Operating Manual

USB Control Interface LUCI-10

COMPATIBLE WITH NATIONAL INSTRUMENTS LabVIEW™

Rev.4

FEMTO®

Thank you…

for choosing a FEMTO® product. We are confident you will be happy with the LUCI-10 USB Interface. We would appreciate your feedback, and any suggestions to further improve the product are highly welcome.

Berlin, August 2017 FEMTO Messtechnik Team

Legal Information

The LUCI-10 interface including any drivers, application programs and related software components (herein referred to as “SOFTWARE”) were developed and tested with the utmost care. Nevertheless, the SOFTWARE is provided “AS IS” and with all faults. FEMTO Messtechnik GmbH disclaims all warranties, whether express or implied, including without limitation warranties that the SOFTWARE is free of defects, merchantable, fit for a particular purpose and non-infringing third party rights. FEMTO Messtechnik GmbH does not warrant that the functions contained in the SOFTWARE will meet your requirements or that the operation of the SOFTWARE will be uninterrupted or error-free. FEMTO Messtechnik GmbH is not obligated to provide any updates to the SOFTWARE. FEMTO Messtechnik GmbH will not be liable to you for any consequential, special, incidental, indirect or exemplary damages of any kind arising out of or in any way relating to the use or inability to use the SOFTWARE, including without limitation damages for loss of goodwill, work stoppage, lost profits, loss of data, and computer failure or malfunction, even if advised of the possibility of such damages and regardless of the theory (contract, tort or otherwise) upon which such claim is based.

The SOFTWARE is licensed, not sold. This license does not transmit any intellectual rights on the SOFTWARE. FEMTO Messtechnik GmbH hereby reserves all intellectual property rights in the SOFTWARE. You may not remove or alter any trademark, logo, copyright or other proprietary notice in or on the SOFTWARE. All title and copyrights in and to the SOFTWARE (including but not limited to any images, photographs, animations, video, audio, music, text and “applets” incorporated into the SOFTWARE), the accompanying printed materials, and any copies of the SOFTWARE, are owned by FEMTO Messtechnik GmbH. The SOFTWARE is protected by copyright laws and international treaty provisions. You may not copy the printed materials accompanying the SOFTWARE.

R4, August 2017

FEMTO Messtechnik GmbH
Klosterstr. 64
D-10179 Berlin, Germany

Tel.: +49 (0)30-280 4711-0
Fax: +49 (0)30-280 4711-11
E-mail: info@femto.de
http://www.femto.de

Table of Contents

1 Introduction

Important information:

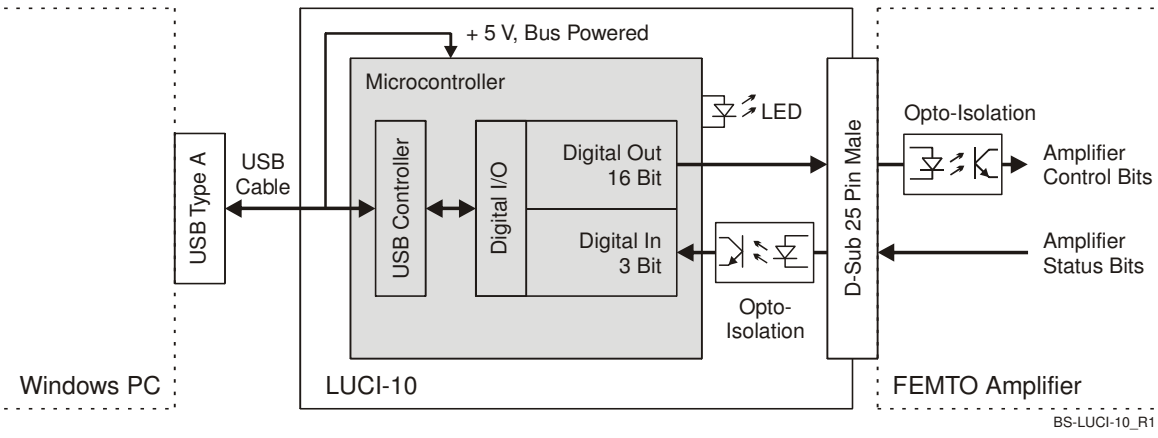
- Please install the software package and driver before connecting the LUCI-10 interface to the USB port. If a previous software version is already installed on your PC, please remove it before installing the current version. The necessary steps for uninstalling the software are described in chapter 2.6.
- For the use of the complete new software package a LabVIEW version 2012 or higher is required. An older version of the program is no longer being supported. If you want to control the new OE-300 photoreceiver by software but reuse your current LUCI-10 application programs, the installation of the OE-300 GUI only is recommended (to be found in the installation directory OE-300 Executable).
- For the installation of the LabVIEW drivers the VI package manager from JKI (version 2016 or higher) is required. The current version is free downloadable under: https://vipm.jki.net/.
- From this new LUCI-10 version 1.3 a windows 64-Bit environment is supported. 64-bit processors can handle arrays of larger sizes and can support more memory capacity. When you combine a 64-bit processor with a 64-bit operating system, you can take advantage of increased memory and data storage for your applications. The LabView 64-Bit installer is available as free download on the National Instruments homepage.
- If you want to reuse your current applications programs but with the new LUCI-10 software version please follow the software adaption description in chapter 2.5.

1.1 Product Description

The *Laboratory USB Control Interface* LUCI-10 was developed for simple and intuitive remote control of FEMTO amplifiers and photoreceivers with D-Sub socket from a standard Windows® PC with USB port. As LUCI-10 is powered through the USB bus, no additional power supply is necessary for its operation.

The interface electronics controlling the data transfer between PC and FEMTO module is all integrated inside the D-Sub connector. By using opto-couplers in the LUCI-10 interface and the FEMTO modules, isolation between USB bus and analog signal path is established. Thus, an impairment of the signal quality through ground loops and noise pick-up is effectively suppressed.

Block Diagram of the LUCI-10 Interface



For remote control of FEMTO modules, LUCI-10 features 16 digital outputs. Additionally, 3 digital inputs are provided to read status information like *OVERLOAD* and *UNLOCKED* from the FEMTO module.

The extensive software package delivered on CD with the LUCI-10 hardware contains a device driver (dynamic link library – .dll) for use on a Windows PC. Furthermore, programs with a graphical user interface (GUIs) are included allowing an intuitive approach to the remote control of FEMTO units. These GUIs are provided not only as executable programs (.exe) but also as virtual instruments (VIs) for use in LabVIEW™. The underlying LabVIEW libraries can be used for creating own projects. Alternatively, the device driver can be also integrated in programming languages like C/C++ or LabWindows™/CVI™. The source code of a 64-Bit C/C++ example program is also included on the CD (see Chapter 3.4).

For complex setups, up to 40 LUCI-10 interfaces can be connected to one single PC. For easy identification each LUCI-10 features a green LED which can be turned on and off by software commands.

1.2 Connectors and Pinning

The LUCI-10 has a type A USB plug for connecting to a standard PC or USB hub. The interface is USB bus powered and therefore it does not require a separate power supply for operation. At the other end it comes with a 25 pin male D-Sub plug mating the interface port of the FEMTO amplifiers and photoreceivers. Jack screws are provided to firmly attach LUCI-10 to the FEMTO module.

Pinning of the 25 pin D-Sub Plug:

Pin	LUCI-10	Pin	LUCI-10
1	NC	14	Digital OUT, Low Byte, Bit 4
2	NC	15	Digital OUT, Low Byte, Bit 5
3	GND (IN)	16	Digital OUT, Low Byte, Bit 6
4	NC	17	Digital OUT, Low Byte, Bit 7, MSB
5	Digital IN	18	Digital OUT, High Byte, Bit 0, LSB
6	Digital IN	19	Digital OUT, High Byte, Bit 1
7	Digital IN	20	Digital OUT, High Byte, Bit 2
8	NC	21	Digital OUT, High Byte, Bit 3
9	GND (OUT)	22	Digital OUT, High Byte, Bit 4
10	Digital OUT, Low Byte, Bit 0, LSB	23	Digital OUT, High Byte, Bit 5
11	Digital OUT, Low Byte, Bit 1	24	Digital OUT, High Byte, Bit 6
12	Digital OUT, Low Byte, Bit 2	25	Digital OUT, High Byte, Bit 7, MSB
13	Digital OUT, Low Byte, Bit 3		

Legend: NC = not connected, do not use
GND (IN) = ground connection for input pins
GND (OUT) = ground connection for output pins
LSB = least significant bit
MSB = most significant bit

The 16 output pins are split up in a Low Byte and in a High Byte of 8 Bit each. These bits can be set by the software and correspond to a certain setting of the attached FEMTO module.

Examples of Control Bytes:

Low byte = pins 10-17								
D-Sub pin	17	16	15	14	13	12	11	10
	0	**0**	**1**	**0**	**1**	**1**	**0**	**0**
Bit priority	MSB							LSB

The binary number is 00101100_{BIN}, equivalent to $2C_{HEX}$ and 44_{DEC}

High byte = pins 18-25								
D-Sub pin	25	24	23	22	21	20	19	18
	0	**0**	**0**	**0**	**1**	**0**	**0**	**1**
Bit priority	MSB							LSB

The binary number is 00001001_{BIN}, equivalent to 09_{HEX} and 9_{DEC}

2 Installation

Important information: Please install the software package and driver before connecting the LUCI-10 interface to the USB port. If a previous software version is already installed on your PC, please remove it before installing the current version. The necessary steps for uninstalling the software are described in chapter 2.5.

The following chapters describe the software installation, verification and uninstall process on a PC running Windows 7. On PCs running a different Windows operating system the procedures might be slightly different. Currently, LUCI-10 does not support operating systems like Linux or Mac OS.

