User Manual

Easylon ISA-Bus Interface
Easylon PC/104 Interface

Gesytec GmbH
Pascalstr. 6
52076 Aachen, Germany

Tel. + (49) 24 08 / 9 44-0
Fax + (49) 24 08 / 94 4-100
email: info@gesytec.de
www.gesytec.com

Doc. ID: 96A0016E01V3.9
Date: June 2014

Gesytec

This manual …

… provides you with all the information which you will need to use the Easylon® ISA-Bus Interface and Easylon PC/104 Interface cards.

However, this manual will neither explain aspects of Echelon's® LONWORKS® technology, nor Echelon's Microprocessor Interface Program (MIP), or Network Service Interface (NSI) used on theses network interface cards. The interface card network drivers have been designed in accordance with the driver specifications of the Echelon Corporation. They are not explained in detail in this manual. For further information on the LONWORKS technology please refer to the extensive documentation provided by Echelon. Especially Echelon's "LONWORKS Host Application Programmer's Guide" will be required if applications are to be developed using Gesytec's Easylon ISA-Bus or PC/104 Interface cards as a network interface.

After a general presentation of both Easylon Interface cards in Chapter 1, Chapter 2 describes the necessary steps to install the cards.

Chapter 3 contains a general technical description.

Chapter 4, "Programming Instructions", contains the information which will be of importance, should you wish to develop your own network driver software for one of the Easylon Interface cards.

Archiv/96A0016E01, V3.9, June 2014

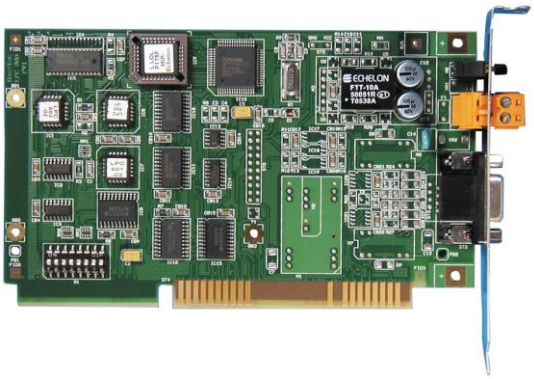
Contents

Archiv/96A0016E01, V3.9, June 2014

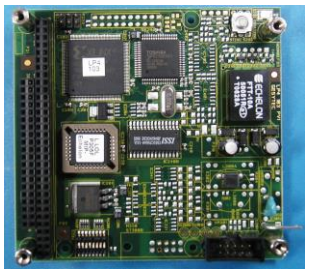
Archiv/96A0016E01, V3.9, June 2014

1 Product Information

Two Easylon Interfaces cards are described in this manual:



Easylon ISA-Bus Interface, plug-in card for short 16 bit ISA bus slots.



Easylon PC/104 Interface, ISA bus Interface card in PC/104 format.

Easylon PC/104 Interface, version LP43

NOTE: In this manual both cards are generally referred to as “Interface cards”. If differences between the two have to be described, they are explicitly referred to as “Easylon ISA-Bus Interface” or “Easylon PC/104 Interface”.

Archiv/96A0016E01, V3.9, June 2014

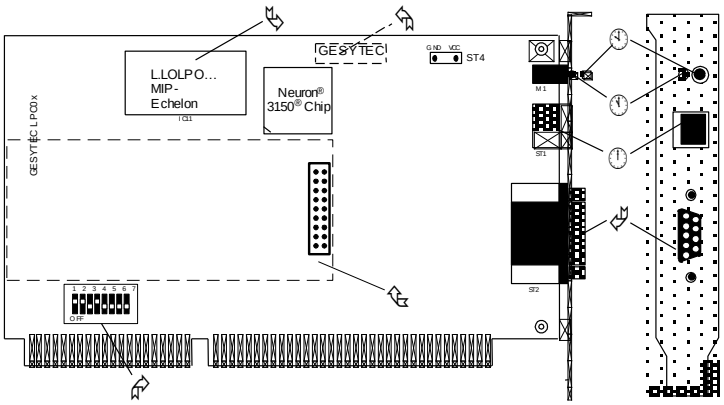


Figure 1-1 Easylon ISA-Bus Interface

(1) Service button
(2) Service LED
(3) Screw-plug terminal (TP/XF and FTT[1] variants only)
(4) 9 pin D-type connector
(5) ROM with Echelon's MIP/P50 or NSI
(6) Type identifier on rear side (see table 1.1)
(7) DIP switches for setting card address
(8) Connector for Watcher[2] piggyback module

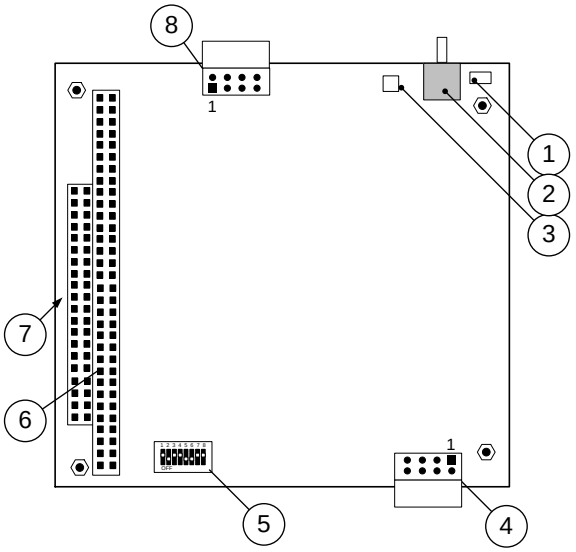


Figure 1-2 Easylon PC/104 Interface, version LP43

Type code: LP43.xxx

(1) Service LED
(2) Service button
(3) LON traffic LED (RX: rot, TX: grün)
(4) Connector LON and shield
(5) DIP switches for setting card address
(6) PC/104-bus connector
(7) Type identifier and serial-# on connect.
(8) Connector for external signals

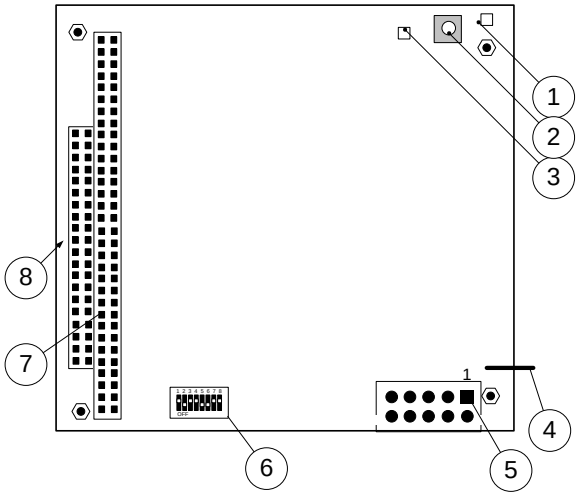


Figure 1-3 Easylon PC/104 Interface, version LP42

Type code: LP42.xxx

(1) Service LED
(2) Service button
(3) LON traffic LED (optional)
(4) Shield
(5) block terminal for LON bus (pin 1)
(6) DIP switches for setting card address
(7) PC/104-bus connector
(8) Type identifier and serial-# on connect.

1 TPT = transformer coupled twisted pair,
FTT = free topology transceiver

2 The Easylon Watcher has been discontinued.

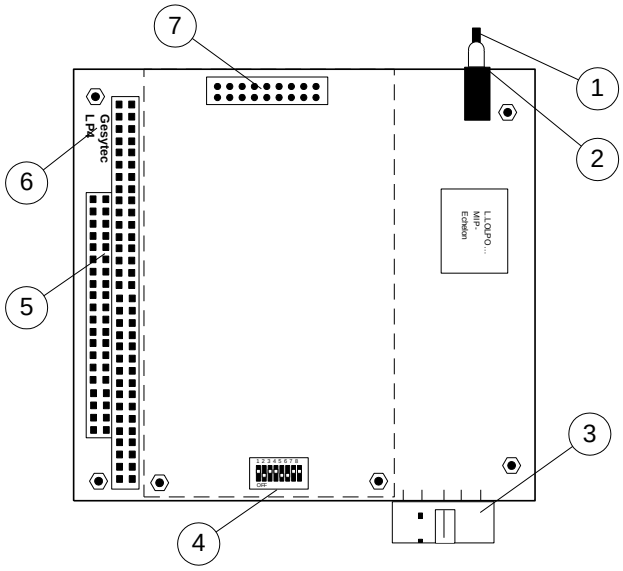


Figure 1-4 Easylon PC/104 Interface, version LP4

Type code: LP4.xxx

(1) Service button
(2) Service LED
(3) 10 pin block connector
(4) DIP switches for setting card address
(5) PC/104 connector
(6) Type identifier on rear side (see table 1.1)
(7) Connector for Watcher[3] piggyback module´

3 The Easylon Watcher has been discontinued.

1.1 Variants

The following variants of the Easylon Interface cards are currently available and describe in this manual. There are further, customer specific versions, to which this manual only partially applies. Each variant is identifiable by a type code sticker on the rear of the card.

