
5	16403	nciMaxSendTime	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc49
6	16404	nciCoolSetpt	Signed 16-bit	INT		720	-4000	4880	1	0	°C	XX.YY	Always visible	Chiller Configuration Property nc75
7	16405	nciHeatSetpt	Signed 16-bit	INT		3500	-4000	9300	1	0	°C	XX.YY	Always visible	Chiller Configuration Property nc78
8	16406	nciCapacityLim	Signed 16-bit	INT		100	0	160	200	0	%		Always visible	Chiller Configuration Property nc81
9	16407	nciDefaults_state	nc71	USINT		0 = use default values			1	0			Always visible	Chiller Configuration Property field of nc71
10	16408	nciMode	nc74	DINT		3 = HVAC_COOL = Cool mode			1	0			Always visible	Chiller Configuration Property nc74
11	16410	nciPwrup	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc72

Edit Mode command

Edit Mode	Description	Notes
Save as Profile	Create a new custom profile.	ChillerCustom
Delete Profile	Deletes custom profiles. If no custom profile have been created, the list of available profile to delete is empty.	ChillerCustom
Relink parameters	Redirects variables. Custom profile only.	ChillerCustom

Creating a custom profile

In the **Resources** area, enable **Edit Mode** and click on **Save as Profile**:


New_LON_ChillerProfile - SoMachine HVAC - Application - C:\New_LON_ChillerProfile\New_LON_ChillerProfile.plcprj - [Resources *]
Resources *
Recalc
Add
Remove
Save as Profile
Delete Profile
Relink parameters
LONNewProfile
OK
Cancel

LON profile Configuration

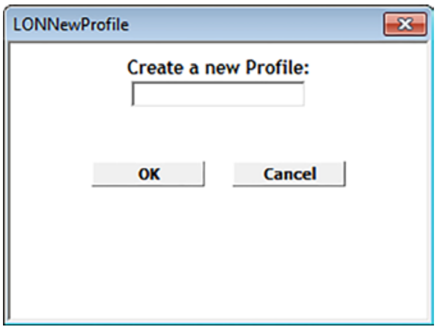
LON Profile: Chiller

Edit Mode: ☑

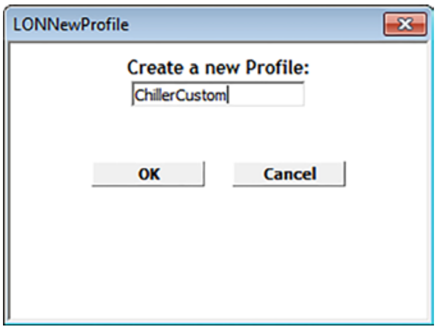
Create a new Profile: CustomChiller

#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Description
1	16384	nciRcvHrtBt	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc48
2	16385	nciLocation_ascii	String	STRING	30	0			1	0			Always visible	Chiller Configuration Property field of nc17
3	16401	nciChillerEnable_state	nc73	USINT		0 = Request Chiller Off			1	0			Always visible	Chiller Configuration Property field of nc73
4	16402	nciMinSendTime	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc52
5	16403	nciMaxSendTime	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc49
6	16404	nciCoolSetpt	Signed 16-bit	INT		720	-4000	4880	1	0	°C	XX.YY	Always visible	Chiller Configuration Property nc75
7	16405	nciHeatSetpt	Signed 16-bit	INT		3500	-4000	9300	1	0	°C	XX.YY	Always visible	Chiller Configuration Property nc78
8	16406	nciCapacityLim	Signed 16-bit	INT		100	0	160	200	0	%		Always visible	Chiller Configuration Property nc81
9	16407	nciDefaults_state	nc71	USINT		0 = use default values			1	0			Always visible	Chiller Configuration Property field of nc71
10	16408	nciMode	nc74	DINT					1	0			Always visible	Chiller Configuration Property nc74
11	16410	nciPwrup	Unsigned 16-bit	UINT					1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc72

Choose a Profile name (for example ChillerCustom):

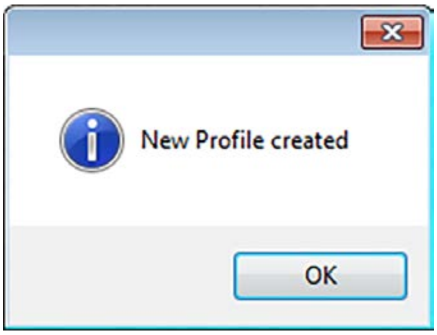


Save as Profile



ChillerCustom

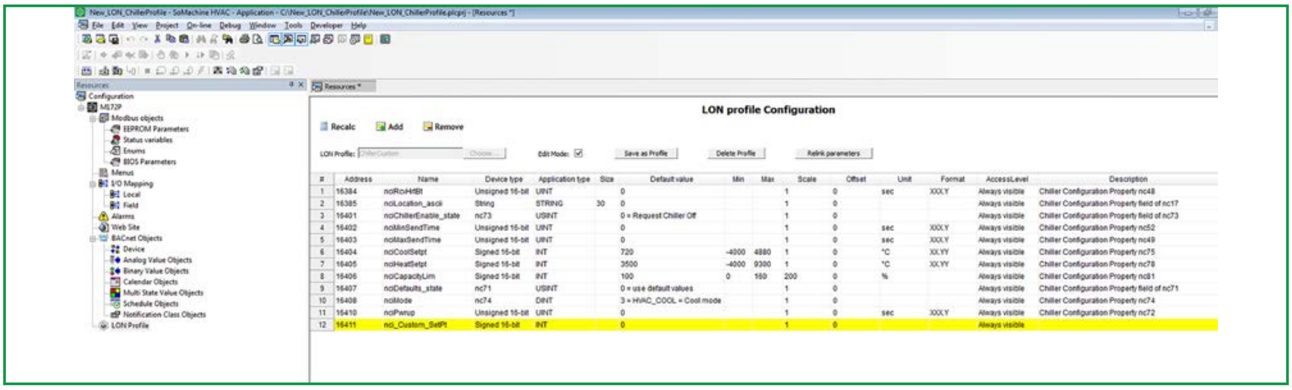
New Profile created



New Profile created

Create a new Profile: ChillerCustom

Once the Custom Profile has been created, you may add or remove properties in Edit Mode:



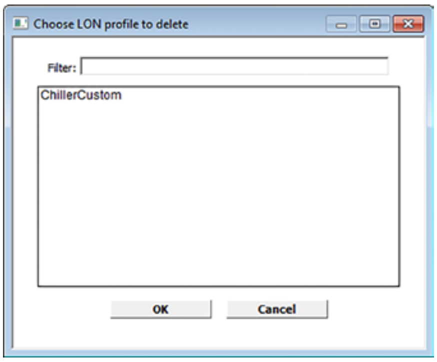
LON profile Configuration

LON Profile: ChillerCustom

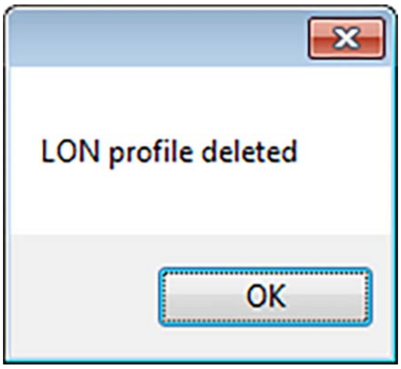
Edit Mode: ☑

#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Description
1	16384	nciRcvHrtBt	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc48
2	16385	nciLocation_ascii	String	STRING	30	0			1	0			Always visible	Chiller Configuration Property field of nc17
3	16401	nciChillerEnable_state	nc73	USINT		0 = Request Chiller Off			1	0			Always visible	Chiller Configuration Property field of nc73
4	16402	nciMinSendTime	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc52
5	16403	nciMaxSendTime	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc49
6	16404	nciCoolSetpt	Signed 16-bit	INT		720	-4000	4880	1	0	°C	XX.YY	Always visible	Chiller Configuration Property nc75
7	16405	nciHeatSetpt	Signed 16-bit	INT		3500	-4000	9300	1	0	°C	XX.YY	Always visible	Chiller Configuration Property nc78
8	16406	nciCapacityLim	Signed 16-bit	INT		100	0	150	200	0	%		Always visible	Chiller Configuration Property nc81
9	16407	nciDefaults_state	nc71	USINT		0 = use default values			1	0			Always visible	Chiller Configuration Property field of nc71
10	16408	nciMode	nc74	DINT		3 = HVAC_COOL = Cool mode			1	0			Always visible	Chiller Configuration Property nc74
11	16410	nciPwrup	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Chiller Configuration Property nc72
12	16411	nci_Custom_SetPt	Signed 16-bit	INT		0			1	0			Always visible	

NOTE. Changes made to the custom parameter list will be also handled in the corresponding .pll custom library. The Custom Profile can be also deleted:



ChillerCustom profile to delete



LON profile deleted

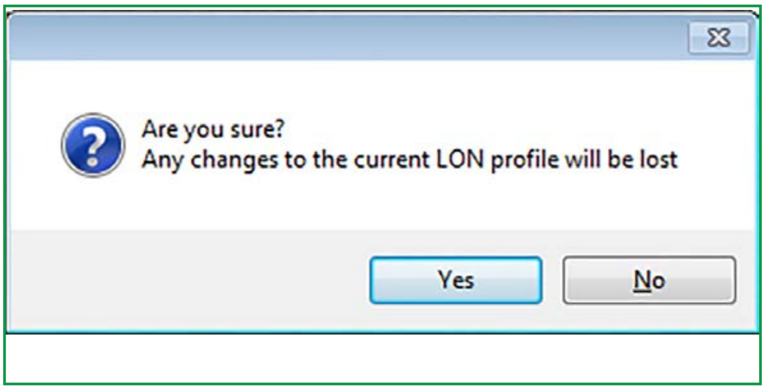
Profile deleted

NOTE. Only a custom profile can be deleted.

2.2.3. Roof top unit custom profile example

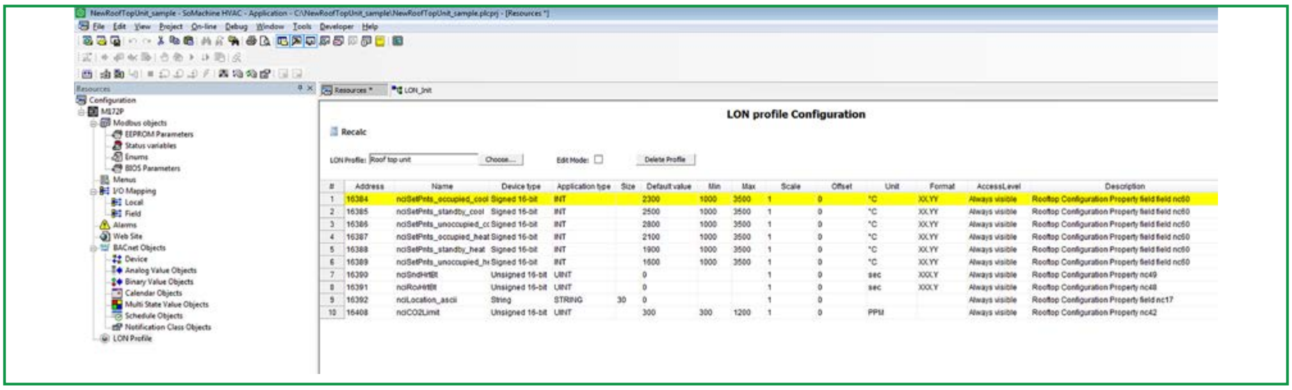
Following the previous steps, the best way to create a custom profile is to use an existing profile as a template. For example, let's choose a Roof top profile.