The new LUCI-10 LabVIEW drivers are available as VI-Package (*.vip). After the installation you can open the package with the free VI package manager from 2016 (VIPM https://vipm.jki.net/, from JKI).

If the package manager is already available on your PC, a query if you want to install the LabVIEW drivers directly occurs while installation of the software.

2.1 Summary of the Software Package

The table below contains a summary of the software components delivered on the driver CD. Further information on the various software products can be found in the following sections.

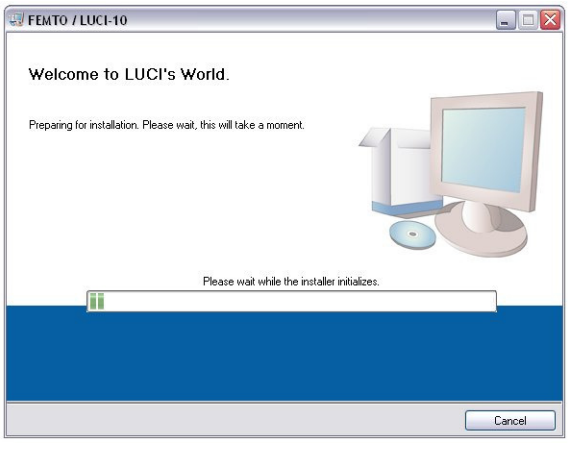
Software	Software product	Note	Further information
Installation program	Setup.exe	Starts the installation process for copying the software components to the host PC.	chapter 2.2
Device driver	LUCI_10.dll	Device driver for Windows operating System (XP SP 3 or higher) incl. header and lib file.	chapter 4.1
Application program	LUCI_Control.exe LUCI-10_Control.exe	Programs for control and functional test of the LUCI-10 hardware for use in Windows.	chapters 3.3 and 3.4
Graphical User Interface	GUIs (.exe)	Executable Windows programs for the intuitive control of FEMTO amplifiers and photoreceivers with 25 pin D-Sub socket.	chapter 3.1
Graphical User	GUIs (VI)	Extensive LabVIEW program package	chapter 4.3

Interface		(front panel and block diagrams) for the control of FEMTO amplifiers and photoreceivers with 25 pin D-Sub socket.	
LabVIEW Library	Control Palette	High level VIs for the simple and quick programming of own LabVIEW code allowing remote control of FEMTO modules.	chapter 4.2.2
LabVIEW Library	Advanced Palette	Low level VIs for programming LUCI-10 on a bit and byte level in a LabVIEW development environment.	chapter 4.2.1
Choose_Device_Enum.ctl	Type Definition	Device selection based on type definition provided by this driver.	chapter 2.5
LabVIEW Run-Time Engine	LabVIEW Run-Time Engine	Platform for use of the executable (.exe) GUI applications.	page 10
Documentation	.pdf files	Datasheet and operating manual	

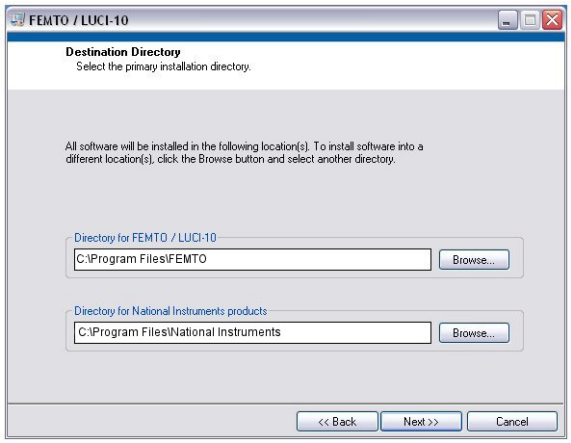
2.2 Installation of the Device Driver and Application Programs

The installation routine usually starts automatically after inserting the driver CD in a CD or DVD drive. If the installation routine does not start automatically, please manually start the *setup.exe* file in the root directory of the driver CD. Please install the software package and drivers before connecting the LUCI-10 interface to the USB port.

Note: For installing the device driver and application software it might be necessary to have administrator rights and to accept a Windows confirmation prompt. Please install the software using an account with administrator rights.

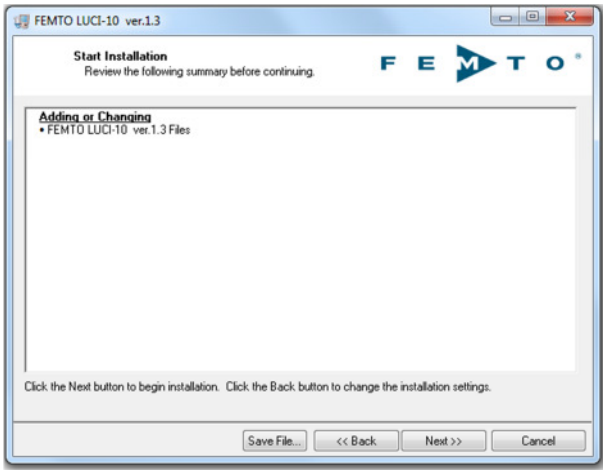


Following the start screen of the installation routine you may select the destination directory for the FEMTO LUCI-10 software package. The default directory is *C:\Program Files\FEMTO\LUCI-10.* It is recommended to use this directory for correct implementing of the LUCI-10 libraries into a LabVIEW development platform. The Advanced and Control palettes will be copied to the directory *C:\Program Files\National Instruments\LabVIEW\user.lib\FEMTO*.

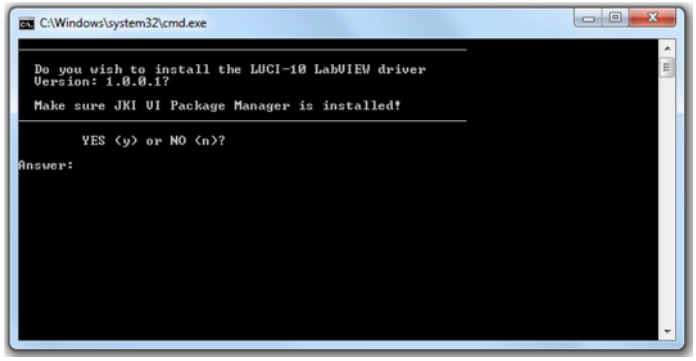


After selecting the installation directories, please accept the license agreement of FEMTO Messtechnik GmbH and National Instruments™.

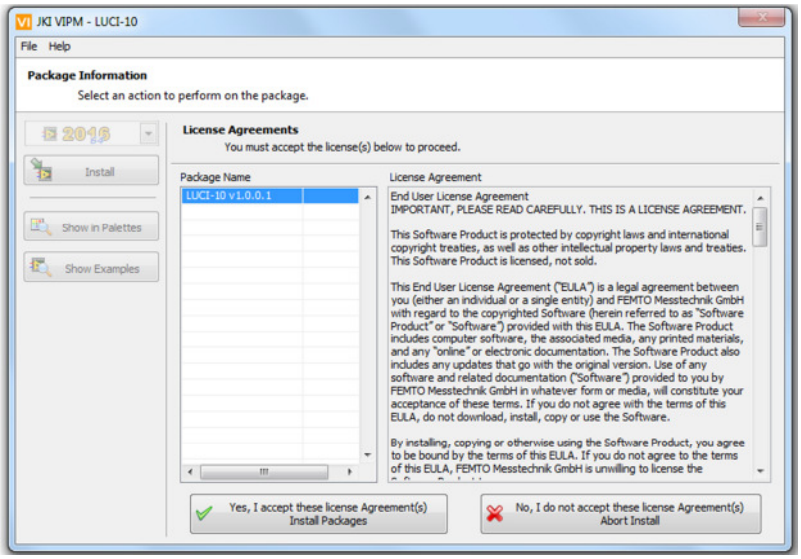
You can start the installation process by clicking the *Next* button in the installation summary window.



Then the question if you want to install the LUCI-10 LabVIEW driver directly occurs. Therefore the JKI VI package manager from 2016 must be installed on your PC. Please confirm this by entering YES (Y).



Again you have to confirm the license agreement



In the JKI VPM menu you can choose the convenient LabVIEW environment (in the picture below LabVIEW 2016, 64 bit).

Note: The NI™ LabVIEW Run-Time Engine will be copied from the CD if no LabVIEW development environment is detected on the host PC during the installation routine. The LabVIEW Run-Time Engine is required for using the executable application programs for remote control of the FEMTO modules (GUIs in .exe format, see chapter 3.2). This installation step will be skipped if LabVIEW or the LabVIEW Run-Time Engine is already installed on the host PC.
Please note that a full license of NI LabVIEW is required for creating your own LabVIEW projects. This license is NOT part of the LUCI-10 software package. Please contact National Instruments for further information on LabVIEW.

2.3 Verification of the Software Installation

Please follow these steps to verify that the LUCI-10 software and the LabVIEW libraries were correctly installed on your PC:

Software Package:
In Windows you will find the application programs in the menu *Start >> Programs >> FEMTO LUCI-10*. The software files (incl. documentation, GUIs etc.) are stored in the directory *C:\Program Files\FEMTO\LUCI-10* of the hard disk.

Device Driver:
The device driver LUCI_10.dll is copied to the Windows system folder *C:\Windows* during installation. When creating your own control software please use a copy of the LUCI_10.dll device driver.