Type Code	Order Code	Network Interface	Variants
Easylon ISA-Bus Interface			
LPC.AA	P.P10004	FTT	MIP/P50
LPC.AB	P.P11004	FTT	MIP/P50 + Watcher[4]
LPC.AC	P.P10014	FTT	NSI
LPC.AD	P.P11014	FTT	NSI + Watcher
LPC.BA	P.P10001	TP/XF-78	MIP/P50
LPC.BB	P.P11001	TP/XF-78	MIP/P50 + Watcher
LPC.BC	P.P10011	TP/XF-78	NSI
LPC.BD	P.P11011	TP/XF-78	NSI + Watcher
LPC.CA	P.P10002	TP/XF-1250	MIP/P50
LPC.CB	P.P11002	TP/XF-1250	MIP/P50 + Watcher
LPC.CC	P.P10012	TP/XF-1250	NSI
LPC.CD	P.P11012	TP/XF-1250	NSI + Watcher
LPC.DA	P.P10003	E/A-485	MIP/P50
LPC.DB	P.P11003	E/A-485	MIP/P50 + Watcher
LPC.DC	P.P10013	E/A-485	NSI
LPC.DD	P.P11013	E/A-485	NSI + Watcher

Table 1-1 Variants, order-codes and type identifiers for Easylon ISA-Bus Interface

Archiv/96A0016E01, V3.9, June 2014

4 The Easylon Watcher has been discontinued.

Type Code	Order Code	Network Interface	Variants
Version LP43			
LP43.FBB	P.P73106	FT-X1	MIP/P50, ext. temp., coating
Version LP42			
LP42.BA	P.P10601	TP/XF-78	MIP/P50
LP42.BC	P.P10611	TP/XF-78	NSI
LP42.CA	P.P10602	TP/XF-1250	MIP/P50
LP42.CC	P.P10612	TP/XF-1250	NSI
LP42.DA	P.P10603	RS-485	MIP/P50
LP42.DC	P.P10613	RS-485	NSI
LP42.AA	P.P10604	FTT10	MIP/P50
LP42.AC	P.P10614	FTT10	NSI
LP42.EA	P.P10605	Direct Connect	MIP/P50
LP42.EC	P.P10615	Direct Connect	NSI
LP42.FA	P.P10606	FTX	MIP/P50
LP42.FC	P.P10616	FTX	NSI
Version LP4			
LP4.AA	P.P10104	FTT	MIP/P50
LP4.AB	P.P11104	FTT	MIP/P50 + Watcher[5]
LP4.AC	P.P10114	FTT	NSI
LP4.AD	P.P11114	FTT	NSI + Watcher
LP4.BA	P.P10101	TP/XF-78	MIP/P50
LP4.BB	P.P11101	TP/XF-78	MIP/P50 + Watcher
LP4.BC	P.P10111	TP/XF-78	NSI
LP4.BD	P.P11111	TP/XF-78	NSI + Watcher
LP4.CA	P.P10102	TP/XF-1250	MIP/P50
LP4.CB	P.P11102	TP/XF-1250	MIP/P50 + Watcher
LP4.CC	P.P10112	TP/XF-1250	NSI
LP4.CD	P.P11112	TP/XF-1250	NSI + Watcher

Table 1-2 Variants, order-codes and type identifiers for Easylon PC/104 Interface

Archiv/96A0016E01, V3.9, June 2014

5 The Easylon Watcher has been discontinued.

1.2 Scope of Delivery

- PC plug-in card with Echelon‘s MIP/P50 or NSI firmware
- Installation and Documentation CD including
  - network drivers for 32 bit and 64 bit versions of Windows XP, Vista, 7, Server 2003, 2008, 2008 R2
  - Easylon RNI Software for remote LONWORKS access
  - EasyCheck utility for Easylon Interfaces
  - WLDV32.DLL
  - Documentation in Adobe Acrobat .PDF format

1.3 Overview

The Easylon Interface cards are a cost-effective link between a standard PC or a PC/104 computer and the LON network. They provide one interface to the network. Variants are available for connection via transformer coupled twisted pair (TP/XF), free topology (FTT) FTX smart, direct connect transceivers and for EIA-485 serial connection.

Firmware is alternatively MIP/P50 or NSI. While MIP is the cheaper solution offering more memory NSI will only be needed if LNS is used.

The Easylon ISA-Bus Interface card is provided with a service button and LED. In the TP/XF and FTT variants, this card is equipped with two connectors, either of which can be used:

- 9 pin D-type connector
- 2 pin screw-plug terminal

Connection to the Easylon PC/104 Interface is made by a 10-pin block connector. Service button and LED are available on this card as well. Optionally (using LP42 cards) the signals can as well be used externally.

The PC/104 card is as well available in version for extended temperature range. Furthermore there are version with a protective coating. These versions comply with the standard relevant to railway technology.

As an alternative solution to the standard LonTalk adapters described in this documentation there is a modern solution available with the Easylon Interfaces^{+}. Especially in more demanding applications these ISO/IEC 14908 based LonWorks compatible network interfaces offer many advantages. Find more information at www.gesytec.com.

Archiv/96A0016E01, V3.9, June 2014

2 Installation

Installation of both types of the Easylon Interface cards is carried out in three steps:

1. Setting the I/O address of the card
2. Insertion of the card
3. Installation of the desired network driver

Note: The Easylon Interface cards are delivered in status "unconfigured". Prior to using it as a LON network interface it has to be set "configured". Standard applications available from the market, such as network management tools, automatically set this status or offer an appropriate command.

For customer specific applications which shall use the Easylon Interface cards the status setting has to be taken care of. Chapter 4 gives further hints on this subject.

The external interface files (.xif) for the different card variants can be found in the XIF directory of the installation CD. Table 4-1 references the different file to the interface card variants.

2.1 Setting the I/O Addresses

Both the Easylon ISA-Bus Interface and the Easylon PC/104 Interface are assigned four I/O addresses on the ISA bus of the PC. The base address of this I/O range is set via DIP switches (cf. Figure 1-1, to Figure 1-4). Before setting the base address, establish which address range is available on your computer in its current configuration.

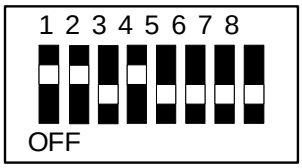


Figure 2-1 DIP switches

From left to right, the switches define address bits 1...8, address bits 9 and 10 are set to 00 and can't be set by the user. When a switch is in its upper position (ON), the address bit is defined as '1'; in the lower position, the address bit is defined as '0'.

The setting shown in Figure 2.1 corresponds to I/O base address $340 (11 0100 0000).

Archiv/96A0016E01, V3.9, June 2014

Examples:

Address	A9	A8	A7	A6	A5	A4	A3	A2
200	1	0	0	0	0	0	0	0
300	1	1	0	0	0	0	0	0
320	1	1	0	0	1	0	0	0
340	**1**	**1**	**0**	**1**	**0**	**0**	**0**	**0**
380	1	1	1	0	0	0	0	0

Table 2-1 Setting the card address

2.2 Insertion of the Card

Prior to inserting the interface cards read the Mounting instructions enclosed in the delivery. When inserting the Easylon ISA-Bus Interface card or the Easylon PC/104 card in your computer, please be sure to observe all the computer manufacturer's instructions regarding the insertion of additional interface cards.

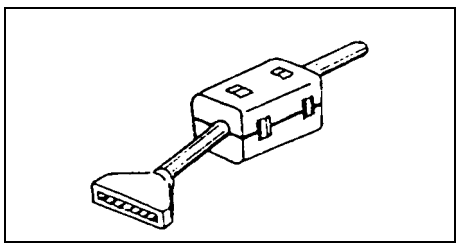
- Insert the Easylon ISA-Bus Interface card into an available 16-bit ISA slot,
- Plug the Easylon PC/104 Interface into the PC/104 bus connector.

Connect the interface card with an appropriate cable to the LON network. Please refer to Section 3.5, for information regarding the connector pin assignment.

2.2.1 Mounting the Ferrite Core

The Easylon Interface cards are CE certified products and meet the intent of *Directive 2004/108 for Electromagnetic Compatibility*.