1. Click on the Choose button to select the Roof top unit profile
2. If you were working on a previous profile, a dialogue message appears to confirm whether you wish to proceed without saving the profile. Click Yes.



Are you sure?
Any changes to the current LON profile will be lost

3. A Roof top unit profile is created:



NewRoofTopUnit_sample - SoMachine HVAC - Application - C:\NewRoofTopUnit_sample\NewRoofTopUnit_sample.plcprj - [Resources *]

LON profile Configuration

LON Profile: Roof top unit

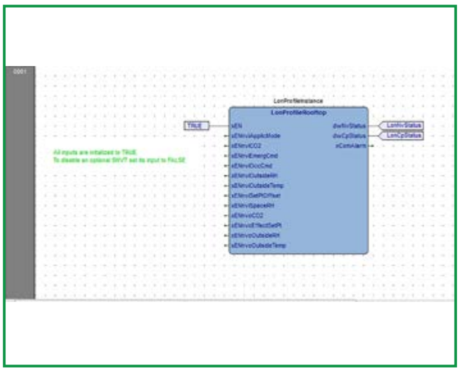
Edit Mode: □

#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Description
1	16384	nciSetPnts_occupied_cool	Signed 16-bit	INT		2300	1000	3500	1	0	°C	XX.YY	Always visible	Rooftop Configuration Property field field nc60
2	16385	nciSetPnts_standby_cool	Signed 16-bit	INT		2500	1000	3500	1	0	°C	XX.YY	Always visible	Rooftop Configuration Property field field nc60
3	16386	nciSetPnts_unoccupied_cc	Signed 16-bit	INT		2800	1000	3500	1	0	°C	XX.YY	Always visible	Rooftop Configuration Property field field nc60
4	16387	nciSetPnts_occupied_heat	Signed 16-bit	INT		2100	1000	3500	1	0	°C	XX.YY	Always visible	Rooftop Configuration Property field field nc60
5	16388	nciSetPnts_standby_heat	Signed 16-bit	INT		1900	1000	3500	1	0	°C	XX.YY	Always visible	Rooftop Configuration Property field field nc60
6	16389	nciSetPnts_unoccupied_he	Signed 16-bit	INT		1600	1000	3500	1	0	°C	XX.YY	Always visible	Rooftop Configuration Property field field nc60
7	16390	nciSndHrtBt	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Rooftop Configuration Property nc49
8	16391	nciRcvHrtBt	Unsigned 16-bit	UINT		0			1	0	sec	XXX.Y	Always visible	Rooftop Configuration Property nc48
9	16392	nciLocation_ascii	String	STRING	30	0			1	0			Always visible	Rooftop Configuration Property field nc17
10	16408	nciCO2Limit	Unsigned 16-bit	UINT		300	300	1200	1	0	PPM		Always visible	Rooftop Configuration Property nc42

- A Roof top unit function block is available in the Project pane.
- The Function block is also available in the Resources pane, LONRTU8030 tab.
- The Roof top unit vars are available in the Resources pane, LONRTU8030 vars tab.

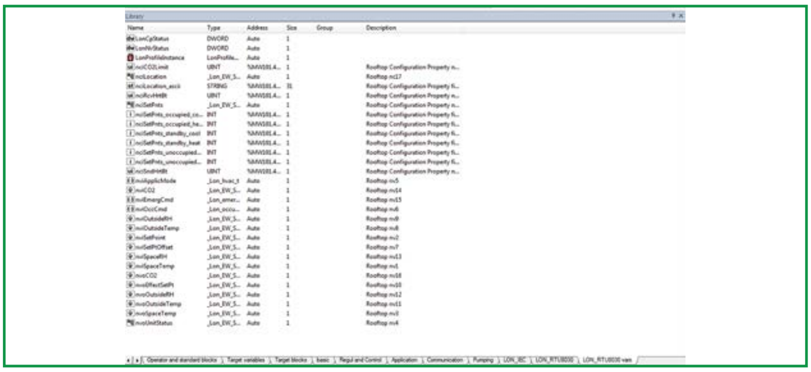
Import the existing FB and variable from the profile LON_RTU8030.PLL.

Rename the FB (for example **LonCustomProfileRooftop**) and the relative instance.



LonProfileInstance
LonProfileRooftop
TRUE
LonNvStatus
LonCpStatus
All inputs are initialized to TRUE
To disable an optional SNVT set its input to FALSE

Roof top unit Function Block



Library

Name	Type	Address	Size	Group	Description
LonCpStatus	DWORD	Auto	1		
LonNvStatus	DWORD	Auto	1		
LonProfileInstance	LonProfile...	Auto	1		
nciCO2Limit	UINT	%MW181.4...	1		Rooftop Configuration Property n...
nciLocation	_Lon_EW_S...	Auto	1		Rooftop nc17
nciLocation_ascii	STRING	%MW181.4...	31		Rooftop Configuration Property fi...
nciRcvHrtBt	UINT	%MW181.4...	1		Rooftop Configuration Property n...
nciSetPnts	_Lon_EW_S...	Auto	1		Rooftop Configuration Property n...
nciSetPnts_occupied_co...	INT	%MW181.4...	1		Rooftop Configuration Property fi...
nciSetPnts_occupied_he...	INT	%MW181.4...	1		Rooftop Configuration Property fi...
nciSetPnts_standby_cool	INT	%MW181.4...	1		Rooftop Configuration Property fi...
nciSetPnts_standby_heat	INT	%MW181.4...	1		Rooftop Configuration Property fi...
nciSetPnts_unoccupied...	INT	%MW181.4...	1		Rooftop Configuration Property fi...
nciSetPnts_unoccupied...	INT	%MW181.4...	1		Rooftop Configuration Property fi...
nciSndHrtBt	UINT	%MW181.4...	1		Rooftop Configuration Property n...
nviApplicMode	_Lon_hvac_t	Auto	1		Rooftop nv5
nviCO2	_Lon_EW_S...	Auto	1		Rooftop nv14
nviEmergCmd	_Lon_emer...	Auto	1		Rooftop nv15
nviOccCmd	_Lon_occu...	Auto	1		Rooftop nv6
nviOutsideRH	_Lon_EW_S...	Auto	1		Rooftop nv9
nviOutsideTemp	_Lon_EW_S...	Auto	1		Rooftop nv8
nviSetPoint	_Lon_EW_S...	Auto	1		Rooftop nv2
nviSetPtOffset	_Lon_EW_S...	Auto	1		Rooftop nv7
nviSpaceRH	_Lon_EW_S...	Auto	1		Rooftop nv13
nviSpaceTemp	_Lon_EW_S...	Auto	1		Rooftop nv1
nvoCO2	_Lon_EW_S...	Auto	1		Rooftop nv18
nvoEffectSetPt	_Lon_EW_S...	Auto	1		Rooftop nv10
nvoOutsideRH	_Lon_EW_S...	Auto	1		Rooftop nv12
nvoOutsideTemp	_Lon_EW_S...	Auto	1		Rooftop nv11
nvoSpaceTemp	_Lon_EW_S...	Auto	1		Rooftop nv3
nvoUnitStatus	_Lon_EW_S...	Auto	1		Rooftop nv4

Roof top unit vars

4. Enable Edit mode and rename the profile. An empty pll library will be created and linked to the project.
5. To enable and disable the profile variable, set the Boolean value to the FB input.

The instance of the LonProfile Function Block is automatically defined in the library LON_RTU8030vars. This variable is automatically defined when a profile is used.

The application can reference directly to the NVI, NVO and NCI profile variables.

List of custom profile variable types

SNVT is the name assigned to LON object properties.
Available SNVT for the custom profile are:

- SNVT_area
- SNVT_chlr_status
- SNVT_count
- SNVT_count_32
- SNVT_entalphy
- SNVT_flow
- SNVT_hvac_overid
- SNVT_hvac_status
- SNVT_INFO
- SNVT_lev_percent
- SNVT_multiplier
- SNVT_power
- SNVT_power_kilo
- SNVT_ppm
- SNVT_press_p
- SNVT_str_asc
- SNVT_switch
- SNVT_temp_diff_p
- SNVT_temp_p
- SNVT_temp_setpt
- SNVT_time_min
- SNVT_time_sec
- SNVT_tod_event

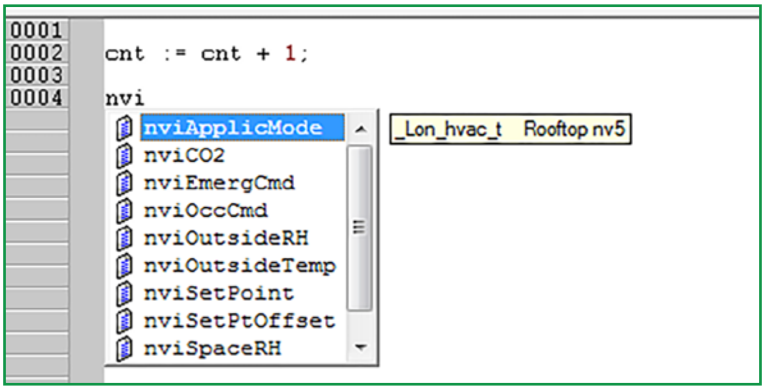
NOTE. The scale and U.M. for SNVT variables are available in the SNVT Master List file at:
https://www.lonmark.org/technical_resources/resource_files/snvt.pdf

NOTE. Maximum numer of SNVT is 255.

NOTE. Variable managment is enabled through boolean variable.
For example: temperature probe is not enabled because I am using my own probe.
If it is mandatory then I cannot enable it; if it is optional then look at the function block.

NOTE: To reference to a profile variable in the Structured Text program it is recommended to :
1. Type **nvi**, **nvo** or **nci**
2. Press CTRL+SPACE.

A drop down menu will show you all the available profile variables:

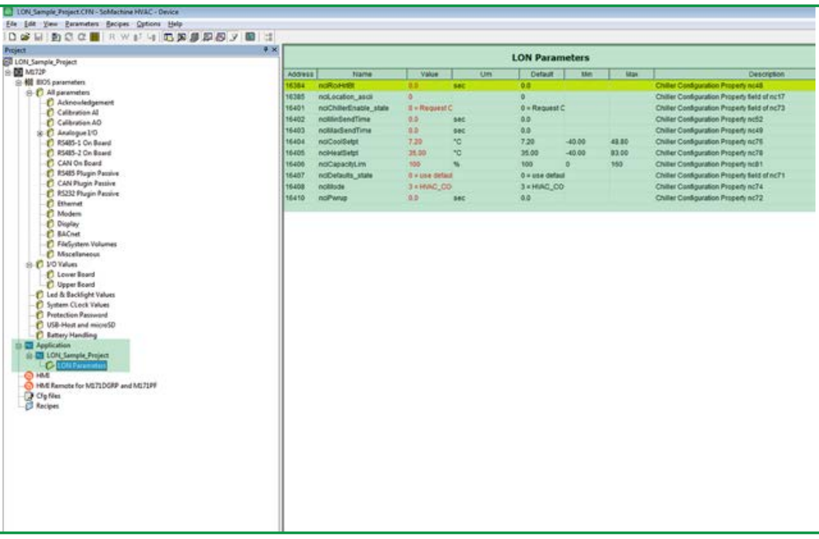


```

cnt := cnt + 1;

nvi
```

2.3. FREE Studio Device

Use Device to view the LonWorks® Configuration Properties.

