USB Socket Interface
User Manual

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: LPM3/UserDoc/LPM3_Manual-EN-v2.2.docx,
Version v2.2, 06/30/2014

This manual …

… provides you with all the information which you will require to use the Easylon® USB Socket Interface.

However, this manual will neither explain aspects of Echelon's® LONWORKS® technology, nor Echelon's Microprocessor Interface Program (MIP) used on this network interface card. The drivers of the USB Socket Interface have been developed in compliance with the driver specifications of the Echelon Corporation. Details of these are as well not described in this documentation. For further information on the LONWORKS technology please refer to the extensive documentation provided by Echelon.

After a general presentation of the Easylon USB Socket Interface in Chapter 1, Chapter 2 describes the necessary steps to install the module.

Chapter 3 gives the technical specifications of the device and Chapter 4 provides some programming instruction for operation under Windows CE. Tips and tricks concerning the operation can be found in Chapter 5.

This documentation is subject to changes without notice. Gesytec assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

Gesytec shall have no liability or responsibility to the original purchaser or any other person or entity with respect to any claim, loss, liability, or damage caused or alleged to be caused directly or indirectly by any Gesytec product or the accompanying documentation.

Contents

1 Product Information

This manual describes the Easylon USB Socket Interface

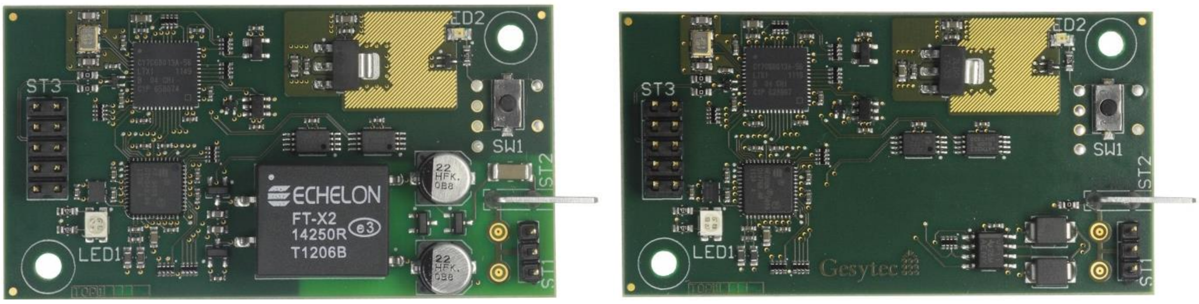


Figure 1-1 Easylon USB Socket Interface[1] FT-X2 and EIA-485

1.1 Variants

The following variants of the Easylon USB Socket Interface are described in this documentation.

Order Code	Network Interface Type	Neuron Firmware	Remark
P.P10503-3	EIA-485	MIP	
P.P20503-3	EIA-485	MIP	extended temperature
P.P10506-3	FTX	MIP	
P.P20506-3	FTX	MIP	extended temperature

Table 1-1 Variants and order-codes of Easylon USB Socket Interfaces

1.2 Scope of Delivery

- Easylon USB Socket Interface module with Echelon‘s MIP firmware
- Technical short information
- Installation and documentation CD with
  - 32/64 bit driver for Windows[2] XP / Vista / 7 / 8 /
    Server 2003 / 2008 / 2008R2 / 2012
  - Easylon RNI Software for remote LONWORKS access
  - EasyCheck diagnosis utility for Easylon interfaces

[1] Modification of connectors on request

[2] A Linux driver is available in source code on demand

- Documentation in Adobe Acrobat .PDF format
- Sample design for a carrier board (Gerber and Step files)

1.3 Overview

The Easylon USB Socket Interface realizes a LON-USB connection as a plug-in module, to be integrated into OEM devices. USB connection to the CPU board is made by a 10pin connector designed according the ASUS standard. Power supply uses this connector as well.

As an OEM module a certain flexibility with respect to customer specific requirements is observed, e.g. with respect to the connector types or positions. The respective module may therefore be different from the description in this documentation.