To ensure electromagnetic compatibility in accordance with *Directive* for operation of the cards the ferrite core enclosed in the delivery has to be mounted around the interface cable to the LON network.



Please put the ferrite core around the interface cable next to the connector on the board end side (see Figure 2.2). The distance between the ferrite core and the connector on the board must not exceed 10 cm. Ensure, if necessary, that it will stay in place.

Figure 2-2 Mount ferrite core close to connector

2.3 Installation of the Network Driver

Drivers for different operating systems are available for the Easylon PCI Interface. Currently these are Windows 2000, XP, Vista and 7 and the Windows Server OS 2003, 2008 und 2008 R2. The drivers support both, the 32 and the 64 bit version of these operating systems. Furthermore there are drivers for Windows CCE, Linux and MS-Dos. Latest driver versions you can download via the Easylon Support pages of our web site: www.gesytec.com.
Installation is described in the following sections:

Windows operating systems	chapter 2.3.1
Windows 95/NT	chapter 2.3.2
16-Bit driver under 32-bit Windows	chapter 2.3.4
Windows CE (x86)	chapter 2.3.5
DOS Driver	chapter 0

This section also describes in short the diagnosis utility “EasyCheck” which can be installed separately from CD.

A Linux driver is available in source code from the “Linux” directory on the CD. This also contains hint s and comments.

The “Driver and Documentation” CD will lead you to the installation of drivers for different operating systems (OS). However, all setups can as well be started manually for each OS and the respective interface card directly from the CD.

2.3.1 Driver for Windows Operating System (WDM Drivers)

This section describes installation and setup of the Easylon Interface card drivers for the Windows operating system from Windows XP onwards.

This setup will install the same driver for all operating systems (Windows Driver Model).

Finally de-installation of the driver is explained

2.3.1.1 Installation

As these Easylon Interfaces are no Plug-and-Play devices the driver must be installed manually. The setup is either started by following the guidance given by the CD or by selecting the directory “Lpclpp” on the Drivers & Documentation CD and starting the file

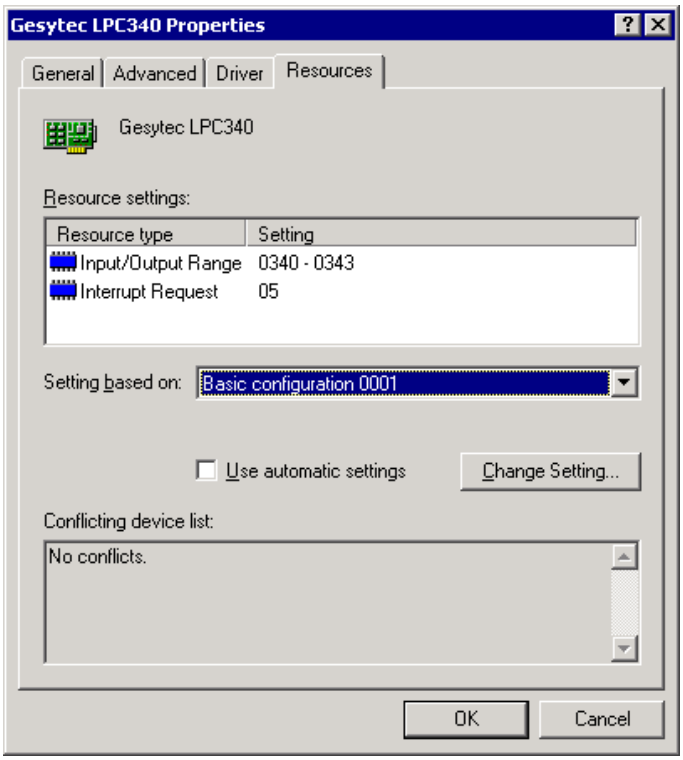
FastInst.exe

The driver will be installed after a few seconds with its standard settings. If you want to install more than one board, start FastInst.exe the appropriate number of times and change the resources manually.

The Easylon Interfaces driver will installed with the following standard configuration:

I/O address range: 340-344,
IRQ 5

If the Easylon Interface is to use different resources, these have to be set manually using the PCs device manager.



Alternative Installation methods

Alternatively you may use the Windows Hardware Assistant for driver installation. However, this may require stepping through up to 15 dialog boxes.

If you are using this method anyway, please select the setup file „LpcWdm.inf“ and the „Gesytec LPC WDM Driver ISA/PC-104“ driver. One advantage you will get using this method: resource selection will be made during the installation procedure.

Archiv/96A0016E01, V3.9, June 2014

2.3.1.2 Update

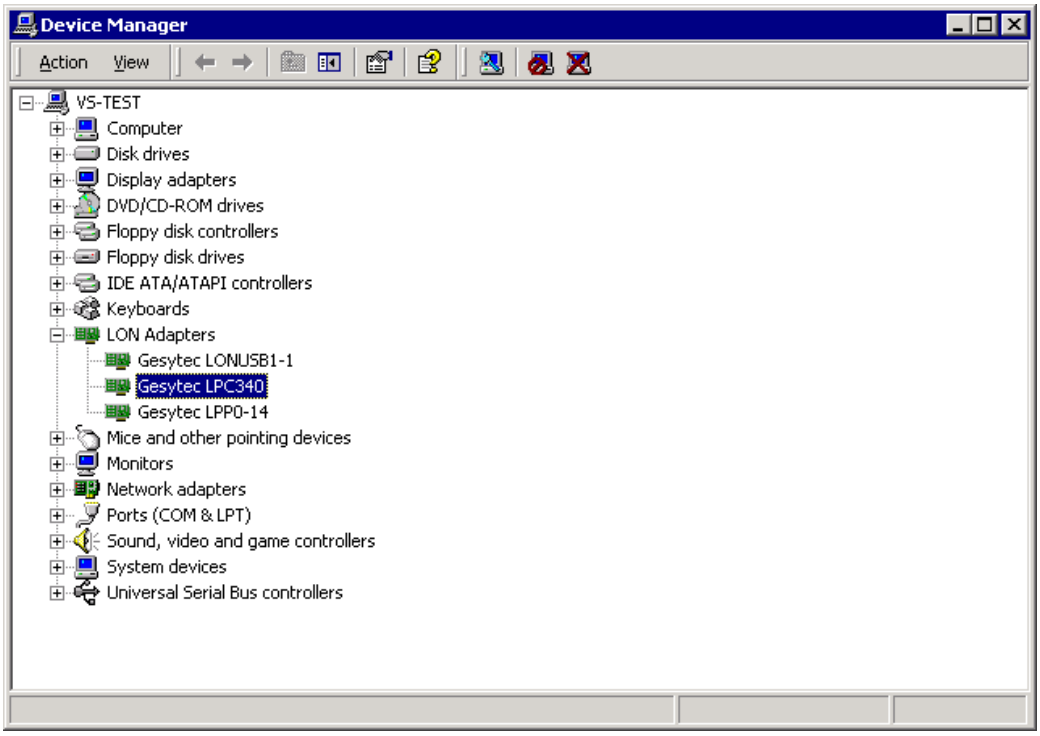
In case you want to update an existing driver start

FastUpd.exe

from the “Lpclpp” directory on the CD-ROM. A new version will be installed from CD within a few seconds.

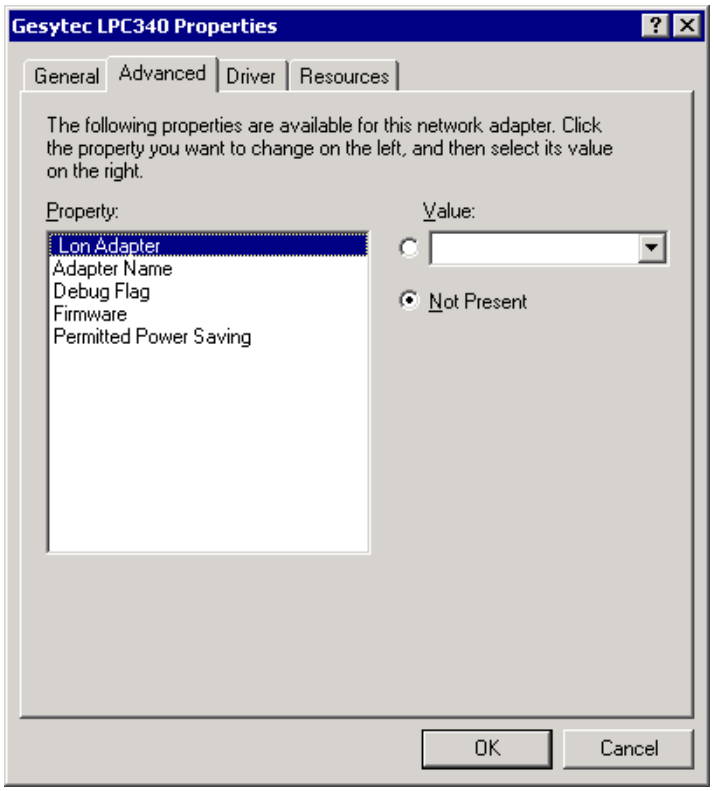
2.3.1.3 Parameter Setting

Certain operating conditions may require special settings for the Easylon Interfaces. In the Device Manager select the respective interface board under LON Adapters and modify the settings.