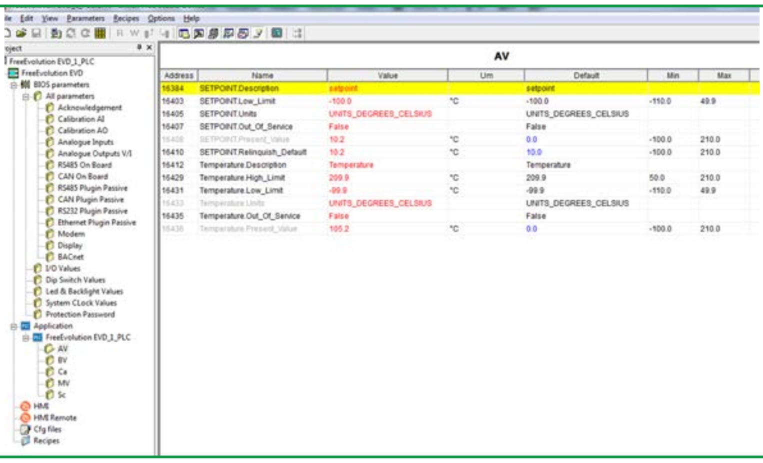
As well as the **Application** environment, the **Device** working environment is used, which contains the parameters of the PLC application and, amongst **Application parameters**, a dedicated **LON** parameter folder:



LON Parameters

Address	Name	Value	Um	Default	Min	Max	Description
16384	nciRcvHrtBt	0.0	sec	0.0			Chiller Configuration Property nc48
16385	nciLocation_ascii	0		0			Chiller Configuration Property field of nc17
16401	nciChillerEnable_state	0 = Request C		0 = Request C			Chiller Configuration Property field of nc73
16402	nciMinSendTime	0.0	sec	0.0			Chiller Configuration Property nc52
16403	nciMaxSendTime	0.0	sec	0.0			Chiller Configuration Property nc49
16404	nciCoolSetpt	7.20	°C	7.20	-40.00	48.80	Chiller Configuration Property nc75
16405	nciHeatSetpt	35.00	°C	35.00	-40.00	83.00	Chiller Configuration Property nc78
16406	nciCapacityLim	100	%	100	0	150	Chiller Configuration Property nc81
16407	nciDefaults_state	0 = use defaul		0 = use defaul			Chiller Configuration Property field of nc71
16408	nciMode	3 = HVAC_CO		3 = HVAC_CO			Chiller Configuration Property nc74
16410	nciPwrup	0.0	sec	0.0			Chiller Configuration Property nc72

The attributes will be displayed in **FREE Studio Device** and will include the Modbus address (**Address** column). Attributes set as **read only (in grey)** will not be editable.



AV

Address	Name	Value	Um	Default	Min	Max
16384	SETPOINT.Description	setpoint		setpoint		
16403	SETPOINT.Low_Limit	-100.0	°C	-100.0	-110.0	49.9
16405	SETPOINT.Units	UNITS_DEGREES_CELSIUS		UNITS_DEGREES_CELSIUS		
16407	SETPOINT.Out_Of_Service	False		False		
16408	SETPOINT.Present_Value	10.2	°C	0.0	-100.0	210.0
16410	SETPOINT.Relinquish_Default	10.2	°C	10.0	-100.0	210.0
16412	Temperature.Description	Temperature		Temperature		
16429	Temperature.High_Limit	209.9	°C	209.9	50.0	210.0
16431	Temperature.Low_Limit	-99.9	°C	-99.9	-110.0	49.9
16433	Temperature.Units	UNITS_DEGREES_CELSIUS		UNITS_DEGREES_CELSIUS		
16435	Temperature.Out_Of_Service	False		False		
16436	Temperature.Present_Value	105.2	°C	0.0	-100.0	210.0

FREE Studio allows the developer to:

- select and configure **LON** as an optional communication module.
- map variables to the **LON** network variables.
- link controller internal variables to **LON** network variables.

List of variable types

The following standard network variable types are supported:

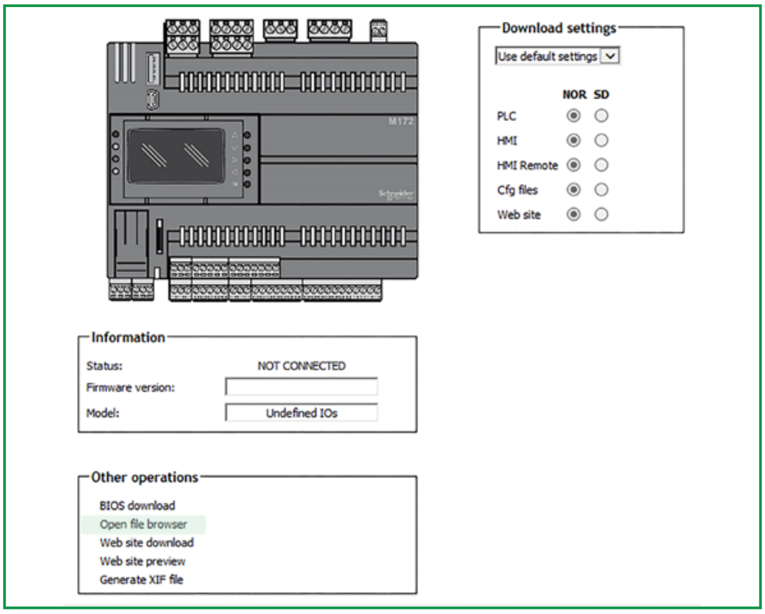
- <name> = <index>
- SNVT_COUNT = 8
- SNVT_FLOW = 15
- SNVT_POWER = 27
- SNVT_POWER_KILO = 28
- SNVT_PPM = 29
- SNVT_STR_ASC = 36
- SNVT_LEV_PERCENT = 81
- SNVT_SWITCH = 95
- SNVT_HVAC_EMERG = 103
- SNVT_TEMP_P = 105
- SNVT_TEMP_SETPT = 106
- SNVT_TIME_SEC = 107
- SNVT_HVAC_MODE = 108
- SNVT_SNVT_OCCUPANCY = 109
- SNVT_HVAC_OVERID = 111
- SNVT_HVAC_STATUS = 112
- SNVT_PRESS_P = 113
- SNVT_TIME_MIN = 123
- SNVT_CHLR_STATUS = 127
- SNVT_ENTHALPY = 153
- SNVT_COUNT_32 = 183

NOTE. No more variable types can be added, as they are not supported by LON stack.

2.4. Generate the LonMark® XIF file

To generate the XIF file:

1. Device command 222 and save it on a USB stick.
2. Use Nodeutil - See Appendix.
3. Connect with Application and view the sysUsbCommands and sysUsbStatus target variable.



Download settings

Use default settings

	NOR	SD
PLC	●	○
HMI	●	○
HMI Remote	●	○
Cfg files	●	○
Web site	●	○

Information

Status: NOT CONNECTED

Firmware version:

Model: Undefined IOs

Other operations

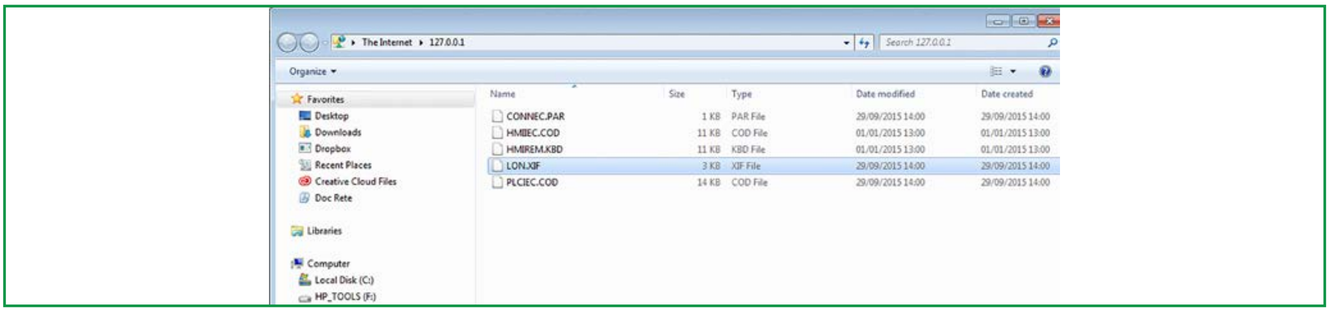
BIOS download

Open file browser

Web site download

Web site preview

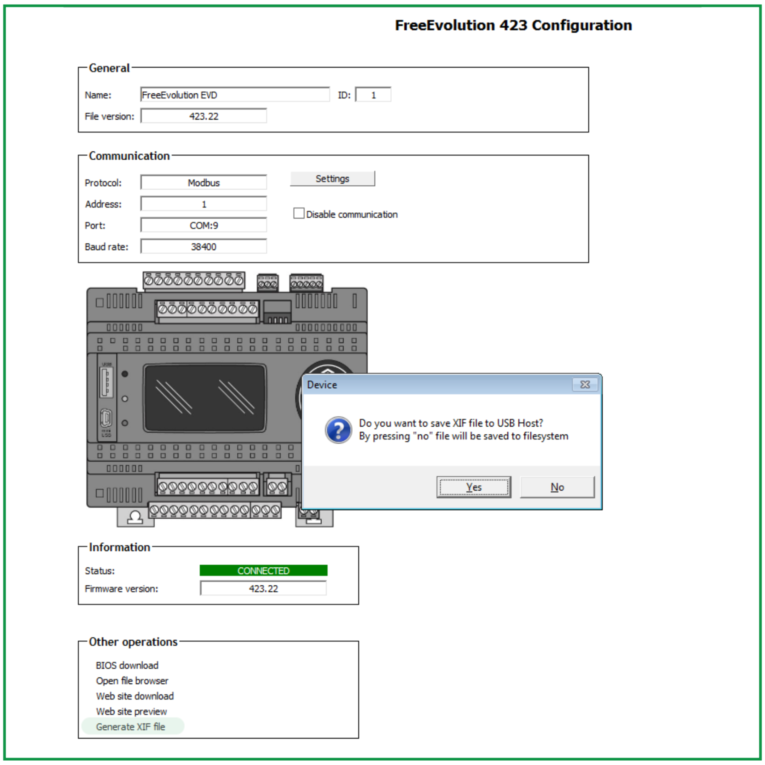
Generate XIF file



Name	Size	Type	Date modified	Date created
CONNEC.PAR	1 KB	PAR File	29/09/2015 14:00	29/09/2015 14:00
HMIIEC.COD	11 KB	COD File	01/01/2015 13:00	01/01/2015 13:00
HMIREM.KBD	11 KB	KBD File	01/01/2015 13:00	01/01/2015 13:00
LON.XIF	3 KB	XIF File	29/09/2015 14:00	29/09/2015 14:00
PLCIEC.COD	14 KB	COD File	29/09/2015 14:00	29/09/2015 14:00

You can now download the XIF file from Windows Explorer.

Generate XIF file from Device:



FreeEvolution 423 Configuration

General

Name: FreeEvolution EVD ID: 1

File version: 423.22

Communication

Protocol: Modbus

Address: 1

Port: COM:9

Baud rate: 38400

Settings

□ Disable communication

Device

Do you want to save XIF file to USB Host?
By pressing "no" file will be saved to filesystem

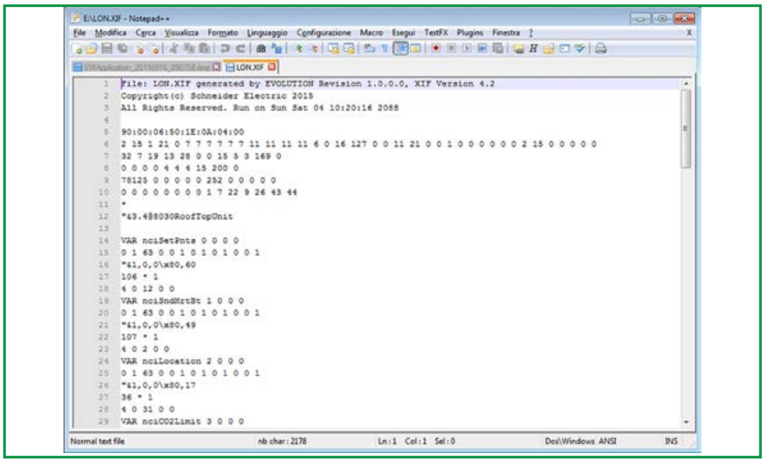
Information

Status: CONNECTED

Firmware version: 423.22

Other operations

NOTE: XIF files generated with the FREE Studio Device and NodeUtil are equivalent.