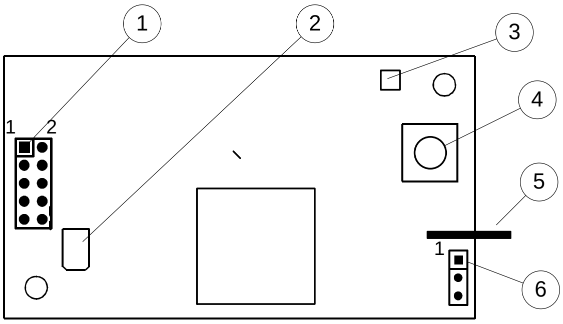
An Evaluation Kit available to the Easylon USB Socket Interface allows easy access to connections and signals of the board.

The module is available in different transceiver variants all with MIP firmware. Additionally to the FTX transceiver there is EIA-485. Furthermore there are variants for extended temperature range. Service button and ~LED are implemented.

Note Due to the usage of the Neuron 5000 with MIP firmware this module is not suited for LNS based applications.

The driver for the Easylon USB Socket Interface is compliant with Echelon’s driver interface. Applications using the driver interface directly can use the interface without problems. The Easylon USB Socket Interface is compatible with the Easylon OPC Server and Gesytec’s WLDV32.DLL.

The Easylon USB Socket Interface is a so called high speed USB device according to USB standard 2.0, however compatible with USB 1.1. The communication between the Neuron chip and USB is handled by a micro controller. The firmware for this micro controller is downloaded automatically, when the PC is started.



(1) USB connector pin 1, 1/10 inch pitch
(2) Status/error LED (green/red)
(3) Service LED (yellow)
(4) Service pin push button
(5) GND connector
(6) LonWorks connector pin 1

Figure 1-2 Module components shown for FTX variant

Service LED

The service LED (Figure 1-2, (3)) signals the card status. The following signals are defined the service LED:

Service LED	Status	Remarks
Flash (1 Hz)	No Neuron communication	Error
Blink (1/2 Hz)	Driver installed, node is "unconfigured"[3]	Configure the node.
Permanently ON	Node is „applicationless" and „unconfigured".	
Permanently OFF	Installation ok or USB not connected or driver not loaded	Normal operation check USB side check Windows device manager for driver

Table 1-2 Service LED

[3] device is delivered "unconfigured"

2 Installation

Please check the delivered items. You must find the Easylon USB Socket Interface and an installation CD, containing drivers and this documentation.

2.1 Hardware Installation

Please refer also to the manual describing the device into which you want to insert the Easylon USB Socket Interface. Turn off power, open the device and plug the USB module into a suitable USB socket. Please observer the following chapter “Pin Assignment”. Restart the PC after the module has been installed and insert the Drivers & Documentation CD in order to get the appropriate driver (cf. chapter 2.2 Driver Installation).

2.1.1 Pin Assignment

PIN	Description
1	+5 Volt
2	+5 Volt
3	USB -
4	-
5	USB +
6	-
7	GND
8	GND
9	-
10	-

Table 2-1 Pin assignment of 10 pin USB connector

PIN	MNEMO	Description
1	RT +	LON data +
2	GND	ground
3	RT -	LON data –

Table 2-2 Pin assignment of 3pin LON connector

Pin 1 position cf. Figure 1-2

2.2 Driver Installation

Drivers for different operating systems are available for the Easylon USB Socket Interface. Currently these are Windows 2000, XP, Vista, 7 and 8 and the Windows Server OS 2003, 2008, 2008 R2 and 2012. The drivers support both, the 32 and the 64 bit version of these operating systems. Latest driver versions you can download via the Easylon Support pages of our web site: www.gesytec.com.
A Linux driver is available in source code on request.

Installation for different operating systems is described in the following sections.

Windows operating systems	section 2.2.1
Windows CE	section 2.2.4
16-Bit driver under 32-bit Windows	section 2.2.2

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

2.2.1 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 XP onwards.

The setup program is using the same WDM driver (Windows Driver Model) for all operating systems.

Note: For installation you can either use the Windows assistant or the program FastUpd.exe for manual installation, which is much more straightforward (cf. chapter 2.2.1.2).
The latter is especially helpful if you are running Windows 7 and later or have to install several instances of the driver.

2.2.1.1 Installation using the Windows Assistant

After the Easylon Interface has been mounted Windows will recognize the new device and start the hardware assistant.