The „Advanced“ TAB allows setting of individual properties.

Archiv/96A0016E01, V3.9, June 2014



Lon Adapter

You can assign a name „LON1“ ... „LON9“ to the interface board, as some applications may require.

ATTENTION The name chosen must not be in use by any other driver. The device will not start if a name is used twice. (Code 10).

Adapter Name

Alternatively a name can be chosen freely (e.g. building 7).

NOTE If names are given in both sections „Lon Adapter“ and „Adapter Name“ only the name defined under „Lon Adapter“ will be used.

Debug Flag

This field contains a DWORD in hexadecimal notation of different flags for debug purposes. Usually it is set to 0 (= not existing). By setting the single bits certain debug features can be turned on. At the moment the bits 0, 1 and 2 are used:

Bit 0: LON telegrams at the interface from and to the application are displayed in the debug output.

Bit 1: LON telegrams at the interface from and to the Neuron Chip are displayed in the debug output.

Archiv/96A0016E01, V3.9, June 2014

Bit 2: LON telegrams at the interface from and to the Easylon Watcher[6] are displayed in the debug output.

Bit 3: CREATE and CLOSE) of the driver are displayed in the debug output.

NOTE: The Debug Output can be displayed using, for example, the program DebugView, available at www.sysinternals.com.

Firmware

The options MIP/NSI or EEBLANK are displayed. These settings are for future releases and do not have any effects now.

Permitted Power Saving

Usually the Easylon Interface allow the standby mode as well as the hibernate mode. In certain operating conditions it can lead to errors if the PC, with a LON application running, changes to hibernate or standby mode. This can be turned off by selecting "None".

2.3.1.4 De-installation

De-installation uses the Windows Device Manager. Select the driver „Gesytec LPCxxx" under „LON Adapters" with the right mouse key and choose de-install

2.3.2 Windows 95 / NT Driver

This section describes the driver setup for the Easylon Interface cards for the Windows 95 and Windows NT operating systems.

The Setup program will automatically install the network driver required by the computer's operating system.

Finally, the de-installation procedure of the network driver is explained.

2.3.2.1 Installation

The setup can be started automatically from the user interface of the „Easylon Drivers & Documentation" CD-ROM.

You can as well start the setup from the CDs file system:
Go to the „Setup" directory and the sub-directory belonging to the Easylon Interface board: „Easylon PC + PC-104 Interface 95+NT". Start the file

Setup.exe

6 The Easylon Watcher has been discontinued.

During the installation, you will be asked to specify the path for the installation of the utilities and demo sources. Optionally, you can accept the suggested path or may specify your own.

At the end of the setup, the following dialogue will appear:

Here, you can adjust the I/O address of the Easylon Interface card and the allocated interrupt (IRQ).

NOTE: This driver does not support multiple Interfaces in one computer!

2.3.2.2 De-installation

De-installation of the drivers is done via the system control software. For this, first choose the item "Easylon ISA-Bus Interface" and then click "Insert/Delete". After the de-installation, the system has to be restarted.

2.3.3 EasyCheck – Test Utility for Windows Drivers

In addition to the drivers, the test utility “EasyCheck” can be installed in the respective program directory (default: : \Easylon\Lpx). The program checks interface and software environment and displays information, from which can be concluded on the reasons for problems in connection with the interface.

EasyCheck runs an analysis of the system’s software. It will open the selected interface, check the driver version and display it. By sending a “query status” command the communication with the hardware is tested. Using the “read memory” command the utility will show if the device is running MIP or NSI firmware. Properly installed Easylon Interfaces will send a corresponding answer.

2.3.4 Windows and 16 Bit Applications

The Windows driver for the 32 bit Windows versions also provides a 16 bit interface. (Unfortunately Microsoft does not support this in the 64 bit versions.) To use it, the following entry has to be made in the file „config.nt“, usually found in the windows\system32 directory:

Device=%SystemRoot%\system32\ lpxdos.exe –Llpcwdm340

A more specific definition of the 32 bit LON device used is made by optional parameter:

/Lname

name =

lpcdrv for device EasyLPC number 1

Note: Two subsequent “l” characters have to be entered, one indicating the parameter -L, the second as first character of the name: –Llxxxx

A more specific definition of the 16 bit LON device used is made by optional parameter:

/Dn with n = 1...9 for LON1 to LON9

Without this parameter, the interface will be assigned the first unused name starting with “LON1”.

2.3.5 Windows CE Driver

The Windows CE driver has been designed for x86 processors. Variants for other processors can be realized on request. There are versions for Windows up to CE 6.0.

Note: Prior to using the interface please check if your Windows CE system supports USB. For instance you could connect a standard USB device like mouse, keyboard or memory stick.

The Windows CE driver comes is a DLL named lonusb.dll. Like all Windows CE drivers it must be in the Windows directory of your system. The required files can be found on the Driver & Documentation CD under Drivers/Windows CE.

If the driver has to be integrated into the Windows CE image, the simplest way is a respective entry in the platform.bib file. This approach is almost the same for all Windows CE versions.

Archiv/96A0016E01, V3.9, June 2014

Foe correct operation the driver requires registry entries. These can be found in the file lonusb.reg. In order to integrate the driver into a Windows CE image, the contents of this file has to be copied into the file platform.reg.

```
; LONUSB - Driver
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\3596\Default\De
fault\LonUsb]
    "DLL"="lonusb.dll"
    "Prefix"="LON"
    "DebugFlag"=dword:0
    "ReadTimeout"=dword:FFFFFFFF
```

2.3.6 DOS Driver

The network driver for MS-DOS supplied with the Easylon Interface card has been designed in accordance with the specifications by Echelon Corporation. For information on the network driver interface which is required to develop applications, please refer to the "LONWORKS Host Application Programmer's Guide" from Echelon.

The driver can be taken form the CD-ROM's „DOS" directory. There are two versions:

- Driver without interrupt „lpcdrv.exe"
- Driver with interrupt „lpcdrv2.exe".

The driver files „lpcdrv.exe" or „lpcdrv2.exe" have to be copied onto the hard disk of your computer, e.g. into a directory named C:\easylon.

The network driver for the Easylon Interface cards requires 1.6 Kbytes of resident program code, 2 Kbytes of output buffer and 2, 4 or 8 Kbytes of input buffer.

2.3.6.1 Installation

The network driver will be installed in the system as device with the first free name starting with „LON1:" by adding in the "config.sys" file the line

device =*{path}*\lpcdrv2.exe /A*[port address] /Q[irq nr]*

or

devicehigh =*{path}*\lpcdrv2.exe /A*[port address] /Q[irq nr]*

{path} is describing the location to which the of the driver file was copied. The *[port address]* is the I/O address previously set on the card by means of the DIP switches. To set the default port address, the IRQ 10 (0Ah) and use the path proposed above, the correct entry here would be:

device=C:\easylon/lpcdrv2.exe /A340 /QA

Options

The following options can be used:

/A This parameter specifies the port address in hexadecimal form. It must always be set, if the default address (340h) is not used. Addresses which are reserved for standard peripherals (COM1: – COM4:, LPT1: – LPT3:, floppy disk, hard disk, video adapter and the I/O modules on the motherboard) **must not** be used! Any attempt to do so will result in the error message:

No or invalid port address

If the address is valid, but no Easylon Interface is installed in the computer or configured for the stipulated address, the message is:

Interface card is not responding

/Q (only valid for lpcdrv2.exe)
This parameter specifies the IRQ number in hexadecimal notation. It must always be set if the default IRQ (5) is not used.
The following IRQs can be used: 3, 5, 7, 9, A, B, C or F
If an invalid IRQ is entered the following error message will be displayed:

Error: Only IRQ 3,5,7,9,A,B,C or F allowed

/D Setting of device number
The device number may be in the range from 1 to 9 (LON1: – LON9:). If this option is not specified, the network driver will be assigned the smallest free number (default).
If another network driver has already been installed with the same device number, this will result in the error message:

Invalid or duplicate device name

If the parameter /D is specified without entering a subsequent numerical value, the device number will be assigned automatically. If all possible device numbers have already been assigned to other network drivers, this will result in the error message:

LON1: ... LON9: already defined

/I Increasing the input buffer
The input and output buffers of the Easylon ISA-Bus Interface network driver are configured as byte-level FIFOs, i.e. the space requirement of a message is dependent on its length. Consequently, a buffer capacity of 2 Kbytes (default, approx. 50–100 messages per buffer) should be quite adequate in most cases. However, should it be necessary to store an even larger number of incoming messages, the input buffer can be increased. Valid values for parameter /I are 2, 4 or 8 (Kbytes).