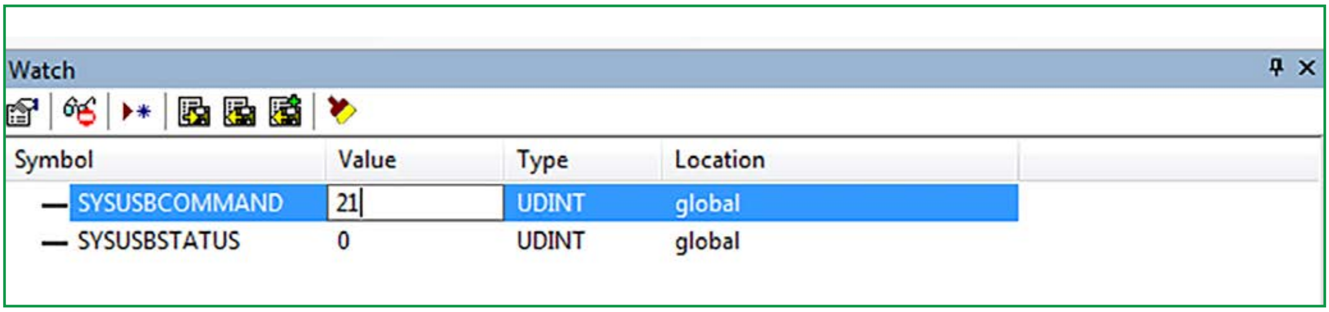
E:\LON.XIF - Notepad++
File Modifica Cerca Visualizza Formato Linguaggio Configurazione Macro Esegui TextFX Plugins Finestra ?
LON.XIF

```
File: LON.XIF generated by EVOLUTION Revision 1.0.0.0, XIF Version 4.2
Copyright(c) Schneider Electric 2015
All Rights Reserved. Run on Sun Sat 04 10:20:16 2088

90:00:06:50:1E:0A:04:00
2 15 1 21 0 7 7 7 7 7 7 11 11 11 11 6 0 16 127 0 0 11 21 0 0 1 0 0 0 0 0 0 2 15 0 0 0 0 0
32 7 19 13 28 0 0 15 5 3 169 0
0 0 0 0 4 4 4 15 200 0
78125 0 0 0 0 0 252 0 0 0 0 0
0 0 0 0 0 0 0 0 1 7 22 9 26 43 44
*
"&3.4@8030RoofTopUnit

VAR nciSetPnts 0 0 0 0
0 1 63 0 0 1 0 1 0 1 0 0 1
"&1,0,0\x80,60
106 * 1
4 0 12 0 0
VAR nciSndHrtBt 1 0 0 0
0 1 63 0 0 1 0 1 0 1 0 0 1
"&1,0,0\x80,49
107 * 1
4 0 2 0 0
VAR nciLocation 2 0 0 0
0 1 63 0 0 1 0 1 0 1 0 0 1
"&1,0,0\x80,17
36 * 1
4 0 31 0 0
VAR nciCO2Limit 3 0 0 0
```

Normal text file nb char : 2178 Ln : 1 Col : 1 Sel : 0 Dos\Windows ANSI INS

Download XIF file from the Controller to USH-Host

```
sysUsbCommand:
      21    =     save LON.XIF file to USB-Host
```



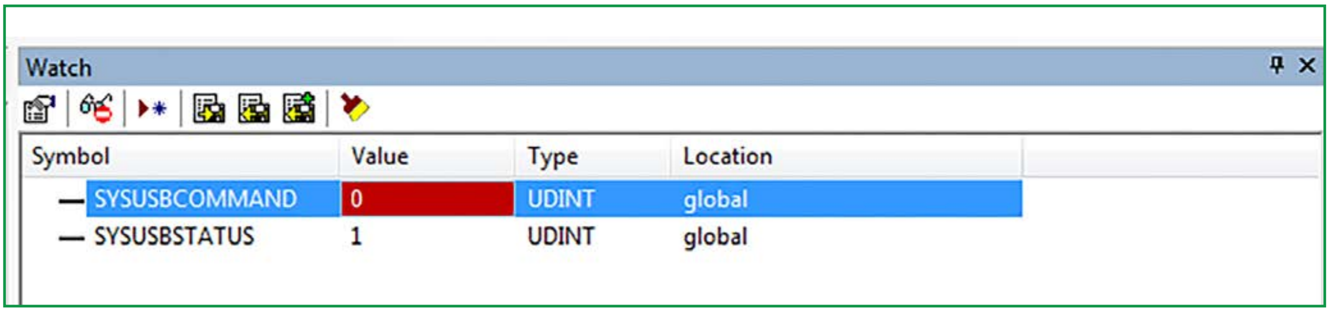
Watch

Symbol	Value	Type	Location
SYSUSBCOMMAND	21	UDINT	global
SYSUSBSTATUS	0	UDINT	global

Below is a list of the possible values returned by sysUsbStatus (highlighted in bold for this situation):

sysUsbStatus:

0	=	command completed
1	**=**	**command processing**
255	=	command failed
254	=	file not present
253	=	file too long
252	=	USBH not connected
251	=	file not compatible
250	=	some parameters fail
249	=	write file failed
248	=	open file in write failed



Watch

Symbol	Value	Type	Location
SYSUSBCOMMAND	0	UDINT	global
SYSUSBSTATUS	1	UDINT	global

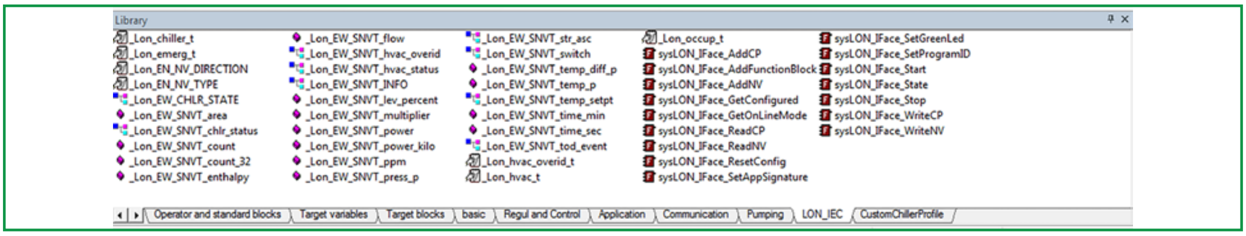
The file produced (if the LONWORKs application is running into the Controller) will be saved as LON.XIF. This file can also be saved in the NOR volume.

CHAPTER 3

LON_IEC Library

List of target blocks library functions

This list includes function block utilities related to LON management



_Lon_chiller_t
_Lon_emerg_t
_Lon_EN_NV_DIRECTION
_Lon_EN_NV_TYPE
_Lon_EW_CHLR_STATE
_Lon_EW_SNVT_area
_Lon_EW_SNVT_chlr_status
_Lon_EW_SNVT_count
_Lon_EW_SNVT_count_32
_Lon_EW_SNVT_enthalpy
_Lon_EW_SNVT_flow
_Lon_EW_SNVT_hvac_overid
_Lon_EW_SNVT_hvac_status
_Lon_EW_SNVT_INFO
_Lon_EW_SNVT_lev_percent
_Lon_EW_SNVT_multiplier
_Lon_EW_SNVT_power
_Lon_EW_SNVT_power_kilo
_Lon_EW_SNVT_ppm
_Lon_EW_SNVT_press_p
_Lon_EW_SNVT_str_asc
_Lon_EW_SNVT_switch
_Lon_EW_SNVT_temp_diff_p
_Lon_EW_SNVT_temp_p
_Lon_EW_SNVT_temp_setpt
_Lon_EW_SNVT_time_min
_Lon_EW_SNVT_time_sec
_Lon_EW_SNVT_tod_event
_Lon_hvac_overid_t
_Lon_hvac_t
_Lon_occup_t
sysLON_IFace_AddCP
sysLON_IFace_AddFunctionBlock
sysLON_IFace_AddNV
sysLON_IFace_GetConfigured
sysLON_IFace_GetOnLineMode
sysLON_IFace_ReadCP
sysLON_IFace_ReadNV
sysLON_IFace_ResetConfig
sysLON_IFace_SetAppSignature
sysLON_IFace_SetGreenLed
sysLON_IFace_SetProgramID
sysLON_IFace_Start
sysLON_IFace_State
sysLON_IFace_Stop
sysLON_IFace_WriteCP
sysLON_IFace_WriteNV

Control Group

sysLON_IFace_GetConfigured

Type: Function
Return value: BOOL
Language type:
Description: configured state of return node.
The function returnss a BOOL:
FALSE = Unconfigured.
TRUE = Configured.

Input:

Name	Type	Description
dummy	BOOL	fix at TRUE

Name: sysLON_IFace_GetOnLineMode

Type: Function
Return value: BOOL
Language type:
Description: OnLine mode of return node.
The function returns a BOOL:
FALSE = Offline.
TRUE = Online.

Input:

Name	Type	Description
dummy	BOOL	fix at TRUE

sysLON_IFace_SetGreenLed

Type: Function
Return value: BOOL
Language type:
Description: Set green led state.

The function returns a BOOL:

FALSE = Error, no PlugIn LON connected
TRUE = Done.

Input:

Name	Type	Description
state	BOOL	TRUE=On, FALSE=Off

`sysLON_IFace_State`

Type: Function
Return value: SINT
Language type:
Description: Returns link state to the neuron chip.

The function returns a SINT:

0	= STOP
1	= RUN
2	= LINKING
3	= ERROR
-1	= UNDEFINED

Input:

Name	Type	Description
dummy	BOOL	fix at TRUE

Init Group

sysLON_IFace_AddCP

Type: Function
Return value: BOOL
Language type:
Description: add a new CP to the current LON profile.
The function returns a BOOL:
FALSE = session is not active.
TRUE = otherwise.

Input:

Name	Type	Description
pnv_info	@_Lon_EW_SNVT_INFO	NV descriptor

sysLON_IFace_AddFunctionBlock

Type: Function
Return value: BOOL
Language type:
Description: add a new function block to the node configuration.
The function returns a BOOL:
FALSE = session is not active.
TRUE = otherwise.

Input:

Name	Type	Description
type	@USINT	function block type, string pointer: i.e. 8030
name	@USINT	function block name, string pointer: i.e. RTU

sysLON_IFace_AddNV

Type: Function
Return value: BOOL
Language type:
Description: add a new NV to the current LON profile
The function returns a BOOL:
FALSE = session is not active.
TRUE = otherwise.

Input:

Name	Type	Description
pnv_info	@_Lon_EW_SNVT_INFO	NV descriptor

sysLON_IFace_ResetConfig

Type: Function
Return value: BOOL
Language type:
Description: reset LON profile NV list
The function returns a BOOL:
FALSE = session is active and running.
TRUE = otherwise.
Input:

Name	Type	Description
dummy	BOOL	fix at TRUE

sysLON_IFace_SetAppSignature

Type: Function
Return value: BOOL
Language type:
Description: new application/node signature.
The function returns a BOOL:
FALSE = session is not active.
TRUE = otherwise.