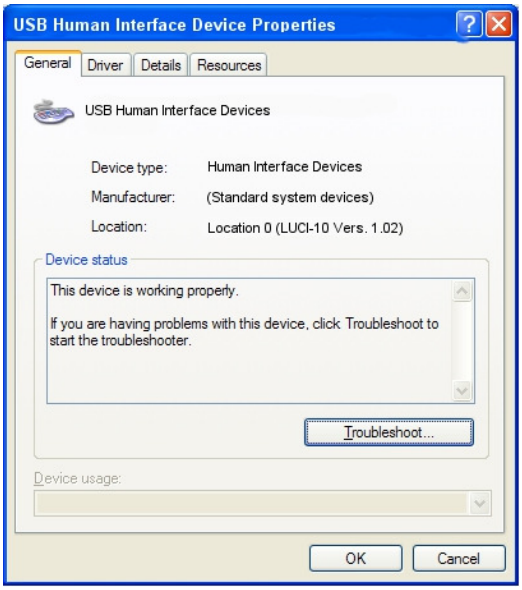
LabVIEW VIs and Palettes:
The Advanced and Control palettes are copied to the current LabVIEW directory. To verify the installation, please start LabVIEW and check the new VIs in the user.lib folder.
Sometimes LabVIEW does not recognize the LUCI-10 VIs properly. In these cases it is necessary to start a mass compilation at *Tools >> Advanced >> Mass Compile* and manually indicate the path to the FEMTO LUCI-10 libraries.
Note: It is always recommended to do a mass compilation before using the LabVIEW libraries and VIs.

2.4 Installation and Verification of the Hardware

After successful installation of the software package, the LUCI-10 hardware can be connected to a USB port or hub through the flexible 1.8 m cable. A passive extension up to 3 m is possible. For larger distances an active USB hub or USB to Ethernet adapter should be used.
The driver software supports up to 40 LUCI-10 units per PC. After connecting LUCI-10 to the PC for the first time, Windows will display a message that new hardware was found. LUCI-10 is installed as Human Interface Device (HID). Therefore, it is not necessary to disconnect it from Windows before removing it from the USB bus.

Please follow these steps to verify that the LUCI-10 hardware was properly recognized by the operating system:

- Open System Properties by pressing [Windows key] + [Pause/Break key].
- Click on the Hardware tab and open the Device Manager.
- Click on the category Human-Interface Devices.
- Right-click on the device and display the LUCI-10 properties.



- A quick functional check can be performed by starting the application LUCI-Control.exe (see chapter 3.3 for details).

After successfully connecting the LUCI-10 interface to the PC, a FEMTO amplifier or photoreceiver can be connected to the other end. LUCI-10 supports any standard FEMTO module with 25 pin D-Sub socket except for the logarithmic amplifier HLVA-100. The FEMTO module should be off at the time it is connected to the LUCI-10 interface. Please firmly fasten the LUCI-10 jack screws to establish a mechanically and electrically stable connection.
For proper remote control of a FEMTO amplifier or photoreceiver it is necessary to set the manual switches of the module to a defined position. Otherwise the communication between PC and module may not work properly. Further information on the required switch settings can be found in the datasheets of the FEMTO modules or in the GUI applications when clicking the Info button (see chapter 3.1).

Example: Current Amplifier DLPCA-200

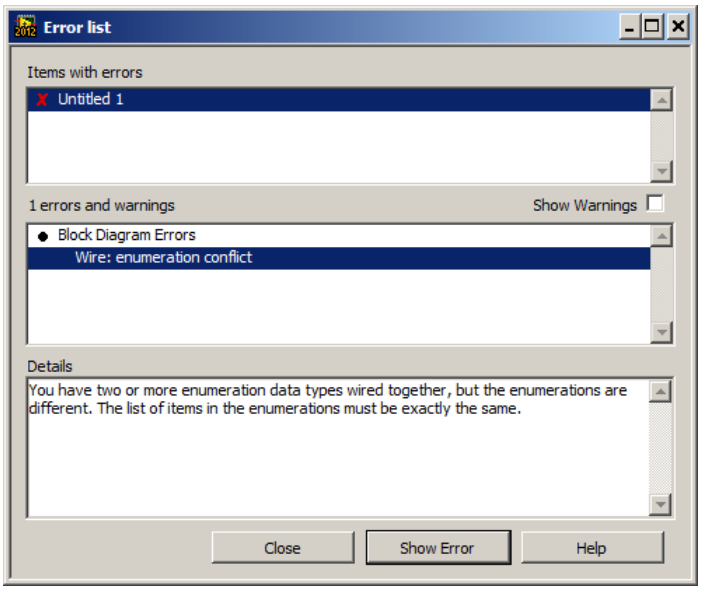
For proper remote control of the DLPCA-200 the local switches must be set to the following positions:

- Rotary switch to the *Remote* position
- Toggle switch to *AC*
- Toggle switch to *H* (High-Speed)

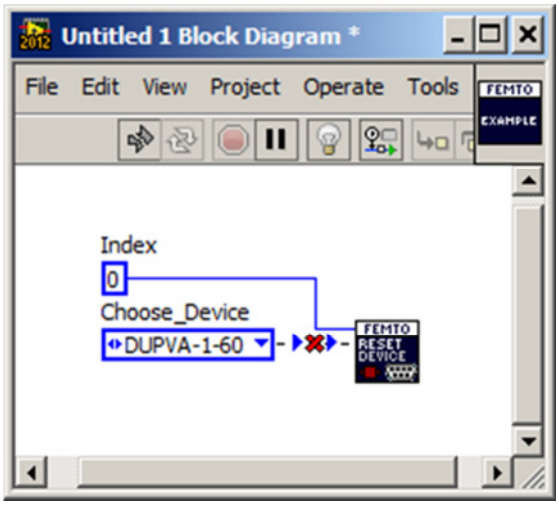
The functions of the toggle switches *FBW / 10 Hz* and *BIAS / GND* are not remote controllable.

2.5 Further notes on the use of current programs with the new LUCI-10 software environment

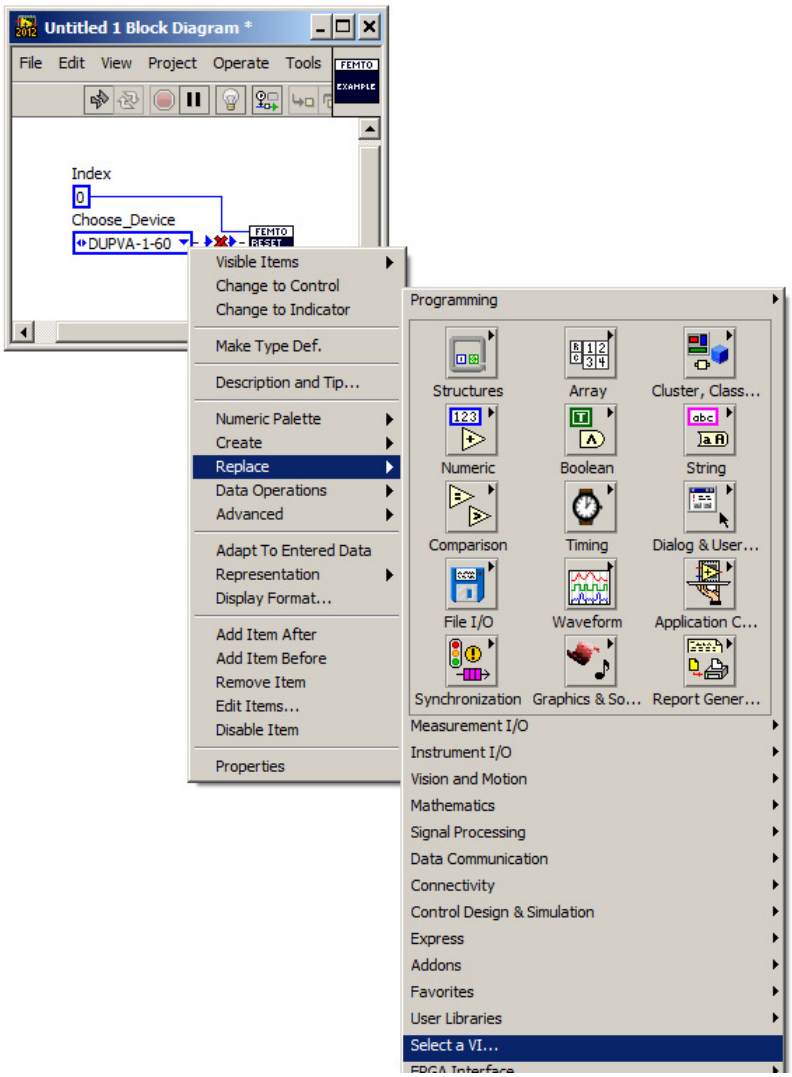
If you use your older application programs with the new LUCI-10 software package the following enumeration conflict will occur:

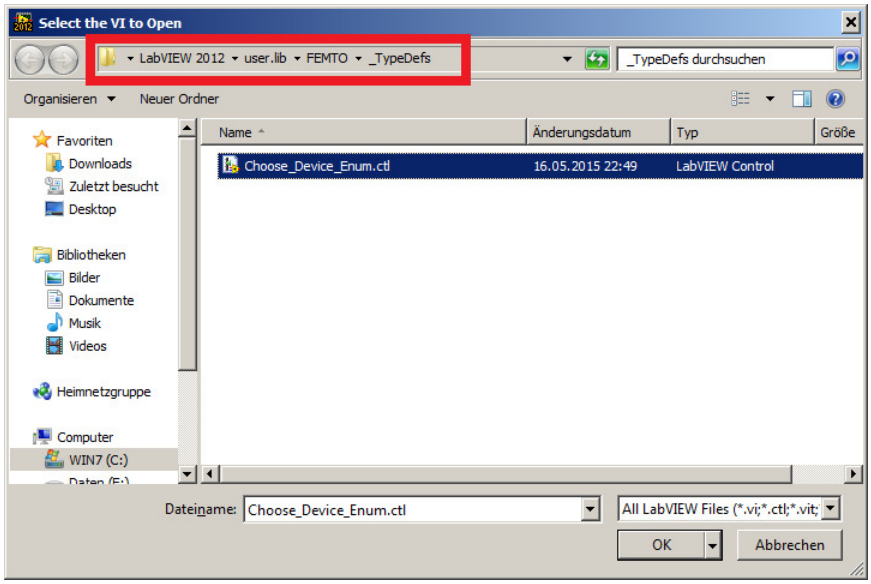


Example using the DUPVA-1-60:



Please choose the data type which you want to replace in your LabView program and replace it like shown below:

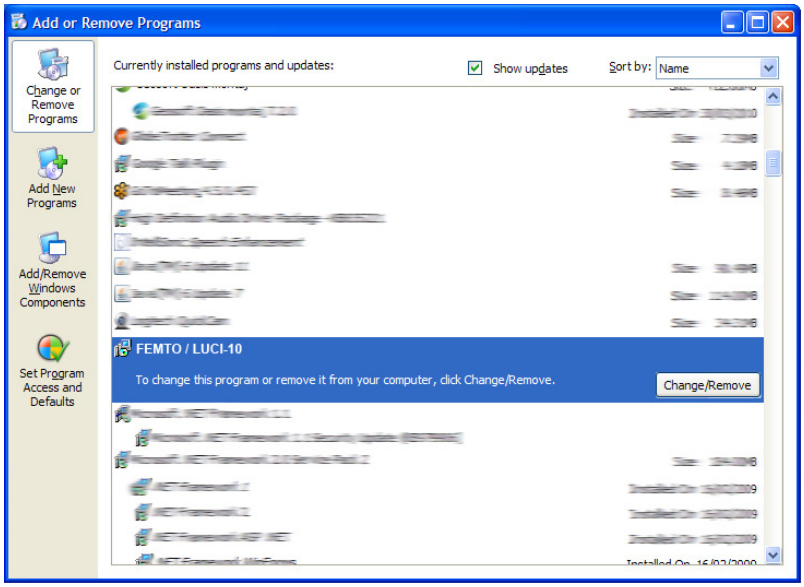

Untitled 1 Block Diagram *
File Edit View Project Operate Tools
Index
0
Choose_Device
DUPVA-1-60
Visible Items
Change to Control
Change to Indicator
Make Type Def.
Description and Tip...
Numeric Palette
Create
Replace
Data Operations
Advanced
Adapt To Entered Data
Representation
Display Format...
Add Item After
Add Item Before
Remove Item
Edit Items...
Disable Item
Properties
Programming
Structures
Array
Cluster, Class...
Numeric
Boolean
String
Comparison
Timing
Dialog & User...
File I/O
Waveform
Application C...
Synchronization
Graphics & So...
Report Gener...
Measurement I/O
Instrument I/O
Vision and Motion
Mathematics
Signal Processing
Data Communication
Connectivity
Control Design & Simulation
Express
Addons
Favorites
User Libraries
Select a VI...


Select the VI to Open
LabVIEW 2012 ▸ user.lib ▸ FEMTO ▸ _TypeDefs
_TypeDefs durchsuchen
Organisieren
Neuer Ordner
Favoriten
Downloads
Zuletzt besucht
Desktop
Bibliotheken
Bilder
Dokumente
Musik
Videos
Heimnetzgruppe
Computer
WIN7 (C:)
Name
Änderungsdatum
Typ
Größe
Choose_Device_Enum.ctl
16.05.2015 22:49
LabVIEW Control
Dateiname:
Choose_Device_Enum.ctl
All LabVIEW Files (*.vi;*.ctl;*.vit;
OK
Abbrechen

2.6 How to Uninstall the Software

Please follow these steps to fully remove the software package including drivers and VIs from your PC:

- In the Windows Control Panel open the *Add or Remove Programs* menu.
- Select *FEMTO / LUCI-10* and click *Remove*. The complete software package will be removed.



- For also removing the NI LabVIEW Run-Time Engine, select the item *Software of National Instruments* and click the *Remove* button.

3 Working with the GUI Application Programs

3.1 Graphical User Interface

A Graphical User Interface (GUI) application is a software program allowing the user to interact with a device with graphical elements rather than text commands. The accessible functions are usually controlled through a pointing device like a PC mouse.

3.2 GUIs for FEMTO Amplifiers as Executable Applications (.exe)

As part of the software package, programs with graphical user interface will be installed to the directory *C:\Program Files\FEMTO\LUCI-10\Software\GUIs*. These GUIs can be used to easily remote control various FEMTO amplifiers and photoreceivers. For using these files in .exe format, the LabVIEW Run-Time Engine is required which is installed on the host PC during the installation process. A full LabVIEW development environment is not required for using these GUIs.
In the menu *Start >> Programs >> FEMTO LUCI-10 >> GUIs* select the application that is mating your FEMTO module. Once selected with a mouse click the program code is executed immediately. By clicking the red *STOP* button in the lower right corner the program can be halted. After clicking the *Start* arrow in the menu bar the program is restarted.

Example: FEMTO GUI of the Current Amplifier DHPCA-100

The following picture shows the front panel of the DHPCA-100 GUI application. In the smaller right part of the window, information about the LUCI-10 configuration is displayed. The larger left part of the window contains the control elements for remote control of the FEMTO amplifier.

Control elements for FEMTO amplifier.

Information about LUCI-10 configuration.

Information about the LUCI-10 Interface Configuration
(right part of the front panel window):

In the upper right field, all LUCI-10 interfaces presently connected to the PC are displayed. To select a certain interface, click on the corresponding entry in the monitor area. For verifying your selection click on the *Check LUCI* button in the lower right corner to turn the green LED on the LUCI-10 interface hardware alternately on and off.

When clicking the *Product Info* button the current firmware version of the interface will be displayed. The red *STOP* button can be used to stop the execution of the program.

Controlling the FEMTO Amplifier
(left part of the front panel window):

For proper remote control of a FEMTO amplifier or photoreceiver, it is necessary to set the manual switches of the module to a defined position. Otherwise the communication between PC and module may not work properly. When clicking the info button  a new window is opened showing the necessary switch settings for the selected FEMTO module. Further information on these settings can be also found in the datasheets of the individual FEMTO modules.

After starting the application, the amplifier settings can be remotely controlled by using your PC mouse to change the front panel buttons of the GUI. The Overload display in the lower left corner of the front panel corresponds to the Overload LED on the FEMTO module. The LED will be lit if the amplifier is saturated.

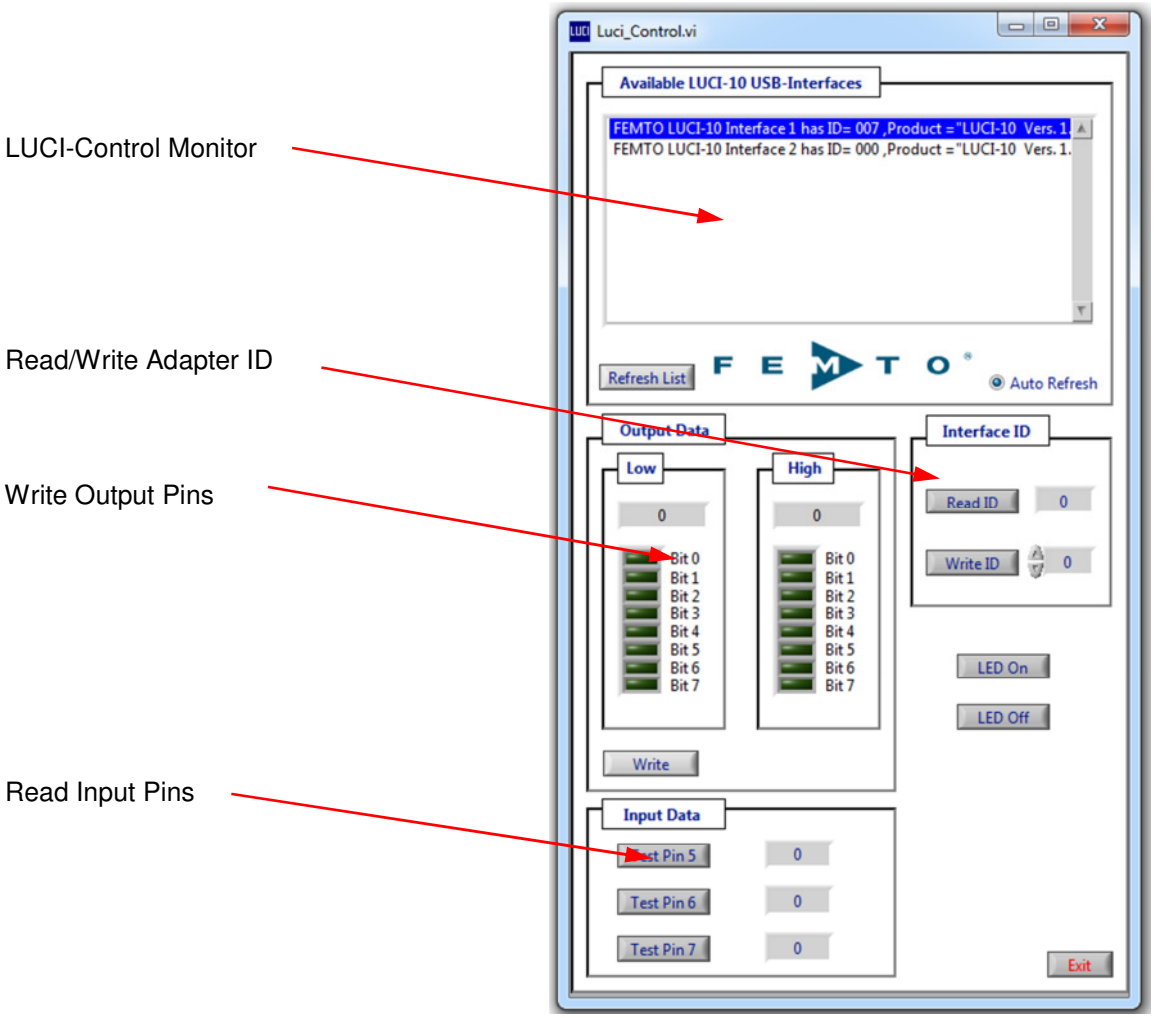
After halting the application by clicking the *STOP* button and restarting it with the *Start* arrow, the amplifier settings will be reset to the start condition. This is also reflected by the GUI front panel switches which also show the start condition.