Archiv/96A0016E01, V3.9, June 2014

Instead of the slash, '/', it is also possible to enter a dash '-' to identify the options. No distinction is made between upper case and lower case characters.

Multiple Easylon Interface cards installed

The network driver for the Easylon Interface cards only supports one interface card. If several of these cards are installed in the computer, the network driver must be installed an appropriate number of times with different addresses in the 'config.sys' file. If it is established during loading that another network driver has already been installed for the Easylon Interface, the copyright message will be suppressed.

2.3.6.2 Display of Network Drivers Installed in the Computer

The 'lpcdrv.exe' file can be called from the DOS command line in the same manner as any program, to show all the network drivers installed in the system, the appurtenant device names and their storage requirements.

/R The option /R additionally enables modification of the device number.

Example: lpcdrv -r13
changes the name LON1: to LON3:

If the first device number does not exist, or if the second number has already been assigned to another device driver, the message

Invalid or duplicate device name

will appear.

Renaming device names is not restricted to network interface devices defined by this Easylon Interface network driver and can also be applied from a Windows DOS box as a global function for the entire system, including 16-bit Windows applications.

3 Technical Description

This chapter describes the ISA-Bus interface card and three generations of the Easylon PC/104 Interface. The PC/104 cards can be identified by the type codes “LP4”, “LP42” and “LP43”. Wherever necessary these names are used to discern them.

3.1 Network Interface

The Easylon Interface cards are based on the NEURON 3150® Chip. Under MIP/P50 firmware the NEURON Chip is operated with up to 32 Kbytes ROM as program memory and 24 Kbytes SRAM as data memory. For the NSI firmware versions the memory is 48 Kbytes ROM and 9 Kbytes SRAM. It is connected to the PC bus in Slave_A mode.

For monitoring purposes, a reset flip-flop is additionally implemented on the interface card; a reset of the NEURON Chip can be identified by the PC via a status byte. The same mechanism is implemented for the interrupt flip-flop.

In order to visualize the status and to initiate the service function of the node, the service pin of the NEURON Chip is available in the front panel with service button and service LED (cf. Figure 1-2 to Figure 1-4) to activate the function.

3.2 ISA Bus Interface

The ISA bus interface has been implemented as an 8-bit I/O interface in accordance with "Personal Computer Bus Standard P996". Specialties of the PC/104 Interface are covered by the “PC/104 Specification, Version 2.3”.

The Easylon Interface cards are assigned four I/O addresses on the ISA bus. The base address of this I/O range is set via DIP switches (cf. Chapter 2.1 “Setting the I/O Addresses”). Two addresses are used for data communication between PC and network node, while two further addresses are available for status checks and to control the NEURON Chip, and the other for the watcher module.

Archiv/96A0016E01, V3.9, June 2014

3.3 Reset Procedure, System Control

Reset of the NEURON Chip of the Easylon Interface cards can be initiated from the computer, with a program-controlled function. After a system reset the NEURON Chip starts up automatically.

3.4 Block Diagram

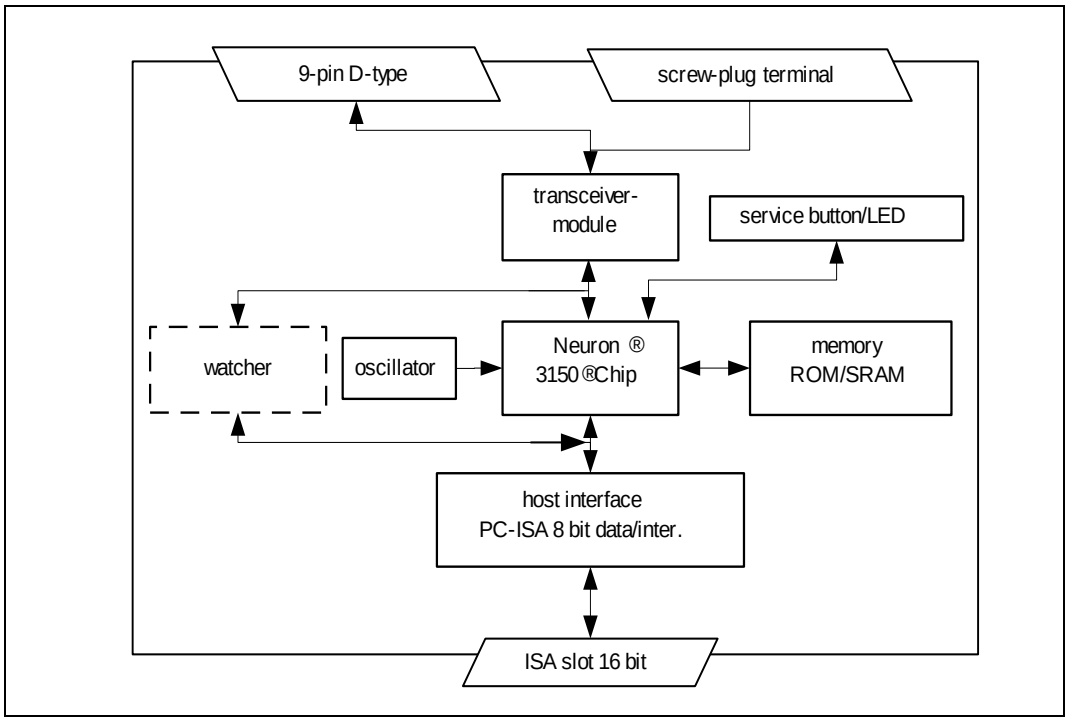


Figure 3-1 Block diagram Easylon ISA-Bus Interface

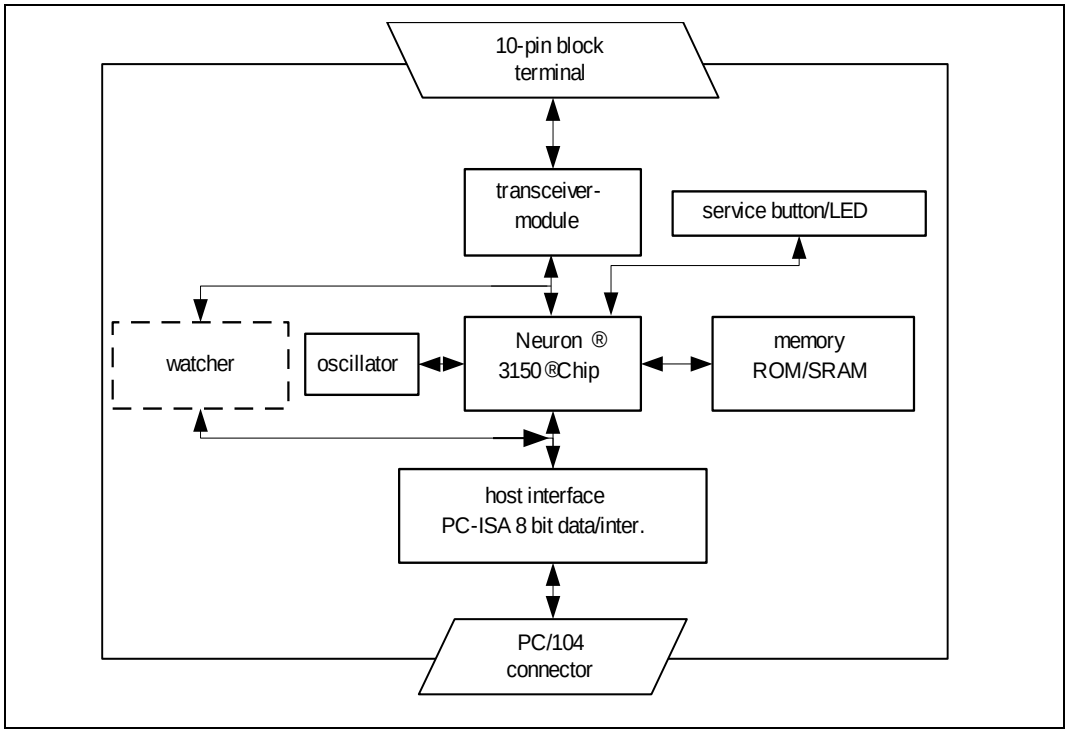


Figure 3-2 Block diagram Easylon PC/104 Interface

3.5 Connector Pin Assignments

NOTE For EMC reasons the cable used for network connection must be fitted with the ferrite core (see 2.3). This is independent of which connector on the board is used.