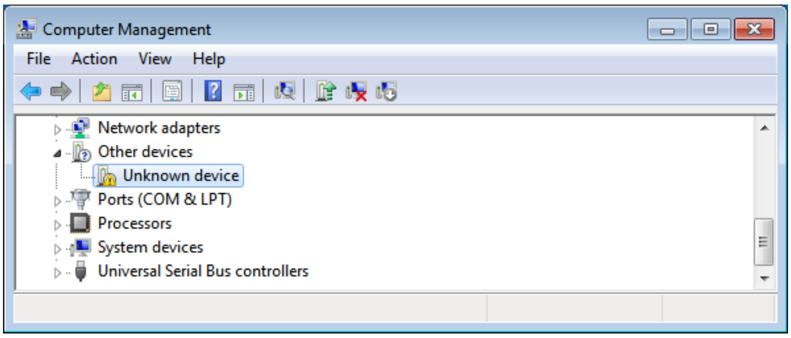
If Windows should not find the driver on the “Drivers and Documentation” CD or the driver should be elsewhere, please select the appropriate drive and select the setup file “lonusb.inf” and the driver “Gesytec LONUSB FX2 Driver” from the “LonUsb“ directory.

Windows 7

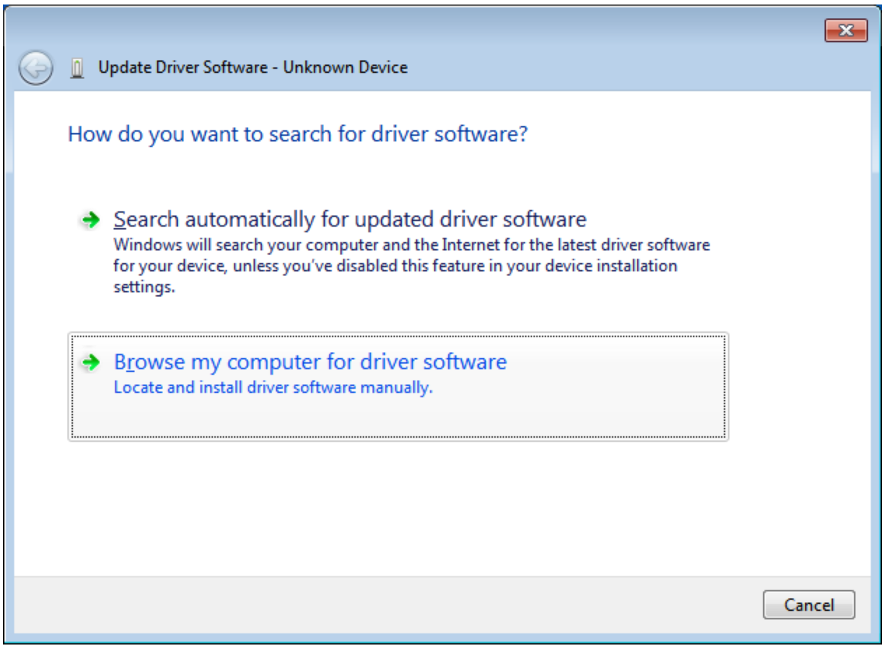
Windows 7 systems directly start looking for a driver at Windows Update and therefore ignore the CD. Consequently the installation using the assistant will fail and manual settings are required.

You can either follow the steps described in 2.2.1.2 “Manual Installation and Update” or proceed as follows:

Open the Device Manager (e.g. via the control panel).



Right click on the entry for the unknown device and select “Update Drivers”.


Update Driver Software - Unknown Device

How do you want to search for driver software?

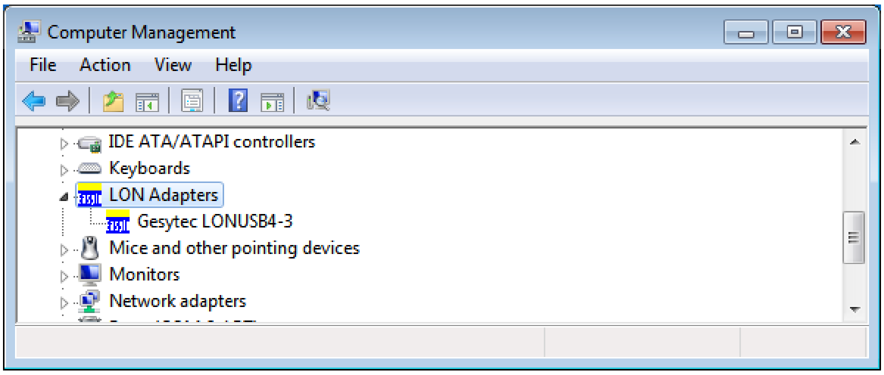
Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

Browse my computer for driver software
Locate and install driver software manually.