Supported FEMTO Modules:

The software package contains GUI applications for all FEMTO standard modules with 25 pin D-Sub socket except for the logarithmic amplifier HLVA-100 which is not supported by LUCI-10. The GUI applications can be found at *Start >> Programs >> FEMTO LUCI-10 >> GUIs >> Name of the FEMTO Module.*

3.3 Application Program LUCI-Control Based on LabVIEW

The application program LUCI-Control allows an easy check and functional test of the LUCI-10 interface independently of a FEMTO amplifier or photoreceiver. The executable program was created in LabVIEW and is located in the LUCI-10 directory of the hard disk. It can be started by clicking on *LUCI-10_Control* in the menu S*tart >> Programs >> FEMTO LUCI-10*. All essential functions of the LUCI_10.dll device driver (see chapter 4.1) are implemented and can be used in this GUI application. One of the main applications of the LUCI-Control program is changing the adapter ID of a LUCI-10 interface.



By clicking the *Exit* button the program is stopped. By clicking the *Start* arrow the program can be restarted afterwards.

Functional Description of the LUCI-Control Application:

The upper part of the LUCI-Control window contains the LUCI-10 Monitor which lists the available interfaces presently connected to the USB bus. The interface number corresponds to the index which is automatically assigned by Windows. The displayed adapter ID is stored inside the LUCI-10 EEPROM and will not change even if the interface is disconnected from the USB port. After connecting or disconnecting a LUCI-10 interface to or from the PC, the monitor area will be updated to show the available devices. A manual update can be performed by clicking the *Refresh List* button.

To select a certain interface, click on the corresponding entry in the monitor area. For verifying your selection click on the *LED On* and *LED Off* buttons to turn the green LED on the LUCI-10 interface hardware on and off, respectively.

The factory setting for the LUCI-10 adapter ID is 255. The adapter ID is preserved even if the PC is turned off or if the LUCI-10 interface is removed from the USB bus. If more than one LUCI-10 interface is connected to the PC, the adapter ID should be changed to achieve an unambiguous identification of the various interfaces. Follow these steps to change the adapter ID:

- Select a LUCI-10 interface by clicking on the corresponding item in the monitor area.
- Enter the new adapter ID in the field next to the *Write ID* button. The adapter ID can be chosen between 0 and 255_{DEC}.
- Confirm your entry by clicking the *Write ID* button. The new adapter ID will be stored in the EEPROM inside the LUCI-10 hardware.
- Click the *Read ID* or *Refresh List* button to display and verify the new adapter ID.
- For further information on the variables *index* and *adapter ID*, please also see chapter 4.1.4.

You can write data to the LUCI-10 interface by clicking on the Bit icons in the area *Output Data*. After creating the desired bit pattern, you can transfer it to the interface by clicking the *Write* button. The decimal code corresponding to the bit pattern is displayed in the field's top of the Bit icons.

Note: Please keep in mind that the bit pattern is transferred to the interface only after clicking the *Write* button.

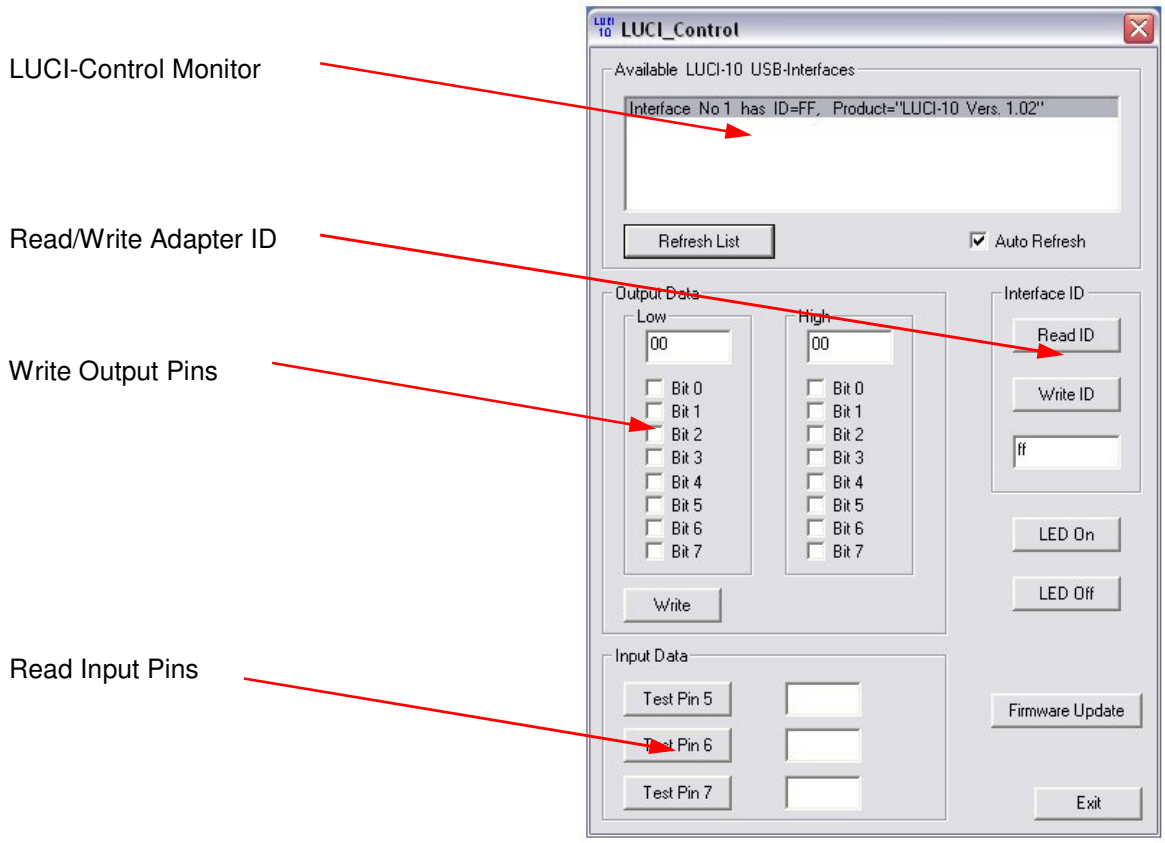
The LUCI-Control application can be also used to read data from the input ports of the interface. Follow these steps to read data from the three input pins.

- Select a LUCI-10 interface by clicking on the corresponding item in the monitor area.
- Click the buttons *Test Pin 5*, *6* or *7* to read the corresponding bit states of the input pins.
- The momentary bit state is displayed in the field next to the *Test Pin* button. A 0 and a 1 corresponds to a logic LOW and HIGH state, respectively.
- The input states are only read once at the time the *Test Pin* button is clicked. The display of the input states is not updated automatically.

3.4 Application Program LUCI-Control Based on C/C++

In Addition to the LabVIEW version of LUCI-Control described in the previous chapter, there is also a version of the program realized in C/C++. This application can be found in the LUCI-10 directory of the hard disk at *C:\Program Files\FEMTO\LUCI-10\Software\LUCI_Control\C Version*. There is no link to this application created in the *Start >> Programs >> FEMTO LUCI-10* menu during the installation routine.

All functions of the LUCI_10.dll device driver (see chapter 4.1) are implemented in the C/C++ version of the LUCI-Control program. The application is also executable if the LabVIEW Run-Time Engine is not or not properly installed on the PC. Thus, the C/C++ version of LUCI-Control allows the test of the LUCI-10 hardware independently of any LabVIEW drivers.



To support you regarding the programming under C\C++ a 64-bit LUCI-Control example with visible source code is available under:

C:\Program Files\FEMTO\LUCI-10\Software\C++Example program

Functional Description of the C/C++ Version of LUCI-Control and Differences to the LabVIEW Version:

The appearance and operation of the C/C++ version of LUCI-Control is almost identical to that of the LabVIEW version described in chapter 3.3. Therefore, we will only focus on the differences between these two programs in the scope of this manual:

- In the C/C++ version of LUCI-Control, any number is displayed as hexadecimal rather than as decimal value. Thus, the range for the adapter ID is 00_{HEX} to FF_{HEX} with the factory setting being FF_{HEX}.
- The C/C++ version supports two different ways of writing data to the LUCI-10 interface:
    - Bit by bit: When clicking on the Bit icons in the area *Output Data*, the data is immediately sent to the interface. Simultaneously, the resulting byte value is displayed in the corresponding field top of the Bit icons as hexadecimal value.
    - Byte by byte: It is also possible to change multiple bits at the same time by writing the Low or High Byte at once (Low Byte = pins 10-17, High Byte = pins 18-25). This can be done by writing a byte value as hexadecimal number in the fields Low and High Byte. The data is sent to the LUCI-10 hardware only after the *Write* button is clicked.
- The C/C++ version of LUCI-Control features another button called *Firmware Update*. This button is reserved for service purposes and should usually not be used. If this button is clicked accidentally, the LUCI-10 hardware enters a service mode and is no longer detected by Windows. In that case unplug and re-plug the interface from and to the USB bus to regain access to the device.

Note: In some cases the program might seem to crash after pressing the Enter key of the keyboard. This is due to the fact that the tab is typically set to the *Exit* button which causes the program to stop if Enter is pressed. Therefore, you should not press Enter after typing a byte code but click on the relevant button with your mouse pointer instead.

4 Advanced Software Procedures

In the following chapters the various software modules, drivers and libraries will be discussed in more detail. Some knowledge in programming languages like C/C++ or LabVIEW might be necessary for full comprehension.