3.5.1 Easylon ISA-Bus Interface

The Easylon ISA-Bus Interface provides 9-pin D-type connectors (Figure 1-1, (4)) for network connection. In the TP/XF and FTT variants, the card is additionally equipped with a 2 pin screw-plug terminal (Figure 1-1, (3)), which can be used alternatively.

Connector type	Pin	Signal	Remark
9-pin D-type	1	Data	Data
	2	Data	Data
	3	—	Reserved
	4	—	Reserved
	5	—	Reserved
	6	—	Reserved
	7	—	Reserved
	8	—	Reserved
	9	—	Reserved
2-pin screw-plug Terminal	1	Data	Data
	2	Data	Data

Table 3-1 Connector pin assignments for TP/XF and FTT network connection of Easylon ISA-Bus Interface

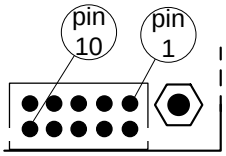
Connector type	Pin	Signal	Remark
9 pin D-type	1	—	Reserved
	2	—	Reserved
	3	DA-	Data
	4	SA-	RTS –
	5	—	Reserved
	6	+5 V	U+ supply voltage, electrically isolated
	7	0 V	U– supply voltage, electrically isolated
	8	DA+	Data
	9	SA+	RTS +

Table 3-2 Connector pin assignments for EIA-485 network connection of Easylon ISA-Bus Interface

Archiv/96A0016E01, V3.9, June 2014

3.5.2 Easylon PC/104 Interface

The Easylon PC/104 Interface has one 10-pin block terminal for LON network connection. (some customized version may have different connectors.) However there are differences with the card generations concerning the signals available.



Pin 1 position

Connector type	Pin	Signal	Remark
10-pin block	1	—	Reserved
	2	—	Reserved
	3	—	Reserved
	4	—	Reserved
	5	—	Reserved
	6	—	Reserved
	7	Data	LON A
	8	—	Reserved
	9	Data	LON B
	10	—	Reserved

Table 3-3 Connector pin assignment for “LP4” Easylon PC/104 Interface

Connector	Pin	Signal	Remark
10pin LON con-nector	1	option: SERVICE	external Service LED 3,3 V, 10 mA, High = ON
	2	—	Reserved
	3	option: RxLED	LON traffic Rx *
	4	—	Reserved
	5	option: TxLED	LON traffic Tx *
	6	—	Reserved
	7	Data	LON A
	8	—	Reserved
	9	Data	LON B
	10	Shield	Shield

Table 3-4 Connector pin assignment for “LP42” Easylon PC/104 Interface

Connector	Pin	Signal	Remark
10pin. LON Connector Figure 1-2 (4)	1	Data	LON A
	2	Data	LON A
	3	nc	frei
	4	nc	frei
	5	Data	LON B
	6	Data	LON B
	7	nc	frei
	8	nc	frei
	9	shield	Protective ground, **must be conneted**
	10	shield	

Table 3-5 Connector pin assignment for "LP43" Easylon PC/104 Interface

The LP43.FBB variant has a connector to lead out external signals, Further details cf. 3.7, „Connecting External LEDs ".

Connector	Pin	Signal	Remark
External signal connector Signal Figure 1-2 (8)	1	dig. GND	
	2	Service Taster	cf. chapter 3.7
	3	dig. GND	
	4	Service LED	cf. chapter 3.7
	5	dig. GND	
	6	RX LED	LON traffic, cf. chapter 3.7
	7	dig. GND	
	8	TX LED	LON traffic, cf. chapter 3.7
	9	dig. GND	
	10	VCC	5 V

Table 3-6 8 pin connector for external signals of LP43.FBB

3.6 Service LED

The service LED (Figure 1-1, (2) and Figure 1-4, (1)) signals the card status. Additional to the service LED signals defined by Echelon following status signals are defined:

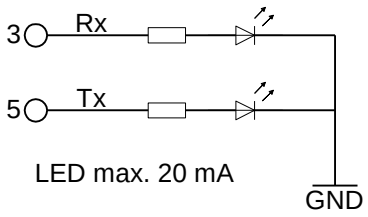
Service LED	Status	Remarks
Flash (1 Hz)	No driver installed or driver conflict.	Check the driver settings, IRQ- or address conflict to other cards?
Blink (1/2 Hz)	Driver installed, node is “unconfigured”[7].	Configure the node.
Permanently ON	Node is “applicationless” and “unconfigured”.	
Permanently OFF	Installation ok	Normal operation

Table 3-7: Service LED

3.7 Connecting External LEDs

This option is available with the LP42.xxx and LP43.xxx types of the Easylon PC/014 Interface. The connection type depends on the transceiver used on the card.

FFT-10



FTX

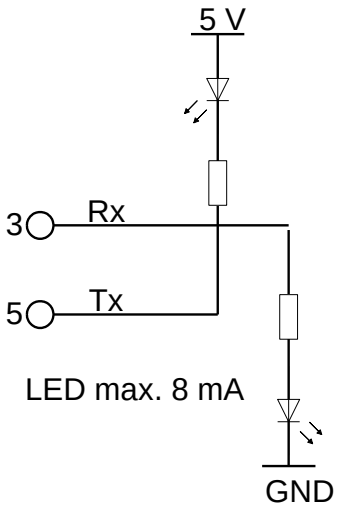
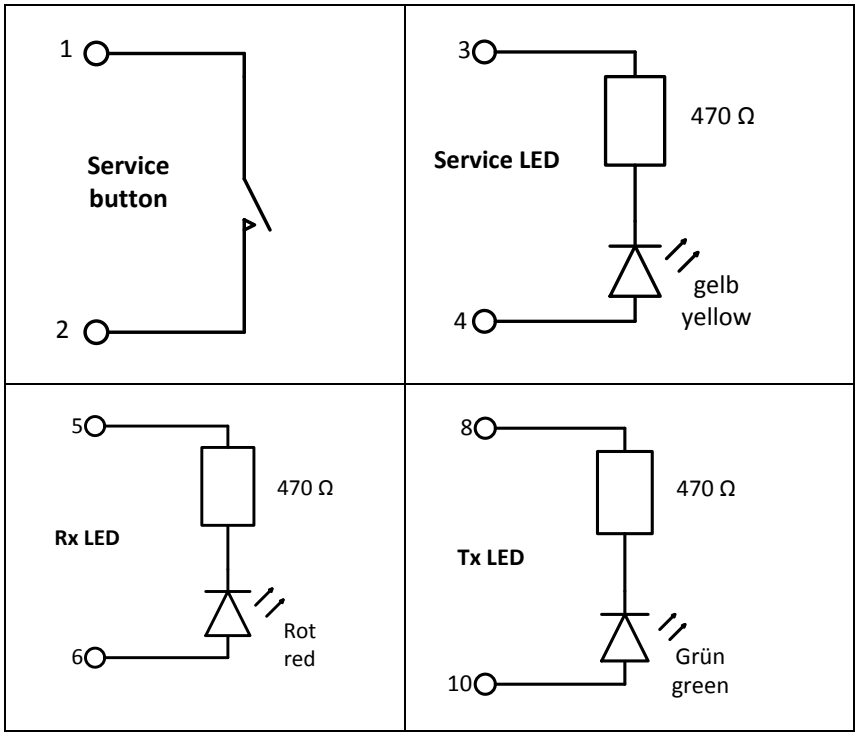


Figure 3-3 External LED connection for LP42

7 boards are delivered “unconfigured”



I_{LED} 8 mA max.