Select “Browse my computer ….” and indicate the drive with the “Drivers & Documentation” CD. Finally give permission for the Gesytec driver setup.

Final Steps

After successful completion of the installation the device manager will show the interface under „LON Adapters“.



Here you will find a „Gesytec LONUSB x-y...“ entry, with x designating the number of the USB host controller and y the port. If external hubs have been cascaded the respective port numbers are given as well.

If, after the installation the green LED does not blink an error has occurred in the Easylon USB Socket Interface installation. In that case, please restart the PC.

During the installation and at each Neuron reset the red LED is shortly flashing.

The device is now ready to access the LONWORKS network.

2.2.1.2 Manual Installation and Update

The easiest way to install the driver is to ignore the hardware assistant and run

FastUpd.exe

from the “Driver/LonUsb” folder of the CD-ROM.

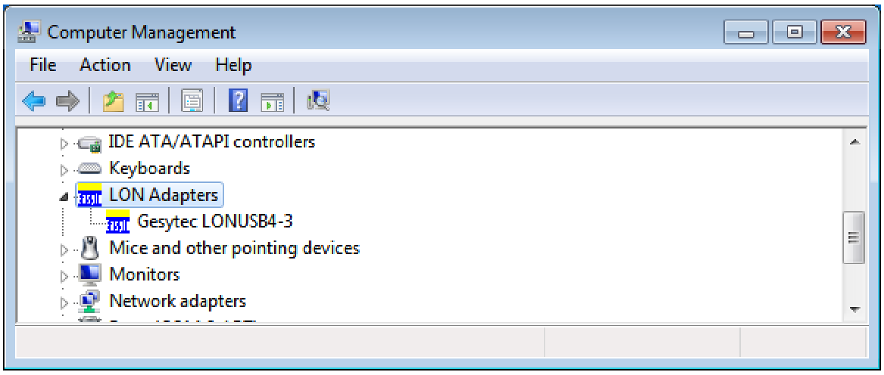
If the “Drivers & Documentation” CD interface has opened in your browser you may access the driver setup as well via “Products” “Easylon USB Socket Interface” and selecting the button for the operating system.

The same program you will use to update an existing driver.

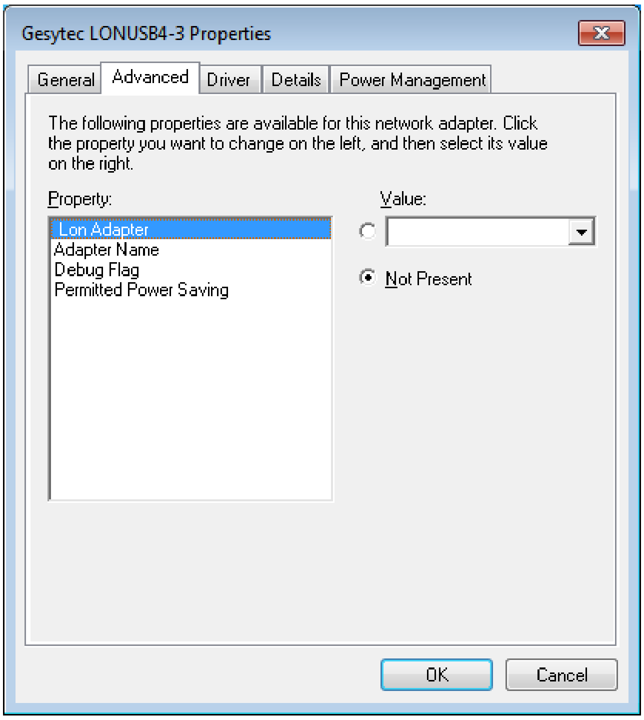
A new version will be installed on the PC within a few seconds. In order to update the firmware in the device as well, you must disconnect the USB Socket Interface and reconnect it again.