4.1 Dynamic Link Library (DLL)

The device driver of the LUCI-10 interface is realized as Dynamic Link Library (DLL). The DLL contains program code but cannot be executed directly. Instead, its functions are called from other applications or scripts. A DLL can be used simultaneously by multiple applications even though the software code is loaded to the system memory only once. Chapter 4.1.5 describes in more detail how the DLL functions can be linked to the application programs.

4.1.1 Summary of the Included Device Driver Files

File	File Type	Description
LUCI_10.h	Header File	The header file can be used in a C/C++ application by using a #include command, e.g. #include "..\..\LUCI_10.h".
LUCI_10.lib	Import Library	Library file for linking to the DLL.
LUCI_10.dll	Dynamic Link Library	Library containing the functions of LUCI-10 provided to external applications.
Choose_Device_Enum.ctl	Type Definition	Device selection based on type definition provided by this driver.

4.1.2 The Header File LUCI_10.h

```
//      Header for LUCI_10.dll for use with LUCI-10 USB Interface
//      This Header file is both valid for the 32- and 64-bit version of the DLL.
//      Version         2.0
//      Date            20.03.17

#ifndef _FEMTO_LUCI10_H_
#define _FEMTO_LUCI10_H_

//  Error return codes
#define LUCI_OK                 0               //  Function returns successful
#define LUCI_ERROR_INDEX  -1                    //  Error, selected LUCI-10 not in list
#define LUCI_ERROR_HID      -2                  //  Error, LUCI-10 doesnt respond

//  return value of GetStatusPin (D-Sub pins 5,6,7)
#define STATUS_PIN_LOW          0               //  probed Pin is logical 0   (TTL low  0 Volt)
#define STATUS_PIN_HIGH         1               //  probed Pin is logical 1   (TTL high 5 Volt)

//  name of the DLL to be loaded
#define FEMTO_LUCI10_DLL "LUCI_10"

//  Implement the DLL export / import mechanism and allow a C-written program to use this DLL
#ifdef  FEMTO_LUCI10_EXPORTS
```

```
#define FEMTO_LUCI10_API extern "C" __declspec(dllexport)
#else
#define FEMTO_LUCI10_API extern "C" __declspec(dllimport)
#endif

//    list of exported functions, that can be used by other modules (.dll or .exe files)
FEMTO_LUCI10_API int EnumerateUsbDevices();
FEMTO_LUCI10_API int LedOn(int index);
FEMTO_LUCI10_API int LedOff(int index);
FEMTO_LUCI10_API int ReadAdapterID(int index, int *id);
FEMTO_LUCI10_API int WriteAdapterID(int index, int id);
FEMTO_LUCI10_API int WriteData(int index, int data_low, int data_high);
FEMTO_LUCI10_API int GetStatusPin5(int index, int *status);
FEMTO_LUCI10_API int GetStatusPin6(int index, int *status);
FEMTO_LUCI10_API int GetStatusPin7(int index, int *status);
FEMTO_LUCI10_API int GetProductString(int index, unsigned char *string, int size);

#endif  // _FEMTO_LUCI10_H_
```

4.1.3 Summary and Description of the LUCI_10.dll Functions

LUCI-10 is installed as Human Interface Device (HID) in Windows. The device driver uses the standard HID Kernel functions of Windows which are implemented into the DLL:

```
#include “98ddk/hidsdi.h“
#include “98ddk/setupapi.h“
```

After connecting a device to the USB bus, Windows queries some identification data like Vendor ID (VID) und Product ID (PID). This data is used to load the corresponding driver for the device.

```
#define MY_VID 0x21ef                    // Vendor-ID = FEMTO Messtechnik GmbH
#define MY_PID 0x010                     // Product-ID = LUCI-10
```

The following functions are defined in the device driver LUCI_10.dll for use with external programs:

`int EnumerateUsbDevices();`	Find and count the LUCI-10 interfaces presently connected to the USB bus; returns the quantity of detected devices. (0 = no device found).
`int LedOn(int index);`	Turn on the LED of LUCI-10 interface with number "index". “index“ is an integer between 1 and the number of detected devices.
`int LedOff(int index);`	Turn off the LED of LUCI-10 interface with number "index".
`int ReadAdapterID(int index, int *id);`	Return the adapter ID of LUCI-10 interface with number “index“. If the relevant interface does not have an adapter ID, the return value is -1.
`int WriteAdapterID(int index, int id);`	Write an adapter ID to LUCI-10 interface with number “index“. The adapter ID can take any value between 0x00 and 0xFF (hexdecimal). The adapter ID is stored in the EEPROM of the LUCI-10 hardware.

int WriteData(int index, int data_low, int data_high);	Write two 8 bit data words to the 16 output pins of LUCI-10 interface with number “index“. data_low: D-Sub pins 10-17 data_high: D-Sub pins 18-25
int GetStatusPin5(int index, int *status);	Read state of digital input pin 5 of LUCI-10 interface with number “index“. The maximum rate is 300 readings per second.
int GetStatusPin6(int index, int *status);	Read state of digital input pin 6 of LUCI-10 interface with number “index“. The maximum rate is 300 readings per second.
int GetStatusPin7(int index, int *status);	Read state of digital input pin 7 of LUCI-10 interface with number “index“. The maximum rate is 300 readings per second.
int GetProductString(int index, unsigned char *string, int size);	Read firmware information of LUCI-10 interface with number “index“.

4.1.4 The Integer Variables *Index* and *Adapter ID*

Each LUCI-10 interface has an interface number called ***index*** which is automatically assigned by Windows. The interface numbers start at 1 and increase by 1 for each LUCI-10 interface that is connected to the PC. If a LUCI-10 interface is disconnected from the PC the remaining interfaces might be assigned a new index from Windows.

Additionally each LUCI-10 has an ***adapter ID*** which is stored in the EEPROM of the interface. The adapter ID is preserved even if the PC is turned off or if the LUCI-10 interface is removed from the USB bus. The factory setting for the LUCI-10 adapter ID is 255_{DEC} or FF_{HEX}. Changing the adapter ID is possible by using the application LUCI-10_Control (see chapter 3.3).

If only one LUCI-10 is connected to your PC, the adapter ID can be left unchanged. Only in cases with more than one LUCI-10 connected to the PC, the adapter ID should be changed to achieve an unambiguous identification of the various interfaces.

The DLL functions use the *index* as parameter. If only one LUCI-10 device is connected to the USB port, you can use the parameter *int index = 1* when addressing the interface. In cases of two or more LUCI-10 devices connected to the USB bus, you might want to consider the *adapter ID (int id)* as relevant parameter because the *index* can change if one of the previously mounted devices is unplugged, or if additional LUCI-10 interfaces are connected. For this purpose, the virtual instrument *Adapter_ID to Index.vi* is provided in the LabVIEW Control palette (see also chapter 4.2.2). When using this VI with the *adapter ID* as parameter, the corresponding *index* is returned. This return value should be used as parameter for the following command, to assure that the correct interface is addressed.

4.1.5 How to Import the DLL in C/C++ Programs

The DLL must be imported in a program before any of its functions can be used. The DLL can be imported in a program in two different ways:

Implicit Linking:

The DLL is linked to the program through an import library with file extension .lib. The .lib file has to be specified in the program. The DLL functions have to be declared in the program, e.g. by using the LUCI_10.h header file, and can then be directly used. All functions with their parameters and data types are available from the import library and can be used in your build progress.

To call a DLL function you first have to import this specific function. Simply add the _declspec (dllimport) declaration ahead of the function.

Example:

```
extern “C“
{
    __declspec(dllimport) int LedOn(int index);            // declaration of the imported DLL function
}
```

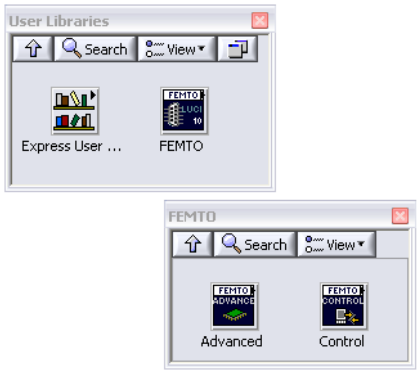
Explicit Run-Time Dynamic Linking:

The DLL is loaded at run-time using the *LoadLibrary()* command. The file-name of the DLL file is used as parameter for the LoadLibrary() command. If the DLL could be loaded, the function will return an Instance-Handle to the specific DLL. You can use this returned handle to get the address of the DLL function. Accessing the DLL functions is realized by using pointers.

Dynamic linking using the LoadLibrary() command will not import all functions of the specific DLL. Instead of the command LoadLibrary() you can use the API function GETProcAddress (Instance of the DLL, LPCSTR, funcName) to get a pointer referred to the selected function. Please keep in mind that the .lib file is required for this. The specific DLL should be in the same directory as the program that links to the DLL.

4.2 LabVIEW Palettes

The LUCI-10 software package contains two compact LabVIEW palettes. During the installation process these are copied to the LabVIEW user library palette. These palettes contain special VIs simplifying the use of LUCI-10 functions in a LabVIEW development environment.



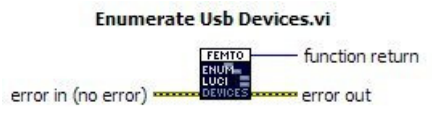
Note: After installation, it is recommended to do a mass compilation in LabVIEW before using the FEMTO libraries and VIs. Mass compilation can be started in *Tools >> Advanced >> Mass Compile*.

4.2.1 Advanced Palette

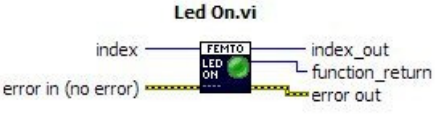
The Advanced palette corresponds to the exported functions of the device driver LUCI-10.dll. These low level VIs use the same parameters as the DLL functions described in chapter 4.1.3. In addition the typical error handling of LabVIEW can be used (error in, error out).