Figure 3-4 External LED connection for LP43

3.8 Technical Specification

3.8.1 General

Bus Interface	8 bit data (I/O) in accordance with Personal Computer Bus Standard P996 (PC/104 Specification, Version 2.3)
I/O addresses	4, settable via DIP switches
Control register	8 bit
Status register	n8 bit
Interrupts	3, 5, 7, 9, 10, 11, 12, 15 software selectable
CPU	NEURON 3150 Chip, 10 MHz LP43: 20 MHz
Coupling	parallel, Slave_A mode
Memory	
MIP/P50	ROM 32 Kbytes RAM 24 Kbytes
NSI	ROM 48,75 Kbytes RAM 9 Kbytes
Voltage supply	5 V, from PC

Power consumption typical 1.5 W

Temperature

operational 0 °C to +50 °C

non-operational -20 °C to +60 °C

Compatibility LonTalk, ISO/IEC 14908

Humidity according to DIN 40040 class F

3.8.2 Easylon ISA-Bus Interface

Dimensions 160 mm x 107 mm, for short 16-bit ISA slot

EMC EN50081-1
EN50082-1

Network interface

Order-Code*	Network interface type	Trans-mission rate	Network connector	Protection
P.P10001 P.P10011 P.P11001	TP/XF	78 kbps	9 pin D-type plus 2 pin screw-plug terminal	Sparc-gaps
P.P10002 P.P10012 P.P11002	TP/XF	1.25 Mbps	9 pin D-type plus 2 pin screw-plug terminal	Sparc-gaps
P.P10003 P.P10013 P.P11003	EIA-485, electr. isol.	39 kbps	9 pin D-type	secondary protec-tion by Zener di-odes
P.P10004 P.P10014 P.P11004	FTT	78 kbps	9 pin D-type plus 2 pin screw-plug terminal	Sparc-gaps

* cf. Table 1-1 for variant identifiers on the card.

3.8.3 Easylon PC/104 Interface

Dimensions 90.2 mm x 95.7 mm, for PC/104 computers

Temperature, extended range

operation -40 °C to +85 °C

storage -40 °C to +85 °C

EMC EN 55 022 A/B
EN 61 000-2

Compatibility EN 50 155 for versions with extended temperature range, must be verified with respect to target sys-tem

Network interface

Order-code*	Network interface type	Trans-mission rate	Network connector	Protection
P.P10101 P.P10111 P.P11101 P.P10601 P.P10611	TP/XF	78 kbps	IDC	Sparc gaps
P.P10102 P.P10112 P.P11102 P.P10602 P.P10612	TP/XF	1.25 Mbps	IDC	Sparc gaps
P.P10603 P.P10613	RS485 electr. isol	39 kbps	IDC	Zener diode
P.P10104 P.P10114 P.P11104 P.P10604 P.P10614	FTT	78 kbps	IDC	Sparc gaps
P.P10605 P.P10615	Direct Connect	1.25 Mbps	IDC	Diode
P.P10605 P.P10615	FTX	78 kbps	IDC	Sparc gaps
P.P73106	FT-X1	78 kbps	IDC, bent 90°	Isolation up to 1 KV sparc gaps common mode choke

* cf. Table 1-2 for variant identifiers on the card.

3.9 Electromagnetic Compatibility

The Easylon ISA-Bus Interface and the Easylon Interface cards are CE certified products and meet the intent of *Directive 2004/108 for Electromagnetic Compatibility*.

To ensure electromagnetic compatibility under operation in accordance with the above mentioned EEC directive the ferrite core enclosed in the delivery must be mounted around the interface cable to the LON network. The ferrite core must be mounted close to the connector on the board end side of the cable. The distance between the ferrite core and the connector on the board must not exceed 10 cm.

Archiv/96A0016E01, V3.9, June 2014

4 Programming Instructions

This chapter gives programming instructions to both the Easylon ISA-Bus Interface and Easylon PC/104 Interface. They are generally referred to as interface card or interface node.

4.1 LONWORKS Network Node

The Easylon Interface card is a network node in the LON network. It is operated under Echelon's Microprocessor Interface Program MIP/P50 or with NSI firmware using the NEURON 3150 Chip as communication processor. The appropriate external interface files (.xif) are on the installation disk. Which .xif-file is describing which interface card variant is explained in table 4.1 below.

Network Interface	Transmission rate	XIF -file
TP/XF	78 kbps	lolp072f.xif
TP/XF	1.25 Mbps	lolp073f.xif
RS485, optically isol.	39 kbps	lolp074f.xif
FTT	78 kbps	lolp075f.xif

Table 4-1 .xif files and interface card variants

4.1.1 Interface to the Network

The different variants of the network interface are each operated directly via the communication port (CP0...CP4) of the NEURON Chip, which is to be configured accordingly for the transmission process concerned.

4.1.2 Node CPU

The interface node is designed on the basis of the NEURON 3150 Chip. The standard clock pulse for the processor is 10 MHz. Under MIP/P50 firmware the processor is equipped with a 32 Kbytes ROM as program memory. A 24 Kbytes SRAM serves as the data memory. The memory for NSI firmware is 48.75 Kbytes ROM and 9 Kbytes SRAM.

The 11 I/O ports of the NEURON Chip are all used for parallel coupling with the ISA bus interface.

The status of the service pin of the NEURON Chip is indicated by a LED. The service function can be activated via the button (cf. Figure 1-1 to Figure 1-4).

Archiv/96A0016E01, V3.9, June 2014

4.1.2.1 Coupling NEURON Chip ‹–› ISA Bus Interface

The NEURON Chip is coupled in Slave_A parallel mode. The handshake bit defined by the NEURON Chip (NHS) to control the data flow can be checked via the status byte of the Easylon ISA-Bus Interface. Please refer to the NEURON 3150 Chip data book with regard to the data communication mechanisms in Slave_A mode.

4.1.2.2 Interrupt Function NEURON Chip –› ISA Bus

An interrupt flip-flop is set via write access to a defined memory address. This interrupt flip-flop is reset (acknowledged) by the host via accesses to an I/O address.

Read-back of the interrupt status by the NEURON Chip is not possible.

4.1.2.3 NEURON Chip Address Map

Address range			Module
$0000 … $7FFF	&	Read	ROM 32 Kbytes, program memory
$8000 … $DFFF	&	Read/Write	SRAM 24 Kbytes, data memory
$E000 … $E7FF	&	Write	Set interrupt flip-flop
$E800 … $FFFF			NEURON Chip internal

Table 4-2 NEURON Chip address map of MIP/P50 variant

Address range			Module
$0000 … $C2FF	&	Read	ROM 48.75 Kbytes, program memory
$C300 … $E6FF	&	Read/Write	SRAM 9 Kbytes, data memory
$E700 … $E7FF	&	Write	Set interrupt flip-flop
$E800 … $FFFF			NEURON Chip internal

Table 4-3 NEURON Chip address map of NSI variant

Note: Setting of the interrupt flip-flop is data-independent

4.2 Device Status

Applications have to take care of the status of the Easylon Interface card. As an example some parts of code are shown below. The structures used are taken from the so called HOST APPLICATION of the Echelon Corp. This application is available from the Echelon web site: www.echelon.com.

```
#pragma pack(1)
#define NM_update_domain  0x63
#define NM_set_node_mode  0x6C
#define SVC_request       0x60
#define niRESPONSE    0x16
#define niLOCAL           0x22
#define niRESET           0x50
#define LDV_OK        0

typedef struct {
    BYTE cmq;       // cmd[7..4]                    queue[3..0]
    BYTE len;
    BYTE svc_tag;   // 0[7] Service[6..5] auth[4]   tag[3..0]
    BYTE flags;     // prio path cplcode[5..4] expl altp pool resp
    BYTE data_len;
    BYTE format;    // rcv: domain[7] flex[6]
    union {
        struct {
            BYTE dom_node;      // domain[7] node/memb[6..0]
            BYTE rpt_retry;     // rpt_timer[7..4]      retry[3..0]
            BYTE tx_timer;      //                      tx_timer[3..0]
            BYTE dnet_grp;      // destination subnet or group
            BYTE nid[6];        // NEURON ID
        } send;
            struct {
            BYTE snet;          // source subnet
            BYTE snode;         // source node
            BYTE dnet_grp;      // destination subnet or group
            BYTE dnode_nid[7];  // destination node or NEURON ID
        } rcv;
        struct {
            BYTE snet;          // source subnet
            BYTE snode;         // source node
            BYTE dnet;          // destination subnet
            BYTE dnode;         // destination node
            BYTE group;
            BYTE member;
            BYTE reserved[4];
        } resp;
    } adr;
    BYTE code;                   // message code or selector MSB
    BYTE data[239];
} ExpAppBuf;

ExpAppBuf msg_out;    // Explicit message buffer for outgoing messages
ExpAppBuf msg_in;     // Explicit message buffer for incoming messages
ExpAppBuf msg_rsp;    // Explicit message buffer for response messages
```

```
int ni_handle;
BYTE my_domain[15] =
    {0,0,0,0,0,0, 0x01, 0xC0, 0, 0xFF,0xFF,0xFF,0xFF,0xFF,0xFF};

int send_local( int len ) {
    int ldv_err;
    msg_out.cmq = niLOCAL;
    msg_out.svc_tag = SVC_request;
    msg_out.flags = 8;
    msg_out.len = len + 15;
    msg_out.data_len = len + 1;
    if( ldv_write( &msg_out, len + 17 ) ) return(0);
    while( 1 ) {
        ldv_err = ldv_read( &msg_in, 256 );
        if( ldv_err == LDV_OK ) {
            if(msg_in.cmq == niRESET) return(0);        // Local reset
            if(msg_in.cmq == niRESPONSE) {
                memcpy(&msg_rsp, &msg_in, msg_in.len + 2);
                return(1);                               // Ok
            }
        }
    }
    return(0);
}

int set_config_online() {
    msg_out.code = NM_update_domain;
    msg_out.data[0] = 0;                          // Domain index 0
    memcpy( &msg_out.data[1], &my_domain, 15 ); // Subnet 1, Node 64
    if( !send_local(16)) return(0);

    msg_out.code = NM_set_node_mode;
    msg_out.data[0] = 3;                          // Change state
    msg_out.data[1] = 4;                          // Configured online
    if( !send_local(2)) return(0);
    return(1);                                    // Success
}
```

4.3 ISA Bus Interface

The ISA bus interface is implemented as an 8-bit I/O interface in accordance with "Personal Computer Bus Standard P996, Draft D2.01".