Input:

Name	Type	Description
app_signature	UINT	application signature

sysLON_IFace_SetProgramID

Type: Function
Return value: BOOL
Language type:
Description: pointer to a byte stream defining program_id.
The function returns a BOOL:
FALSE = session is not active.
TRUE = otherwise.

Input:

Name	Type	Description
program_id	@BYTE	pointer to a byte stream defining program_id: 8 BYTE

sysLON_IFace_Start

Type: Function
Return value: BOOL
Language type:
Description: Start communications to/from neuron chip.
The function returns a BOOL:
FALSE = if session was not started.
TRUE = otherwise.

Input:

Name	Type	Description
dummy	BOOL	fix at TRUE

sysLON_IFace_Stop

Type: Function
Return value: BOOL
Language type:
Description: stop communications to/from neuron chip.
The function returns a BOOL always TRUE

Input:

Name	Type	Description
dummy	BOOL	fix at TRUE

Update Group

sysLON_IFace_ReadCP

Type: Function
Return value: USINT
Language type:
Description: returns actual value of indicated CP.

The function returns a USINT:

0	= ERROR
1	= NOT CHANGED
2	= CHANGED
3	= TIMEOUT

Input:

Name	Type	Description
nv_index	UDINT	index of the desired NV [0..253]
snvt	UDINT	address of specific snvt descriptor

sysLON_IFace_ReadNV

Type: Function
Return value: USINT
Language type:
Description: returns actual value of indicated NV.

The function returns a USINT:

0	= ERROR
1	= NOT CHANGED
2	= CHANGED
3	= TIMEOUT

Input:

Name	Type	Description
nv_index	UDINT	index of the desired NV [0..253]
snvt	UDINT	address of specific snvt descriptor

sysLON_IFace_WriteCP

Type: Function
Return value: USINT
Language type:
Description: update actual value of indicated CP.
The function returns a USINT:
0 = ERROR
1 = NOT WRITTEN
2 = WRITTEN

Input:

Name	Type	Description
nv_index	UDINT	index of the desired NV [0..253]
snvt	UDINT	address of specific snvt descriptor

sysLON_IFace_WriteNV

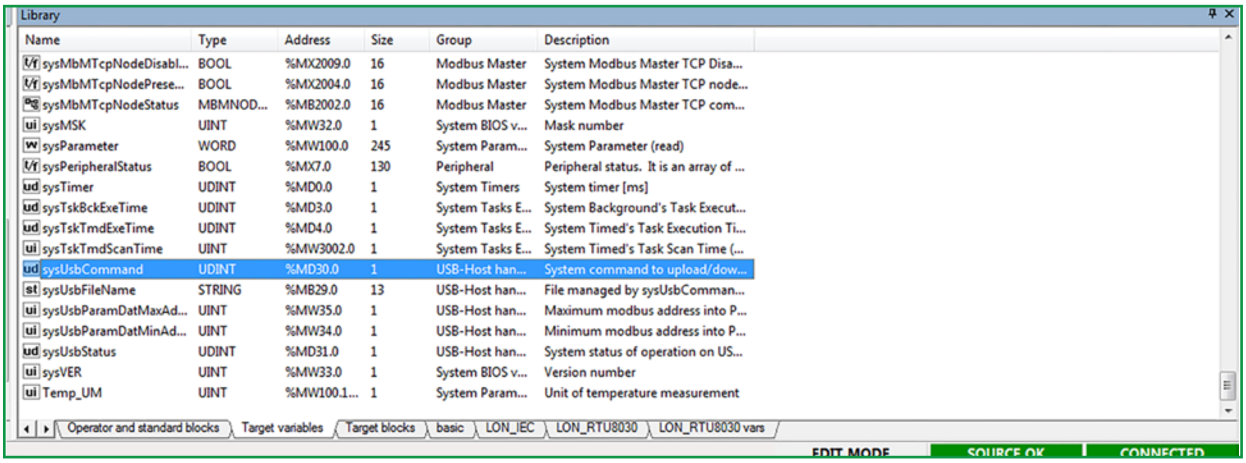
Type: Function
Return value: USINT
Language type:
Description: update actual value of indicated NV.
The function returns a USINT:
0 = ERROR
1 = NOT WRITTEN
2 = WRITTEN

Input:

Name	Type	Description
nv_index	UDINT	index of the desired NV [0..253]
snvt	UDINT	address of specific snvt descriptor

List of USB target variables

This list includes USB target variables related to LON management



Library

Name	Type	Address	Size	Group	Description
sysMbMTcpNodeDisabl...	BOOL	%MX2009.0	16	Modbus Master	System Modbus Master TCP Disa...
sysMbMTcpNodePrese...	BOOL	%MX2004.0	16	Modbus Master	System Modbus Master TCP node...
sysMbMTcpNodeStatus	MBMNOD...	%MB2002.0	16	Modbus Master	System Modbus Master TCP com...
sysMSK	UINT	%MW32.0	1	System BIOS v...	Mask number
sysParameter	WORD	%MW100.0	245	System Param...	System Parameter (read)
sysPeripheralStatus	BOOL	%MX7.0	130	Peripheral	Peripheral status. It is an array of ...
sysTimer	UDINT	%MD0.0	1	System Timers	System timer [ms]
sysTskBckExeTime	UDINT	%MD3.0	1	System Tasks E...	System Background's Task Execut...
sysTskTmdExeTime	UDINT	%MD4.0	1	System Tasks E...	System Timed's Task Execution Ti...
sysTskTmdScanTime	UINT	%MW3002.0	1	System Tasks E...	System Timed's Task Scan Time (...
sysUsbCommand	UDINT	%MD30.0	1	USB-Host han...	System command to upload/dow...
sysUsbFileName	STRING	%MB29.0	13	USB-Host han...	File managed by sysUsbComman...
sysUsbParamDatMaxAd...	UINT	%MW35.0	1	USB-Host han...	Maximum modbus address into P...
sysUsbParamDatMinAd...	UINT	%MW34.0	1	USB-Host han...	Minimum modbus address into P...
sysUsbStatus	UDINT	%MD31.0	1	USB-Host han...	System status of operation on US...
sysVER	UINT	%MW33.0	1	System BIOS v...	Version number
Temp_UM	UINT	%MW100.1...	1	System Param...	Unit of temperature measurement

`Name: sysUsbCommand`

Type: UDINT

Address: %MD30.0

Description: System command to upload/download to/from USB-Host:

7	=	load PARAM.BIN from USBH
8	=	load PLCIEC.COD from USBH
9	=	load HMIIEC.COD from USBH
10	=	load PARAM.DAT from USBH
11	=	save PARAM.DAT to USBH
12	=	load CONNEC.PAR from USBH
13	=	load HMIREM.KBD from USBH
14	=	save sysUsbFileName file to USBH, file name can be name.ext or *.ext
15	=	load sysUsbFileName file from USBH, file name can be name.ext or *.ext
16	=	load file sysUsbFileName from filesystem, file must have PARAM.DAT format and filename name.DAT or name.RAW
18	=	load BACNET.DAT from USBH
20	=	save LON.XIF to filesystem
21	=	save LON.XIF to USBH

`Name: sysUsbStatus`

Type: UDINT

Address: %MD31.0

Description: System status of operation on USB-Host

0	=	command completed
1	=	command processing
255	=	command failed
254	=	file not present
253	=	file too long
252	=	USBH not connected
251	=	file not compatible
250	=	some parameters failed
249	=	write file failed
248	=	open file in write failed

CHAPTER 4

Appendix - Testing

This chapter describes how to test the LON Communication Module with NodeUtil, where to find information on the Web regarding the LON protocol, LON profiles, links for downloading and tips.

4.1. Website

This document and other **Application Notes** are available on the Eliwell site.

4.1.1. LON Profiles and Standard Program Identifier (SPID) Master List

The LON Profile and Standard Program Identifier (SPID) Master List are available at:
http://www.lonmark.org/technical_resources/resource_files/spid_master_list#DeviceClasses

Profile name	Profile #
Roof top unit	8030
Chiller	8040
Heat Pump	8051
Discharge Air Controller	8610

http://www.lonmark.org/technical_resources/guidelines/docs/profiles/8030_11.pdf
http://www.lonmark.org/technical_resources/guidelines/docs/profiles/8040_10.pdf
http://www.lonmark.org/technical_resources/guidelines/docs/profiles/8051_10.pdf
http://www.lonmark.org/technical_resources/guidelines/docs/profiles/8610_10.pdf
http://www.lonmark.org/technical_resources/guidelines/docs/profiles/8500_20.pdf

4.1.2. NodeUtil

Installing the driver of the test tool TP/FT-10

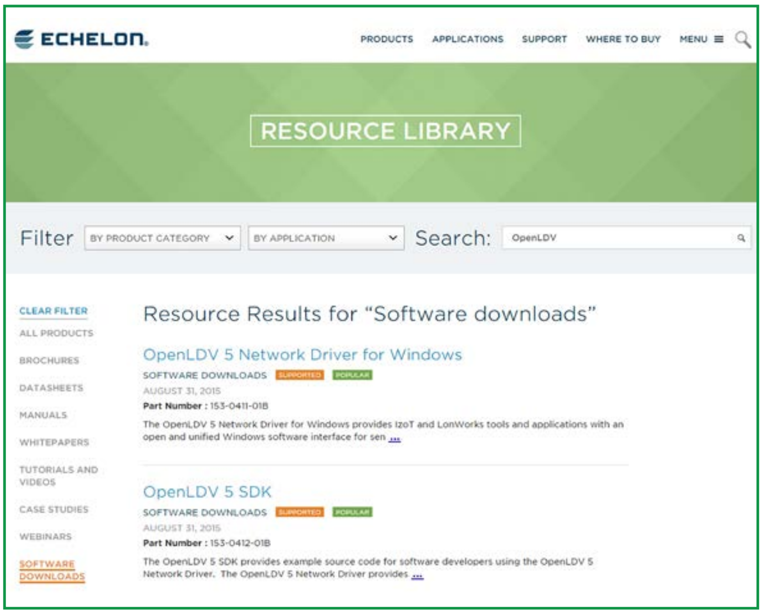
1. Use the SB - LON network converter provided by Echelon Corporation:
   p/n: U10/U20 USB Network Interface

Install LON driver from CD or from the Echelon website: www.echelon.com/downloads

This link redirects to:

http://www.echelon.com/resource-library-results?filters=software_downloads

2. OpenLDV 5 Network Driver for Windows:



Resource Results for "Software downloads"

OpenLDV 5 Network Driver for Windows

SOFTWARE DOWNLOADS

AUGUST 31, 2015

Part Number : 153-0411-01B

The OpenLDV 5 Network Driver for Windows provides IzoT and LonWorks tools and applications with an open and unified Windows software interface for sen ...