2.2.1.3 Settings

There are further settings available for the Easylon USB Socket Interface which may be helpful in certain operating conditions. They can be found in the Universal Serial Bus Controller section of the device manager. Select the properties of the desired device.



The „Advanced Properties“ offer the following settings:



Gesytec LONUSB4-3 Properties

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:
- Lon Adapter
- Adapter Name
- Debug Flag
- Permitted Power Saving

Value:
○
● Not Present

Lon Adapter

This will assign a name „LON1“ ... „LON9“ to the LON USB adapter, which certain application will require. Remember that the name must not be in use by any other device driver. In case of a name conflict the device cannot be started. (Code 10).

Adapter Name

Alternatively an arbitrary name can be assigned to the adapter (e.g. floor 7). If both „Lon Adapter“ and „Adapter Name“ are assigned to the same device only the entry for „Lon Adapter“ will be used.

Debug Flag

The value comprises a DWORD in hexadecimal notation of different flags for debug purposes. Usually it is set to 0 (not existing). Setting the single bits will turn on special debug features. In the current driver versions bits 0, 1 and 3 are used.

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

Bit 1: LON telegrams at the interface from and to the USB bus are shown in debug output.

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

Note The debug output for instance can be displayed using the DebugView program, which is freely available at www.sysinternals.com.

Permitted Power Saving
Usually the LON USB adapter allows a standby mode with applications running (Standby). At certain conditions however, (e.g. LON USB using an external hub under Windows 2000) the current supply to the LON USB adapter will be shortly interrupted during return from the standby mode by the external hub. Under such conditions a standby mode must be turned off (None).

2.2.2 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 –Llonusb1-2

The 32 bit LON device used is specified by the optional –L or /L parameter:

/L*name*

name =

lonusb1-2 for device LONUSB at USB host controller 1 and with port number 2 at USB root. If several hubs have been cascaded the respective port numbers have to be provided as well.

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

The 16 bit LON device used is specified by the following optional parameter:

/D*n*

with *n* = 1...9 for LON1 to LON9

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

2.2.3 EasyCheck – Quick Interface Diagnosis

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.

The program “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.

2.2.4 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/xx/LonUsb.

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.