This module is assigned four I/O addresses:

For timing reasons, the data-flow control information (NEURON Chip and data driver) is not derived from the read and write strobes of the ISA bus, but is implemented on an address-related basis. Two I/O addresses are thus reserved for this purpose.

The PC employs one 8-bit control and status register each in the ISA bus interface for control and status-checking of the NEURON Chip respectively.

4.3.1 I/O-Address Map

Base address +	Device
$000 & Read/Write	Data port watcher
$001 & Write	Write control byte
$001 & Read	Read status byte
$002 & Read	Read data, NEURON Chip
$002 & Write	Reset interrupt flip-flop from NEURON Chip
$003 & Write	Write data, NEURON Chip
$003 & Read	not assigned

Table 4-4 I/O address map, ISA bus

Attention **Special addresses for LP42, LP43**

Base address +	Device
$002 & Write	Reset interrupt flip-flop from NEURON Chip. For NSI write “0”, for MIP write “1”.

Signals for resetting the Interrupt Flip-Flop

As the bit combination at the same time sets the firmware mode (MIP or NSI), a special byte must be written when resetting the interrupt flip-flop.

Data bit	Description
D7...D2	reserved, must be written as 0
D1, D0	......00: NSI, ...01: MIP, ...10: reserved (currently acts like NSI) ......11: EEBLANK

Archiv/96A0016E01, V3.9, June 2014

4.3.1.1 Signal Assignments Control Byte

Data bit	Signal	Description
D7	EIN	Interrupt enable, see Table 4.6
D6	IL2	Interrupt select, see Table 4.6
D5	IL1	Interrupt select, see Table 4.6
D4	IL0	Interrupt select, see Table 4.6
D3	NSERV	NEURON Chip service pin, high active
D2	/NCF	NEURON Chip clear reset flip-flop, low active
D1	WTCRES	Watcher reset, high active
D0	NRES	NEURON Chip reset, high active

Table 4-5 Signal assignments control byte

The control byte of the Interface card is reset at Power On reset; the reset signal to the NEURON Chip and the service pin are not activated.

Interrupts	D7	D6	D5	D4
Disabled	0	X	X	X
IRQ 3	1	0	0	0
IRQ 5	1	0	0	1
IRQ 7	1	0	1	0
IRQ 9	1	0	1	1
IRQ 10	1	1	0	0
IRQ 11	1	1	0	1
IRQ 12	1	1	1	0
IRQ 15	1	1	1	1

Table 4-6 Bit D7...D4 of control byte

Archiv/96A0016E01, V3.9, June 2014

4.3.1.2 Signal Assignments Status Byte

Data bit	Signal	Description
D7	EIN	Read back D7 of Control register
D6	IL2	Read back D6 of Control register
D5	IL1	Read back D5 of Control register
D4	IL0	Read back D4 of Control register
D3	/NINT	Status of NEURON Chip interrupt flip-flop, low active
D2	/NRF	Status of NEURON Chip reset flip-flop, low active
D1	/WTCHS	Watcher handshake, low active
D0	/NHS	NEURON Chip handshake, low active

Table 4-7 Signal assignments status byte

4.3.2 Reset Procedure

The NEURON Chip starts up automatically when the power of the PC is switched on.

During operation, a hardware reset of the NEURON Chip can be initiated via a control bit (NRES).

The NEURON Chip is able to initiate a reset independently during operation. An additional reset flip-flop is implemented on the NEURON Chip, to enable the PC to identify such a reset. The status of this flip-flop (/NRF) can be checked via the status byte of the Interface card. The flip-flop is reset and deactivated via the control bit (/NCF). When /NCF is hold 'low', the reset flip-flop is deactivated (/NCF = '0').

4.4 Windows CE – Application Interface

4.4.1 CreateFile

Opens a LON device.

Syntax:

```
ni_handle = CreateFile(szDevName,
GENERIC_READ|GENERIC_WRITE, 0, NULL, OPEN_EXISTING, 0,
NULL);
```

Parameter	Type	Description
SzDevName	TCHAR*	Device name, e.g. TEXT("LON1:")

Return value	Type	Description
ni_handle	HANDLE	file handle of the LON device or `INVALID_HANDLE_VALUE`

4.4.2 CloseHandle

Closes a LON device.

Syntax:
```
CloseHandle(ni_handle);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device that should be closed

4.4.3 ReadFile

This synchronous function reads a telegram according to the application layer format. Synchronous means the function returns only if the NEURON received the telegram.

Syntax:
```
ReadFile(ni_handle, pMsg, len, &rLen, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
pMsg	void*	pointer to an „explicit message buffer“
len	DWORD	length of the buffer [bytes]
rlen	DWORD	length of the received telegram [bytes]

4.4.4 WriteFile

Writes a telegram according to the application layer format. This function returns immediately.

Syntax:
```
WriteFile(ni_handle, pMsg, len, &rLen, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
pMsg	void*	pointer to an „explicit message buffer“
len	DWORD	length of the buffer [bytes]
rlen	DWORD	length of the telegram to be send [bytes]

Remark: The telegram according to the application layer format contains a length information of the buffer itself. That is why we ignore the parameter `len` in the use of function `ReadFile()` and `WriteFile()`. Note: Use the maximum length (256 bytes) of the buffer while reading a telegram.

4.4.5 GetVersion

Returns the version number of the driver as unicode string, e.g. `TEXT("LPCDRV v1.00")`.

Syntax:

```
#define IOCTL_GETVERSION  0x43504C00
result = DeviceIoControl(ni_handle, IOCTL_GETVERSION,
szInfo, sizeof(szInfo), & NULL, 0, BytesReturned, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
szInfo	TCHAR*	Buffer for version string
BytesReturned	DWORD	length of the string [bytes] = (number of characters + 1) * 2

Return value	Type	Description
Result	BOOL	FALSE if buffer is too small, else TRUE

4.4.6 Watcher

Depending on the buffer contents, several watcher commands (including down-load of firmware for watcher module) are initiated.

Syntax:

```
#define IOCTL_WATCHER     0x43504C01
result = DeviceIoControl(ni_handle, IOCTL_WATCHER,
inbuffer, sizeof(inbuffer), outbuffer, sizeof(outbuffer),
&BytesReturned, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
outbuffer	BYTE*	pointer to the buffer that contains the commands and data to send to the watcher
inbuffer	BYTE*	pointer to the buffer that contains the data sent by the watcher
BytesReturned	DWORD	number of bytes which are received by the watcher. OK, if BytesReturned >= 2

Return value	Type	Description
Result	BOOL	FALSE, if no Watcher was found, else TRUE

Archiv/96A0016E01, V3.9, June 2014

4.4.7 ReadFile with Timeout

Reads a telegram according to the application layer format. The `Timeout` parameter determines the functions behavior acts while the receive buffer is empty:

Timeout = 0:	function returns immediately
Timeout = n:	function waits n milliseconds to receive a telegram.
Timeout = INFINITE:	function works as synchronous function, see also function ReadFile.

Syntax:

```
#define IOCTL_READ    0x43504C02
result = DeviceIoControl(ni_handle, IOCTL_READ, pMsg, len,
&timeout, 4, &rLen, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
timeout	DWORD	Timeout [Milliseconds]
pMsg	void*	pointer to an „explicit message buffer“
len	DWORD	length of the buffers [bytes]

Return value	Type	Description
Result	BOOL	TRUE, if telegram was received was FALSE at timeout

Remark: Undefined IOCTL-Codes will return FALSE.

5 List of Figures

6 List of Tables

Archiv/96A0016E01, V3.9, June 2014

Archiv/96A0016E01, V3.9, June 2014

7 Index

Archiv/96A0016E01, V3.9, June 2014