Summary of the Advanced Palette

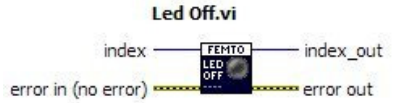
Explanation of the Low Level VIs



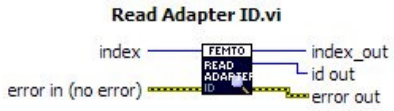
Find and count the LUCI-10 interfaces presently connected to the USB bus; returns the quantity of detected devices (0 = no device found).



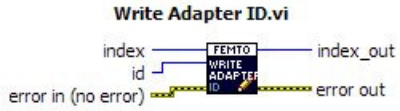
Turn on the LED of LUCI-10 interface with number "index". “index“ is an integer between 1 and the number of detected devices.



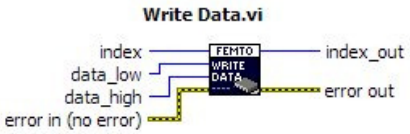
Turn off the LED of LUCI-10 interface with number "index".



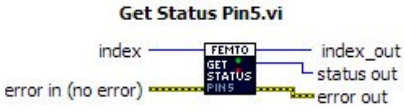
Return the adapter ID of LUCI-10 interface with number “index“. If the relevant interface does not have an adapter ID, the return value is -1.



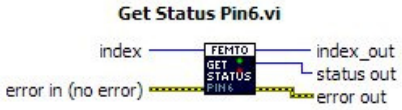
Write an adapter ID to LUCI-10 interface with number “index“. The adapter ID can take any value between 0 and 255_{DEC}. The adapter ID is stored in the EEPROM of the LUCI-10 hardware.



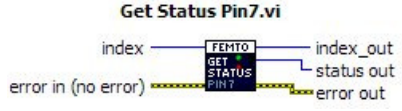
Write two 8 bit data words to the 16 output pins of LUCI-10 interface with number “index“.
data_low: D-Sub pins 10-17
data_high: D-Sub pins 18-25



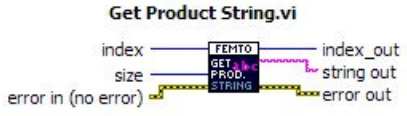
Read state of digital input pin 5 of LUCI-10 interface with number “index“. The status is read for a duration of 1 ms.



Read state of digital input pin 6 of LUCI-10 interface with number “index“. The status is read for a duration of 1 ms.



Read state of digital input pin 7 of LUCI-10 interface with number “index“. The status is read for a duration of 1 ms.

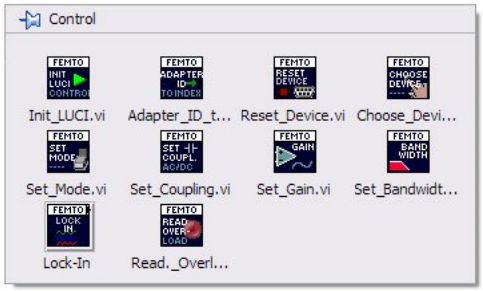


Read firmware information of LUCI-10 interface with number “index“.

4.2.2 Control Palette

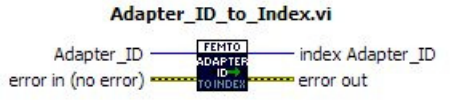
This palette was developed for the direct remote control of FEMTO modules through the LUCI-10 interface. The contained high level VIs perform complex functions for an easy use in LabVIEW. After choosing a specific FEMTO amplifier or photoreceiver through the VI *Choose_Device*, parameters like gain, bandwidth or signal filters can be directly set. Thus, the high level VIs of the Control palette reflect the manual control elements of the FEMTO modules without the need to go down to the bit level for changing the settings.

Summary of the Control Palette



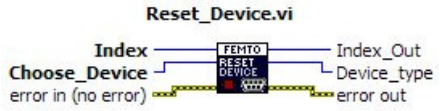
Explanation of the High Level VIs

Find and count the LUCI-10 interfaces presently connected to the USB bus; returns the quantity of detected devices.
(0 = no device found).

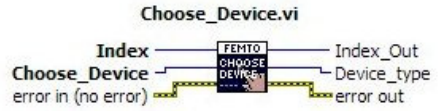


Return the current *index* corresponding to the *adapter ID* passed as parameter to the VI.

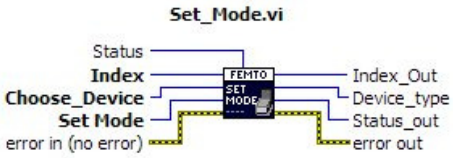
Note: This function is especially useful if more than one LUCI-10 interface is connected to the USB bus. The *index* can change if one of the previously mounted devices is unplugged, or if additional LUCI-10 interfaces are connected but the *adapter ID* is constant as it is stored in the EEPROM inside the LUCI-10 hardware. The VI *Adapter_ID_to_Index* establishes a link between the *adapter ID* and the currently valid *index*. The return value of this VI should be used as parameter for the following command, to assure that the correct interface is addressed (see also chapter 4.1.4).
(0 = device not found).



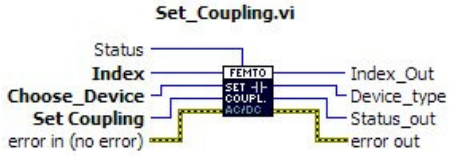
Reset the connected FEMTO module selected through the VI *Choose Device* to a standard setting. In the standard setting all control bits are set to LOW.



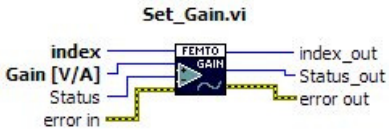
Select a FEMTO module and provide this information to the following VIs for a simplified programming.



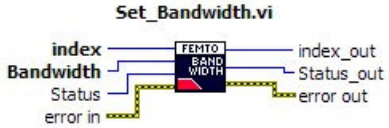
Set the selected FEMTO module to *High Speed* or *Low Noise* mode. This VI supports the FEMTO modules DHPCA-100, DLPCA-200, OE-200 and OE-300.



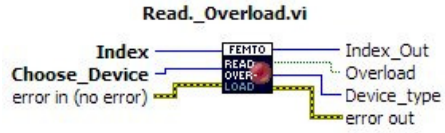
Set the selected FEMTO module to *AC* or *DC coupling.* This VI supports the FEMTO modules DHPCA-100, DHPVA-100/200, DLPCA-200, DLPVA-100-B/F, DLPVA-100-BLN-S, OE-200 and OE-300.



Set the gain of the selected FEMTO module.



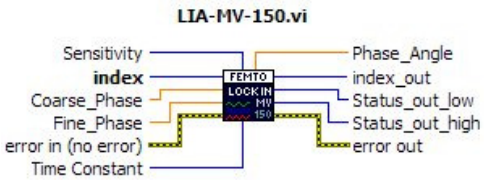
Set the low pass signal filter of the selected FEMTO module.



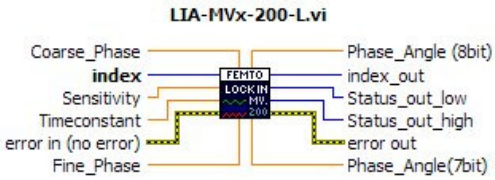
Read the *Overload* status of the selected FEMTO module. This VI supports the FEMTO modules DDPCA-300, DHPCA-100, DLPCA-200, DLPVA-100-B/F, DLPVA-100-BLN-S, DLPVA-100-BUN-S, LIA-MV-150, LIA-MV(D)-200, OE-200 and OE-300.

4.2.3 Lock-In Sub-Palette

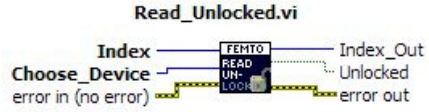
The Control palette contains the sub-palette Lock-In with special commands for controlling FEMTO lock-in amplifiers.



Set the sensitivity, phase and time constant of the LIA-MV-150-S/D lock-in amplifier.



Set the sensitivity, phase and time constant of the LIA-MV(D)-200-L/H lock-in amplifier.



Read the *Unlocked* status of the selected lock-in amplifier. This VI supports the FEMTO modules LIA-MV-150 and LIA-MV(D)-200.

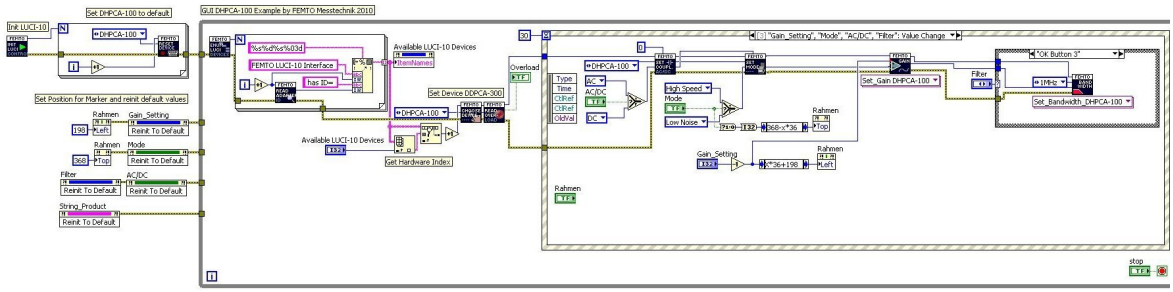
Note: Some VIs feature a *Status* and *Status_Out* pin. This is necessary as some functions changing only one bit would reset all other settings of the device by writing logic LOWs to the remaining 15 bits. This can be avoided by linking the current setting (i.e. the current bit pattern) by a logic OR function to the new bit value of the corresponding VI. In LabVIEW this is realized by connecting the *Status_Out* Pin of a VI to the *Status* Pin of the successive VI.