OpenLDV 5 SDK

SOFTWARE DOWNLOADS

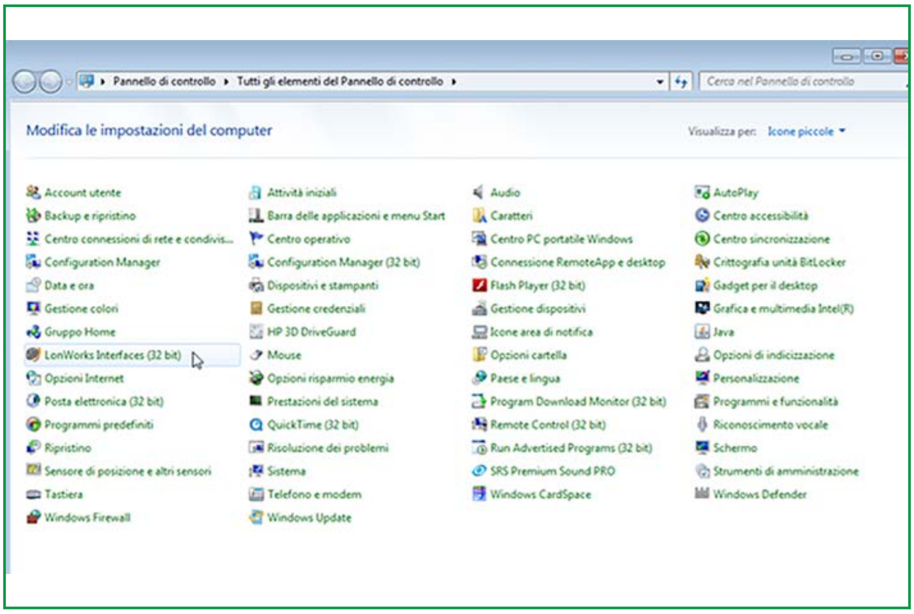
AUGUST 31, 2015

Part Number : 153-0412-01B

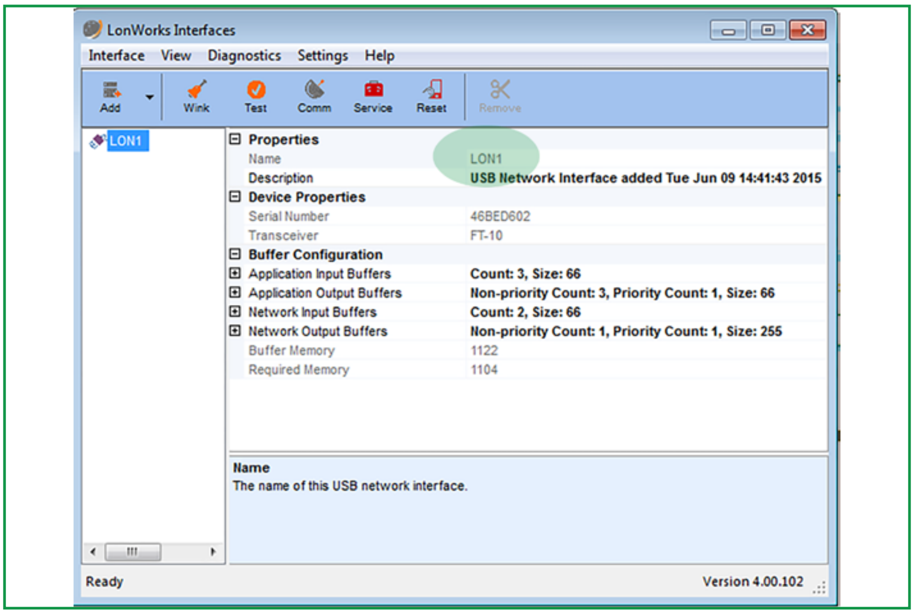
The OpenLDV 5 SDK provides example source code for software developers using the OpenLDV 5 Network Driver. The OpenLDV 5 Network Driver provides ...

NOTE: Do NOT open the OpenLDV 5 SDK page, instead use the link as listed above.

If installation is successfully completed, LonWorks Interfaces will be available in the Control Pane section:



Launch LonWorks Interfaces and check the LON name:



⊟ Properties	
Name	LON1
Description	USB Network Interface added Tue Jun 09 14:41:43 2015
⊟ Device Properties	
Serial Number	46BED602
Transceiver	FT-10
⊟ Buffer Configuration	
⊞ Application Input Buffers	Count: 3, Size: 66
⊞ Application Output Buffers	Non-priority Count: 3, Priority Count: 1, Size: 66
⊞ Network Input Buffers	Count: 2, Size: 66
⊞ Network Output Buffers	Non-priority Count: 1, Priority Count: 1, Size: 255
Buffer Memory	1122
Required Memory	1104

Name
The name of this USB network interface.

Ready

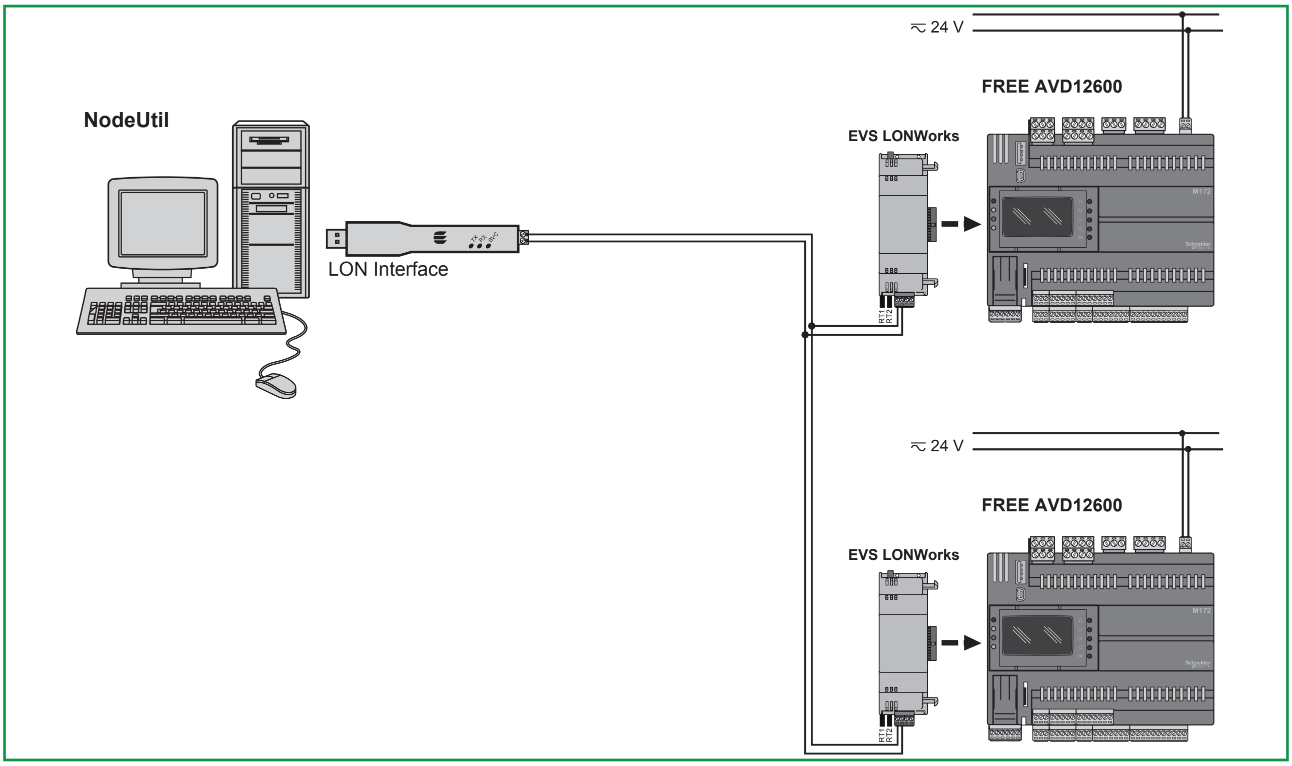
Version 4.00.102

It will be used to reference the LON device. It is the logic name assigned to the device interface.

NOTE:

- Logic name is fixed. Remember this for projects developed with Nodeutil.
- If you connect a second network interface, another name will be assigned (for example LON2).
- The association name/network interface is permanent.
- Each interface (and its logic name) has its own MACADDRESS.

4.2. LonWorks® Interface and Communication Module



NOTE: The LonWorks communication module supports up to 63 nodes. Exceeding this specification may result in an electrical overload condition in the EVS LON Communication Module and consequently in the controller.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

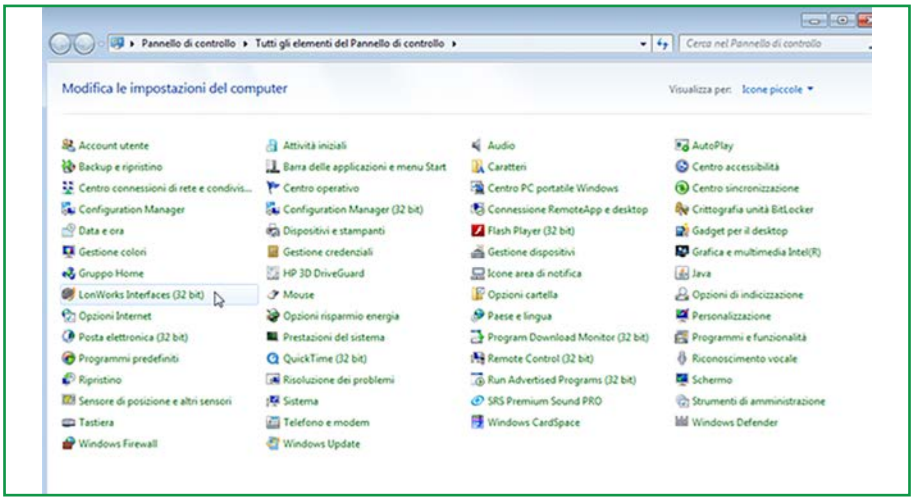
Do not exceed the maximum of 63 nodes on the EVS LON Communication Module.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

For more information on the LonWorks network, visit www.echelon.com/technology/lonwork/

NOTE. Wiring polarity is not mandatory.
NOTE. Service PIN: to be used with NodeUtil or other configuration tools. It will identify the node in the network.

4.3. Testing the Lon Communication Module with NodeUtil



The NodeUtil Node Utility is a testing and diagnostic tool for use with Echelon interfaces.
NodeUtil is NOT a network management tool: use it to configure a single AVD/AVC - EVD/EVC equipment unit, but not to manage the network configuration.

NOTE: Configure the network with LonMaker®; refer to the related Echelon document describing the LNS® based network tools available.

NOTE: Using NodeUtil, it is possible to make a device inoperable so that it no longer communicates with other devices in its network. Certain operations, such as modifying communications parameters to incorrect values, can cause an unrecoverable device failure.

NOTE: The NodeUtil node utility is provided as an UNSUPPORTED software tool.
Refer to the related Echelon document describing the NodeUtil node utility, including how to install and run it.

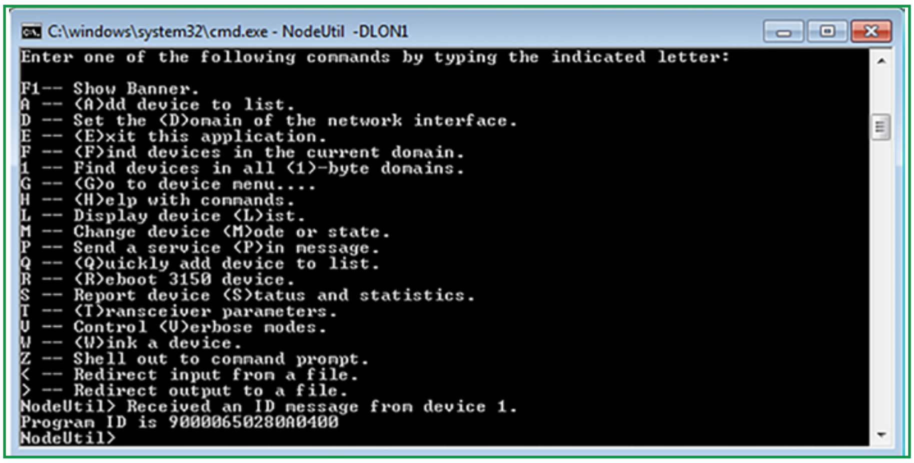
4.3.1. NodeUtil commands

NOTE: Logic name is fixed. Remember this for projects developed with Nodeutil.