Foe correct operation he 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\Default\LonUsb]
     "DLL"="lonusb.dll"
     "Prefix"="LON"
     "DebugFlag"=dword:0
     "ReadTimeout"=dword:FFFFFFFF
```

3 Technical Specifications

CPU	EIA-485 version FTX version	Neuron 5000, Neuron FT5000
Clock	80 MHz	
USB Interface		
Type	USB slave interface accord. to high speed USB standard 2.0	
Connector	10 pin 2 row edge connector	
LONWORKS Interface		
Transceiver	FT-X2, electrically isolated EIA-485, not electrically isolated, 1.25 Mbps max.	
Connector	3 pin edge connector	
Power Supply		
Power Supply	5 V DC +-10%, externally via USB connector	
Power Consumption	< 100 mA, typically	
Operating Conditions		
Temperature		
Operation	0 ºC – +70 ºC	
extended range	-40 ºC – +85 ºC	
Storage	-40 ºC – +85ºC	
Humidity	90%, no condensation	
Display and Operation	Neuron service push button Neuron service LED (yellow) status LED (green) error LED (red)	

Dimensions

Board 69.34 x 36.96 [mm]

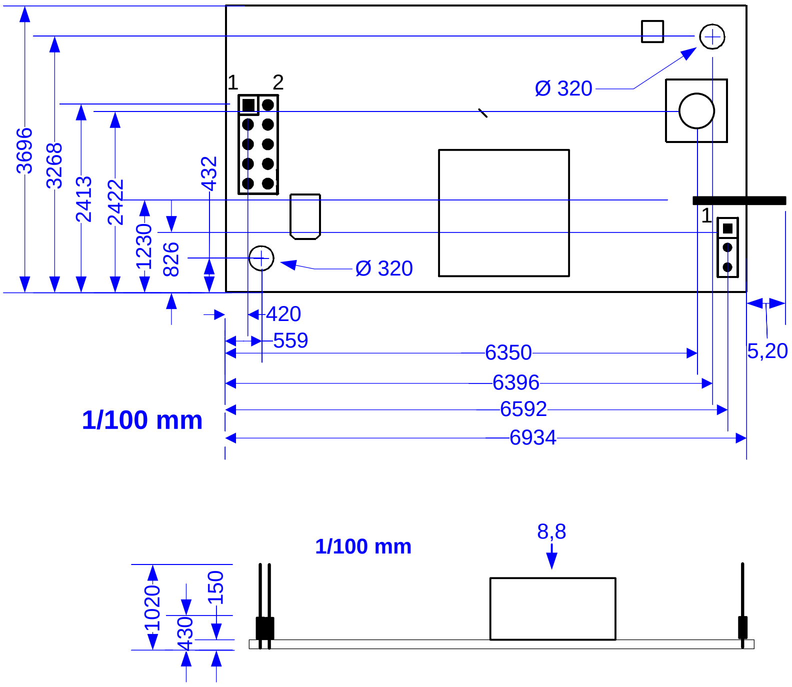


Figure 3-1 Dimensions FTX version

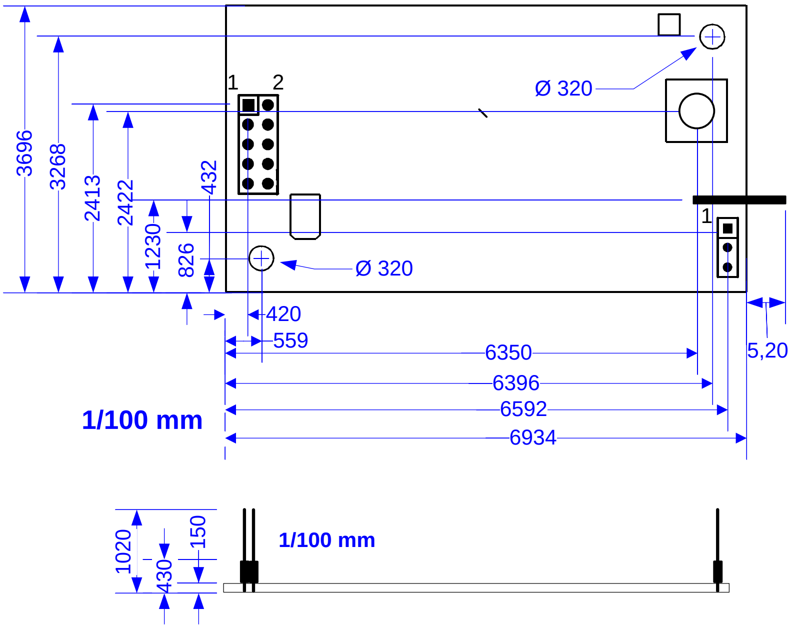


Figure 3-2 Dimensions EIA-485 version

4 Programming Instructions

4.1 Windows CE Application Interface

Note: Some of the functions described below are marked “obsolete”. These functions and control codes are referenced her only for compatibility with older versions of LPCDRV/LG2DRV and should not be used for development of new software.

4.1.1 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`

4.1.2 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

4.1.3 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 or the handle is closed.

The timeout of this blocking call can be changed via registry or via DeviceIo-Control. A timeout value of 0 means, that this function returns immediately, if no data are available.

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]

4.1.4 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]

Note: 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.

4.1.5 GetVersion

Returns the version number of the driver as Unicode string, e.g. `TEXT("Easylon LonUsb Version 1.00 for WinCE from 11/05/2002").`

Syntax:

```
#define IOCTL_LPCDRV_GET_VERSION \
     CTL_CODE( FILE_DEVICE_LPCDRV, 0x900, \
     METHOD_BUFFERED, FILE_READ_ACCESS )

#define IOCTL_GETVERSION  0x43504C01           //obsolete
result = DeviceIoControl(ni_handle,
IOCTL_LPCDRV_GET_VERSION,
     NULL, 0,  szVersion, sizeof(szVersion),
     BytesReturned, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
szVersion	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