4.3 GUIs for FEMTO Amplifiers as Virtual Instruments (.vi)

In chapter 3.2 we described the executable GUI.exe files which can be used to easily remote control FEMTO amplifiers and photoreceivers. These files are also provided as virtual instruments in the form of .vi projects. The GUIs were created using the functions contained in the Advanced and Control palettes. The GUI.vi files can be loaded as virtual instruments (incl. front panel and block diagram) in LabVIEW. They can be also altered and adapted to meet the specific needs of your own application. Please feel free to use them as programming examples when realizing your own LUCI-10 projects. After installation, the sample GUI.vi files are located in the directory

C:*Program Files\FEMTO\LUCI-10\Software\LabVIEW\GUIs.*

The following picture shows the block diagram of the GUI DHPCA-100.vi. The corresponding front panel with its functions was discussed in chapter 3.2. A detailed explanation of the block diagram is beyond the scope of this operating manual.



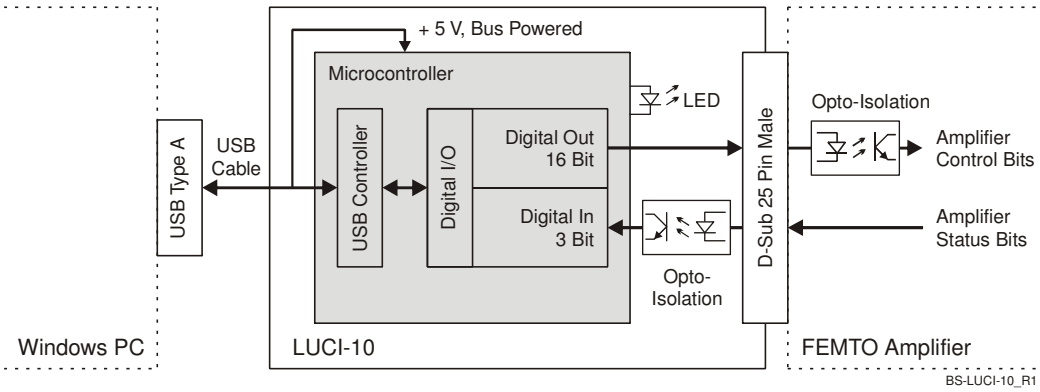
5 Important User Information

- The LUCI-10 interface is designed and suited for low voltages only. Please pay attention to the relevant regulations!
- The interface may be used with fully closed housing only.
- The interface has to be mounted to the FEMTO amplifier and photoreceiver only if the relevant module is without power. After mounting, the jack screws have to be securely tightened.
- There are no serviceable parts inside the interface housing. Keep the housing closed at all times.
- If the LUCI-10 interface needs repair please contact FEMTO Messtechnik GmbH for details. Do not try to repair the interface yourself!
- The case of the LUCI-10 interface is on the same ground potential as the FEMTO amplifier or photoreceiver connected to it. The ground potential of the USB bus is isolated from the module ground inside the interface. Keep the USB bus ground and module ground separate, and avoid ground loops as this could impact the signal quality.
- The LUCI-10 interface may not be used in security relevant or medical applications.
- Opening the housing or operating the interface outside the intended or permitted use will render all warranty agreements null and void.

The WEEE symbol (crossed-out wheelie-bin) indicates that the product may not be treated as household waste. At the end of life, the product must be either properly disposed of in accordance with existing guidelines or sent back to FEMTO Messtechnik GmbH at your own expense. By ensuring the product is disposed of correctly, you will help protect the environment.

6 Technical Data

6.1 Block Diagram of the LUCI-10 Interface



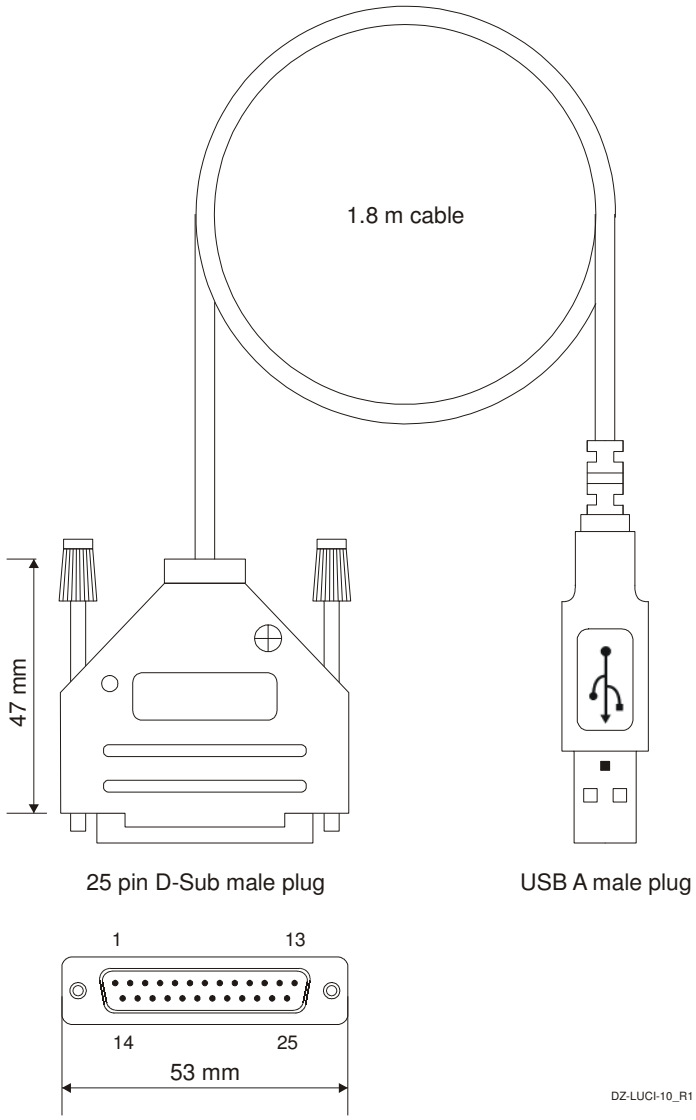
6.2 Software Specifications

System Requirements	Operating System	Microsoft® Windows XP SP 3, or higher
	Processor	Intel Pentium III or equivalent/higher
	System Memory	1 GB RAM or more
	Hard Disk Space	ca. 5 GB
	Interface Port	USB 1.1 or USB 2.0
	Max. Number of LUCI-10 Interfaces Connected to one PC	40
	Supported FEMTO Modules	All standard FEMTO amplifiers and photo-receivers with 25 pin D-Sub socket except for model HLVA-100
Optional Requirements	Development Platform (for own projects)	e.g. NI LabVIEW Version 2012, or C/C++ environment
Software Package (inclusive)	Installation Routine	Setup.exe
	Device Driver	Dynamic Link Library (.dll) for use in Windows with C/C++, LabVIEW, LabWindows/CVI, or similar
	Applications Programs with Graphical User Interface (GUIs)	Executable Programs (.exe) and LabVIEW project files (.vi) for test and control of the LUCI-10 hardware, and for easy remote control of FEMTO amplifiers and photoreceivers
	LabVIEW Libraries	Advanced Library (low level VIs) Control Library (high level VIs)
	LabVIEW Run-Time Engine	Platform necessary to run the executable (.exe) GUI applications if LabVIEW is not installed on the host PC
	Documentation	Operating manual and datasheet in pdf format

6.3 Hardware Specifications

PC-Side	Bus Interface	USB 2.0 (full-speed)
	Supply	PC USB-Port, + 5 V, typ. 100 mA, bus-powered (no auxiliary power supply required)
	Connector	USB Type A
	Cable	AWG 28, length 1,8 m
Module-Side	Interface Type	Digital I/O Control
	Connector	D-Sub, 25 pin, male
Output	Number of Channels	16 output lines, supporting opto-isolation inside FEMTO amplifiers and photoreceivers
	Output Voltage Range	LOW bit: 0 … +0.5 V (@ 0 ... 2 mA output current) HIGH bit: +4 … +5.5 V (@ 0... 2 mA output current)
	Max. Output Current	6 mA, per channel
	Writing Rate	max. 600 operations per second, software dependent
Input	Number of Channels	3 opto-isolated digital input lines
	Input Voltage Range	LOW bit: -20 ... +1.5 V HIGH bit: +3 ... +20 V
	Switching Current	1 mA typ. @ 5 V, per channel
	Reading Rate	max. 300 operations per second, software dependent
Case	Case Type	D-Sub metal hood for EMI/RFI shielding, with jack screws UNC 4-40
	Material	zinc die-cast, nickel plated
	Weight	130 g (0.3 lb.)
Temperature Range	Storage Temperature	-40 … +100 °C
	Operating Temperature	0 … +50 °C
	Relative Humidity	10 % … 90 %, non-condensing
Absolute Maximum Ratings	Max. Voltage at Input	+/-30 V
	Max. Short Circuit Output Current	+/-20 mA per channel, 200 mA total
	Max. Isolation Voltage	+/-60 V (Input Ground to Output Ground)
Conformity	CE	yes
	RoHS	yes
	WEEE	yes

6.4 Technical Drawing



7 Technical Support

We will be happy to assist you regarding questions related to the LUCI-10 interface and the related software components. We would be happy to hear about your feedback and suggestions for any improvements.

We hope for your understanding that we cannot offer support for LabVIEW or C/C++ projects realized by yourself. For questions related to LabVIEW please contact the provider of this software, National Instruments.

FEMTO Messtechnik GmbH
Klosterstr. 64
D-10179 Berlin · Germany
Tel.: +49 (0)30-280 4711-0
Fax: +49 (0)30-280 4711-11
e-mail: info@femto.de
http://www.femto.de