Run the Windows command shell and execut the NodeUtil application with the parameter **-DLONX** where **X** = number of the network interface

1st Level: to select one from among n devices. 2nd Level: to select single device
Select Interface to use. For example DLON1:

Press the service button to send a program ID:



```
Enter one of the following commands by typing the indicated letter:

F1-- Show Banner.
A -- <A>dd device to list.
D -- Set the <D>omain of the network interface.
E -- <E>xit this application.
F -- <F>ind devices in the current domain.
1 -- Find devices in all <1>-byte domains.
G -- <G>o to device menu....
H -- <H>elp with commands.
L -- Display device <L>ist.
M -- Change device <M>ode or state.
P -- Send a service <P>in message.
Q -- <Q>uickly add device to list.
R -- <R>eboot 3150 device.
S -- Report device <S>tatus and statistics.
T -- <T>ransceiver parameters.
V -- Control <V>erbose modes.
W -- <W>ink a device.
Z -- Shell out to command prompt.
< -- Redirect input from a file.
> -- Redirect output to a file.
NodeUtil> Received an ID message from device 1.
Program ID is 90000650280A0400
NodeUtil>
```

Use only **G, L** commands (at this level).

NOTICE

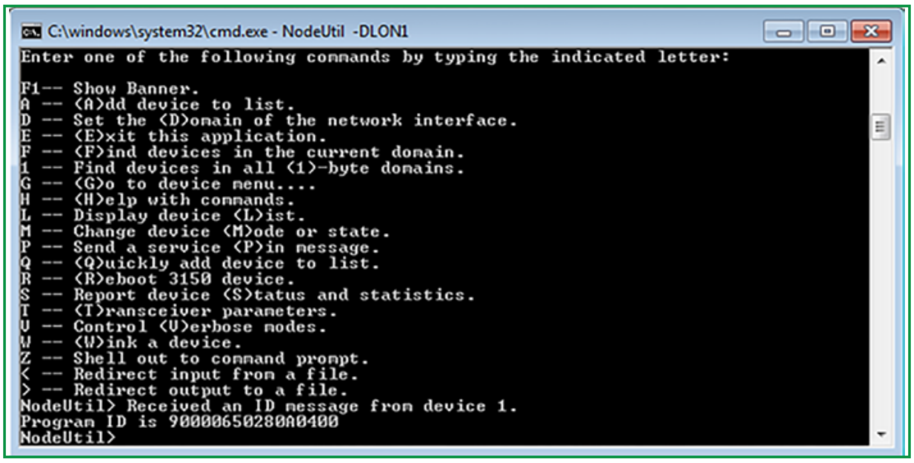
UNINTENDED EQUIPMENT OPERATION

- Use only G, L command at this menu level
- Other commands may damage the equipment.

Failure to follow these instructions can result in equipment damage.

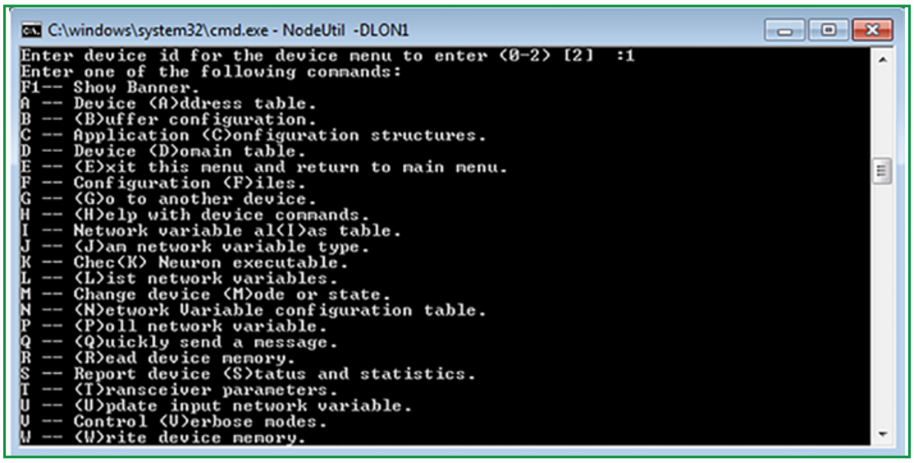
NOTE:
The Profile does not change. The Signature is not constant.
The Profile is defined by LONMARKS.
Adding variables outside the profile cannot be read.
For example: optional variables identify the current set.

L command. Displays the device list:



```
Enter one of the following commands by typing the indicated letter:

F1-- Show Banner.
A -- <A>dd device to list.
D -- Set the <D>omain of the network interface.
E -- <E>xit this application.
F -- <F>ind devices in the current domain.
1 -- Find devices in all <1>-byte domains.
G -- <G>o to device menu....
H -- <H>elp with commands.
L -- Display device <L>ist.
M -- Change device <M>ode or state.
P -- Send a service <P>in message.
Q -- <Q>uickly add device to list.
R -- <R>eboot 3150 device.
S -- Report device <S>tatus and statistics.
T -- <T>ransceiver parameters.
V -- Control <V>erbose modes.
W -- <W>ink a device.
Z -- Shell out to command prompt.
< -- Redirect input from a file.
> -- Redirect output to a file.
NodeUtil> Received an ID message from device 1.
Program ID is 900006502800A0400
NodeUtil>
```

G command. Device menu selection. A new menu level appears:

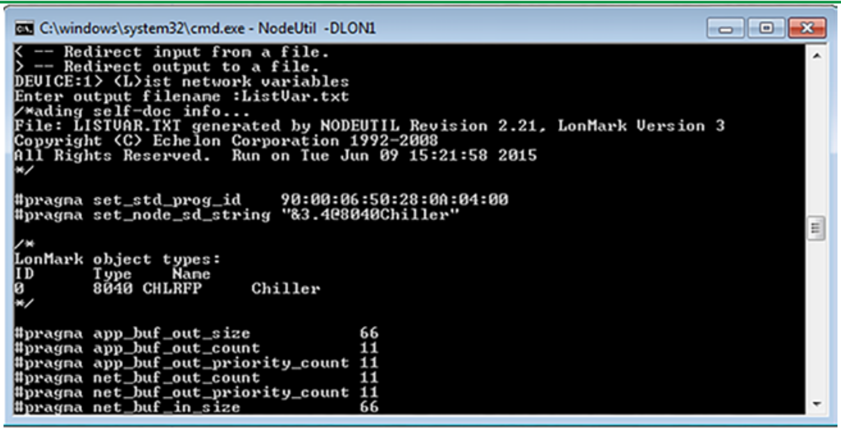


```
Enter device id for the device menu to enter <0-2> [2]  :1
Enter one of the following commands:
F1-- Show Banner.
A -- Device <A>ddress table.
B -- <B>uffer configuration.
C -- Application <C>onfiguration structures.
D -- Device <D>omain table.
E -- <E>xit this menu and return to main menu.
F -- Configuration <F>iles.
G -- <G>o to another device.
H -- <H>elp with device commands.
I -- Network variable al<I>as table.
J -- <J>am network variable type.
K -- Chec<K> Neuron executable.
L -- <L>ist network variables.
M -- Change device <M>ode or state.
N -- <N>etwork Variable configuration table.
P -- <P>oll network variable.
Q -- <Q>uickly send a message.
R -- <R>ead device memory.
S -- Report device <S>tatus and statistics.
T -- <T>ransceiver parameters.
U -- <U>pdate input network variable.
V -- Control <V>erbose modes.
W -- <W>rite device memory.
```

Useful NodeUtil commands

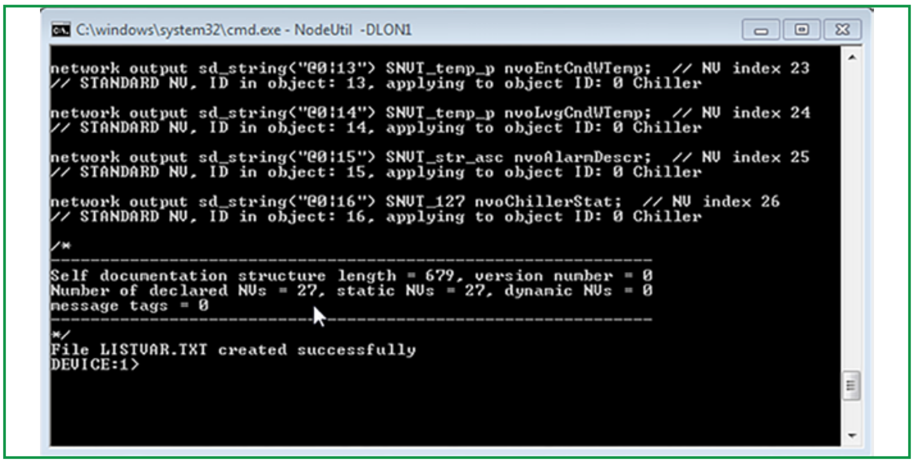
L command: To check the number of variables available in the profile (displays network variables list). Enter variable name to export it to a file.

X command. Generate XIF file. Specify the XIF file name.
Saving on file…
Set_std_Prog_id is a constant related to the current profile
8040 CHLRFP Chiller is the profile name:



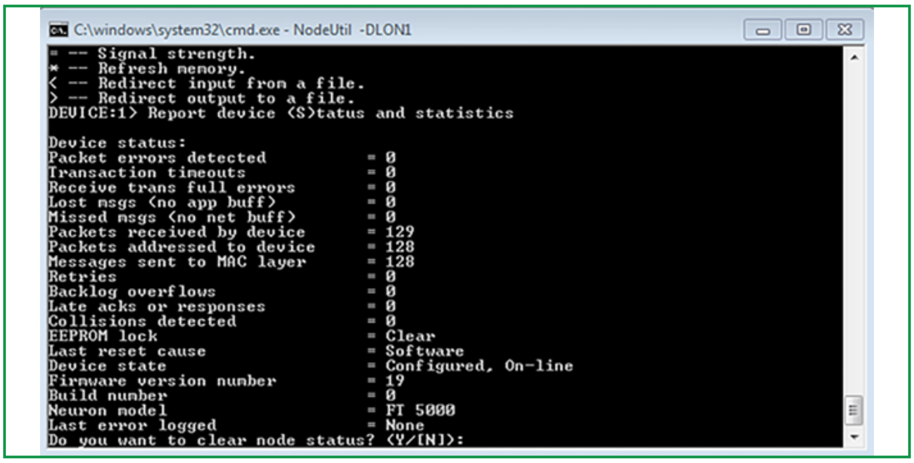
```
< -- Redirect input from a file.
> -- Redirect output to a file.
DEVICE:1> <L>ist network variables
Enter output filename :ListVar.txt
/*ading self-doc info...
File: LISTVAR.TXT generated by NODEUTIL Revision 2.21, LonMark Version 3
Copyright <C> Echelon Corporation 1992-2008
All Rights Reserved.  Run on Tue Jun 09 15:21:58 2015
*/

#pragma set_std_prog_id    90:00:06:50:28:0A:04:00
#pragma set_node_sd_string "&3.4@8040Chiller"

/*
LonMark object types:
ID      Type     Name
0       8040 CHLRFP     Chiller
*/

#pragma app_buf_out_size           66
#pragma app_buf_out_count          11
#pragma app_buf_out_priority_count 11
#pragma net_buf_out_count          11
#pragma net_buf_out_priority_count 11
#pragma net_buf_in_size            66
```