4.1.6 ReadFile with Timeout

Reads a telegram according to the application layer format. The `Timeout` parameter determines the functions behavior 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_LPCDRV_READ_WAIT \
     CTL_CODE( FILE_DEVICE_LPCDRV, 0x908, \
     METHOD_BUFFERED, (FILE_READ_DATA | FILE_WRITE_DATA) )
result = DeviceIoControl(ni_handle, IOCTL_LPCDRV_READ_WAIT,
     &timeout, 4, pMsg, len,
     &rLen, NULL);

#define IOCTL_READ0x43504C02                   // obsolete
result = DeviceIoControl(ni_handle, IOCTL_READ,
     pMsg, len, &timeout, 4,
     &rLen, NULL);
```

Note: Using IOCTL_READ the Parameters lpInBuffer and lpOutBuffer as well as nInBufferSize and nOutBufferSize are permuted as defined in the API Reference of DeviceIoControl.

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 FALSE at timeout

4.1.7 Set Timeout for ReadFile

Reads a telegram according to the application layer format. The `Timeout` parameter determines the functions behavior 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_LPCDRV_SET_READ_TIMEOUT \
     CTL_CODE( FILE_DEVICE_LPCDRV, 0x909, \
     METHOD_BUFFERED, FILE_WRITE_DATA)
```

```
result = DeviceIoControl(ni_handle, IOCTL_LPCDRV_READ_WAIT,
      &timeout, 4, NULL, 0,
      &rLen, NULL);
```

Parameter	Type	Description
ni_handle	HANDLE	file handle of the LON device
timeout	DWORD	Timeout [Milliseconds]
Return value	Type	Description
Result	BOOL	TRUE, if timeout was stored, FALSE if an error has occurred

Note: Undefined IOCTL-Codes will return FALSE and set LastError to ERROR_NOT_SUPPORTED.

4.1.8 Registry entries for Easylon USB Interface

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

DebugFlag

The value comprises a DWORD in hexadecimal notation of different flags for debug purposes. Usually it is set to 0 (not existing). Setting the single bits will turn on special debug features. In the current driver versions bits 0 and 1 are used.

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

Bit 1: LON telegrams at the interface from and to the USB bus are shown in debug output.

ReadTimeout

The value (in milliseconds) comprises a DWORD in hexadecimal notation to affect the behavior of ReadFile().

A value of INFINITE (= 0xffffffff) makes ReadFile() a blocking call. This is the default behavior, if no parameter is given (like lpcdrv, lg2drv).

A timeout value of 0 means, that this function returns immediately, if no data are available.

5 Tips and Tricks

5.1 Hot Plugging

The Easylon USB Socket Interface may be connected and disconnected, when the PC is already running. Windows recognizes plugging the device in and starts the driver automatically. You should not remove the device, when an application is using it.

5.2 Using an USB Hub

Of course the Easylon USB Socket Interface can be used with an USB hub. If there are a couple of USB devices active, the communication between PC and Easylon USB Socket Interface may be slowed down.

5.3 Standby Mode of PC

A PC with connected Easylon USB Socket Interface may be set to standby mode, because the device will be powered during standby. However, if the device is used with an external USB hub under Windows 2000, it was observed that, at returning from the standby mode, some hubs shortly interrupt the power supply to the Easylon USB Interface. This USB hub behavior will reinitialize the device and active applications, using the Easylon USB Socket Interface before entering standby mode, are not able to communicate with device any longer.

In such configurations please refer to section Settings and set “Permitted Power Saving" to “None” to disable the standby mode. The LON USB driver will then inhibit the standby mode with applications running.

5.4 Hibernation Mode of PC

The Easylon USB Socket Interface does not support the hibernation mode. When the PC enters hibernation mode, the USB will not be powered any longer. As this would lead to a loss of the Neuron Chip settings the LON USB driver will inhibit Windows from turning into the hibernation mode with applications running.

5.5 Registry Key

The driver of the Easylon USB Socket Interface makes an entry in the registry database for each found device, according to Echelon's guidelines. You will find this entry at:

\\HKEY_LOCAL_MACHINE\Software\LonWorks\DeviceDrivers.

For each Easylon USB Socket Interface you will find a key with the device name (Gesytec LONUSBx-y...) and a character value with the driver name.

6 List of Figures

7 List of Tables

8 Index