The output is the NodeUtil.exe folder



```
network output sd_string("@0!13") SNVT_temp_p nvoEntCndWTemp;  // NV index 23
// STANDARD NV, ID in object: 13, applying to object ID: 0 Chiller

network output sd_string("@0!14") SNVT_temp_p nvoLvgCndWTemp;  // NV index 24
// STANDARD NV, ID in object: 14, applying to object ID: 0 Chiller

network output sd_string("@0!15") SNVT_str_asc nvoAlarmDescr;  // NV index 25
// STANDARD NV, ID in object: 15, applying to object ID: 0 Chiller

network output sd_string("@0!16") SNVT_127 nvoChillerStat;  // NV index 26
// STANDARD NV, ID in object: 16, applying to object ID: 0 Chiller

/*
-------------------------------------------------------------
Self documentation structure length = 679, version number = 0
Number of declared NVs = 27, static NVs = 27, dynamic NVs = 0
message tags = 0
-------------------------------------------------------------
*/
File LISTVAR.TXT created successfully
DEVICE:1>
```

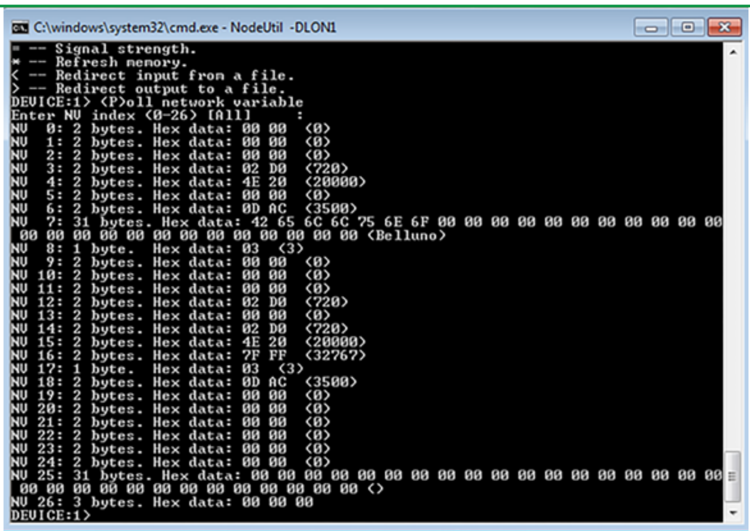
S command. Display Device status. Only LON side:
Check the variables to get an application overview



```
= -- Signal strength.
* -- Refresh memory.
< -- Redirect input from a file.
> -- Redirect output to a file.
DEVICE:1> Report device (S)tatus and statistics

Device status:
Packet errors detected          = 0
Transaction timeouts            = 0
Receive trans full errors       = 0
Lost msgs (no app buff)         = 0
Missed msgs (no net buff)       = 0
Packets received by device      = 129
Packets addressed to device     = 128
Messages sent to MAC layer      = 128
Retries                         = 0
Backlog overflows               = 0
Late acks or responses          = 0
Collisions detected             = 0
EEPROM lock                     = Clear
Last reset cause                = Software
Device state                    = Configured, On-line
Firmware version number         = 19
Build number                    = 0
Neuron model                    = FT 5000
Last error logged               = None
Do you want to clear node status? (Y/[N]):
```

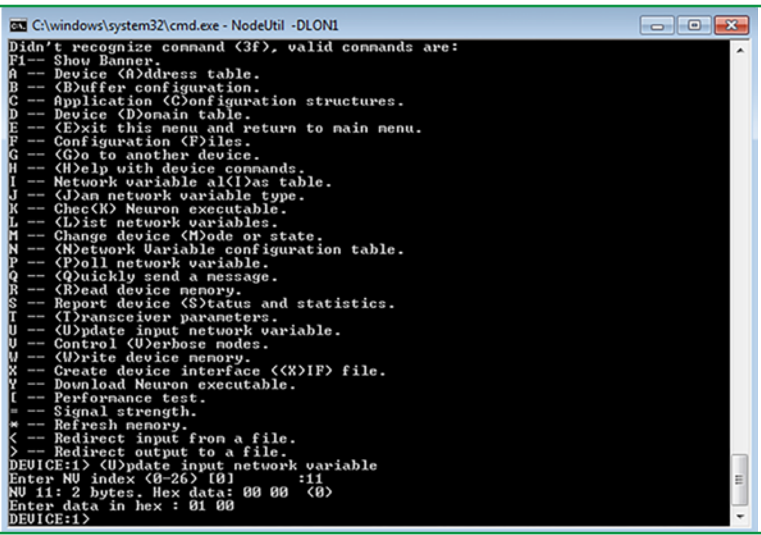
Y command. Clear node statistics.

P command. List of variables in binary mode. Poll all the network variables:

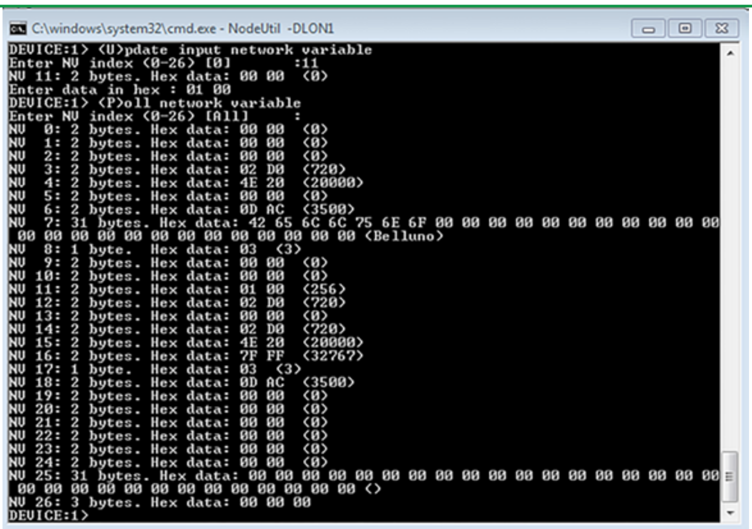


```
= -- Signal strength.
* -- Refresh memory.
< -- Redirect input from a file.
> -- Redirect output to a file.
DEVICE:1> (P)oll network variable
Enter NV index (0-26) [All]     :
NV  0: 2 bytes. Hex data: 00 00  (0)
NV  1: 2 bytes. Hex data: 00 00  (0)
NV  2: 2 bytes. Hex data: 00 00  (0)
NV  3: 2 bytes. Hex data: 02 D0  (720)
NV  4: 2 bytes. Hex data: 4E 20  (20000)
NV  5: 2 bytes. Hex data: 00 00  (0)
NV  6: 2 bytes. Hex data: 0D AC  (3500)
NV  7: 31 bytes. Hex data: 42 65 6C 6C 75 6E 6F 00 00 00 00 00 00 00 00 00 00 00
 00 00 00 00 00 00 00 00 00 00 00 00 00 (Belluno)
NV  8: 1 byte.  Hex data: 03  (3)
NV  9: 2 bytes. Hex data: 00 00  (0)
NV 10: 2 bytes. Hex data: 00 00  (0)
NV 11: 2 bytes. Hex data: 00 00  (0)
NV 12: 2 bytes. Hex data: 02 D0  (720)
NV 13: 2 bytes. Hex data: 00 00  (0)
NV 14: 2 bytes. Hex data: 02 D0  (720)
NV 15: 2 bytes. Hex data: 4E 20  (20000)
NV 16: 2 bytes. Hex data: 7F FF  (32767)
NV 17: 1 byte.  Hex data: 03  (3)
NV 18: 2 bytes. Hex data: 0D AC  (3500)
NV 19: 2 bytes. Hex data: 00 00  (0)
NV 20: 2 bytes. Hex data: 00 00  (0)
NV 21: 2 bytes. Hex data: 00 00  (0)
NV 22: 2 bytes. Hex data: 00 00  (0)
NV 23: 2 bytes. Hex data: 00 00  (0)
NV 24: 2 bytes. Hex data: 00 00  (0)
NV 25: 31 bytes. Hex data: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
 00 00 00 00 00 00 00 00 00 00 00 00 00 ()
NV 26: 3 bytes. Hex data: 00 00 00
DEVICE:1>
```

U command. To update an input network variable:



```
Didn't recognize command (3f), valid commands are:
F1-- Show Banner.
A -- Device (A)ddress table.
B -- (B)uffer configuration.
C -- Application (C)onfiguration structures.
D -- Device (D)omain table.
E -- (E)xit this menu and return to main menu.
F -- Configuration (F)iles.
G -- (G)o to another device.
H -- (H)elp with device commands.
I -- Network variable al(I)as table.
J -- (J)am network variable type.
K -- Chec(K) Neuron executable.
L -- (L)ist network variables.
M -- Change device (M)ode or state.
N -- (N)etwork Variable configuration table.
P -- (P)oll network variable.
Q -- (Q)uickly send a message.
R -- (R)ead device memory.
S -- Report device (S)tatus and statistics.
T -- (T)ransceiver parameters.
U -- (U)pdate input network variable.
V -- Control (V)erbose modes.
W -- (W)rite device memory.
X -- Create device interface ((X)IF) file.
Y -- Download Neuron executable.
[ -- Performance test.
= -- Signal strength.
* -- Refresh memory.
< -- Redirect input from a file.
> -- Redirect output to a file.
DEVICE:1> (U)pdate input network variable
Enter NV index (0-26) [0]        :11
NV 11: 2 bytes. Hex data: 00 00  (0)
Enter data in hex : 01 00
DEVICE:1>
```



```
DEVICE:1> (U)pdate input network variable
Enter NV index (0-26) [0]        :11
NV 11: 2 bytes. Hex data: 00 00  (0)
Enter data in hex : 01 00
DEVICE:1> (P)oll network variable
Enter NV index (0-26) [All]      :
NV  0: 2 bytes. Hex data: 00 00  (0)
NV  1: 2 bytes. Hex data: 00 00  (0)
NV  2: 2 bytes. Hex data: 00 00  (0)
NV  3: 2 bytes. Hex data: 02 D0  (720)
NV  4: 2 bytes. Hex data: 4E 20  (20000)
NV  5: 2 bytes. Hex data: 00 00  (0)
NV  6: 2 bytes. Hex data: 0D AC  (3500)
NV  7: 31 bytes. Hex data: 42 65 6C 6C 75 6E 6F 00 00 00 00 00 00 00 00 00 00 00
 00 00 00 00 00 00 00 00 00 00 00 00 00 (Belluno)
NV  8: 1 byte.  Hex data: 03  (3)
NV  9: 2 bytes. Hex data: 00 00  (0)
NV 10: 2 bytes. Hex data: 00 00  (0)
NV 11: 2 bytes. Hex data: 01 00  (256)
NV 12: 2 bytes. Hex data: 02 D0  (720)
NV 13: 2 bytes. Hex data: 00 00  (0)
NV 14: 2 bytes. Hex data: 02 D0  (720)
NV 15: 2 bytes. Hex data: 4E 20  (20000)
NV 16: 2 bytes. Hex data: 7F FF  (32767)
NV 17: 1 byte.  Hex data: 03  (3)
NV 18: 2 bytes. Hex data: 0D AC  (3500)
NV 19: 2 bytes. Hex data: 00 00  (0)
NV 20: 2 bytes. Hex data: 00 00  (0)
NV 21: 2 bytes. Hex data: 00 00  (0)
NV 22: 2 bytes. Hex data: 00 00  (0)
NV 23: 2 bytes. Hex data: 00 00  (0)
NV 24: 2 bytes. Hex data: 00 00  (0)
NV 25: 31 bytes. Hex data: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
 00 00 00 00 00 00 00 00 00 00 00 00 00 ()
NV 26: 3 bytes. Hex data: 00 00 00
DEVICE:1>
```

E+E command. Type twice to exit the application.

4.4. Commissioning

The LonMaker® Integration Tool is a software package for designing, installing, and maintaining multi-vendor, LonWorks® control networks.
The LonMaker® is a licensed application. For more information on LonMaker® and LonWorks® visit www.echelon.com/technology/network

NOTE: you can use the Echelon LonMaker® to connect the **nvo** of a node with the **nvi** of another node.