Your Global Automation Partner



TBEN-L5-4RFID-8DXP-WIN Compact RFID and I/O Module

Operating Instructions



Table of Contents

1	About these Instructions	7
1.1	Target Groups	7
1.2	Explanation of Symbols	7
1.3	Additional Documents	7
1.4	Feedback on these Instructions	8
2	Notes on the Product	9
2.1	Product Identification	9
2.2	Scope of Delivery	9
2.3	Legal Requirements	9
2.4	Manufacturer and Service	9
3	For Your Safety	10
3.1	Intended Use	10
3.2	General Safety Instructions	10
4	Product Description	11
4.1	Device Overview	11
4.1.1	Display Elements	11
4.2	Properties and Characteristics	11
4.3	Functional Principle	12
4.4	Functions and Operating Modes	12
5	Mounting	13
5.1	Grounding the Device	14
5.1.1	Grounding and Shielding Concept Grounding the Module (FE)	14
6	Connecting	16
61	Connecting the Modules to the Ethernet	16
62	Connecting the Power Supply	10
6.3	Connecting the RFID Read/Write Heads	18
6.4	Connecting Digital Sensors and Actuators	19
7	Commissioning	20
7.1	Setting the IP Address	20
7.1.1	Setting the IP Address via Switches on the Device	20
7.1.2	Setting the IP Address via the Turck Service Tool	22
7.2	Getting Started	25
7.2.1 7.2.2	Prerequisites Creating Applications	25
73	Drivers	25
7.3.1	Ethernet	26
7.3.2	NAND Flash	26
7.3.3	USB Host	26
/.3.4 725		26
7.5.5 736	GPIO	27
7.3.7	SPI	33

7.3.8	12C	36
7.3.9	RTC	37
7.3.10	Using the Application	37
7.3.11	Debugging the Application	38
7.3.12	Using a Network Socket in C#	38
7.3.13	Using the TBOX API Library	40
7.3.14	Procedure with a C# Application	42
7.4	Specific Settings / Implementations	43
7.4.1	Autostart Application	43
7.4.2	Image Version Readout	43
7.4.3	Addressing Mode Readout	43
8	Operation	44
8.1	LED Indicators	44
9	Troubleshooting	45
10	Maintenance	45
10.1	Carrying out a Firmware Update	45
11	Repairs	55
11.1	Returning Devices	55
12	Disposal	55



TBEN-L5-4RFID-8DXP-WIN

Hans Turck GmbH & Co. KG | T +49 208 4952-0 | F +49 208 4952-264 | more@turck.com | www.turck.com



1 About these Instructions

These operating instructions describe the structure, functions and the use of the product and will help you to operate the product as intended. Read these instructions carefully before using the product. This is to avoid possible damage to persons, property or the device. Retain the instructions for future use during the service life of the product. If the product is passed on, pass on these instructions as well.

1.1 Target Groups

These instructions are aimed at qualified personnel and must be carefully read by anyone mounting, commissioning, operating, maintaining, dismantling or disposing of the device.

The system integrator must be familiar with implementing applications under Windows and must be capable of commissioning the device for implementation on the Windows operating system without any additional support. Knowledge of integrating RFID read/write head protocols is also required.

1.2 Explanation of Symbols

The following symbols are used in these instructions:



DANGER

DANGER indicates an imminently hazardous situation with a high risk of death or serious injury if it is not prevented.



WARNING

WARNING indicates a potentially hazardous situation with a medium risk of death or serious injury if it is not prevented.



CAUTION

CAUTION indicates a situation that may result in damage to property if it is not prevented.

NOTE

NOTE indicates tips, recommendations and important information. The notes will make work easier, contain information on specific action steps and help prevent more work due to incorrect processes.

CALL TO ACTION

This symbol identifies action steps that the user has to perform.

ACTION RESULT

This symbol identifies relevant results of actions and action sequences.

1.3 Additional Documents

The following additional documents are available online at www.turck.com:

- Configuration manual
- Operating instructions for the read/write heads

1.4 Feedback on these Instructions

We are committed to always keeping these instructions as informative and as clear as possible. Should you have any suggestions for a better design or if any information is missing from the instructions, please send your suggestions to techdoc@turck.com.



2 Notes on the Product

2.1 Product Identification

These instructions apply for the following compact RFID interfaces: TBEN-L5-4RFID-8DXP-WIN

2.2 Scope of Delivery

Included in the scope of delivery:

- Compact RFID interface
- Closure caps for M12 connectors
- Quick start guide
- 2.3 Legal Requirements

The device falls under the following EU directives: 2014/30/EU (electromagnetic compatibility)

2.4 Manufacturer and Service

Turck supports you with your projects, from initial analysis to the commissioning of your application. The Turck product database contains software tools for programming, configuration or commissioning, data sheets and CAD files in numerous export formats. You can access the product database at the following address: www.turck.de/products Should you have any further questions, please contact the sales and service team in Germany under the following telephone numbers: Sales: +49 208 4952-380 Technology: +49 208 4952-390 Internet: www.turck.com/support

Outside Germany, please contact your local Turck representative.

Hans Turck GmbH & Co. KG Witzlebenstraße 7 45472 Mülheim an der Ruhr Germany

3 For Your Safety

The product is designed according to state-of-the-art technology. However, residual risks still exist. Observe the following warnings and safety notices to prevent damage to persons and property. Turck accepts no liability for damage caused by failure to observe these warning and safety notices.

3.1 Intended Use

The devices are only intended for use in industrial applications.

The block module TBEN-L5-4RFID-8DXP-WIN is an RFID interface for use in the Turck RFID system. The Turck RFID system is used for contactless exchange of data between a data medium and a read/write head in order to identify objects. The interfaces communicate with third-party systems such as ERP systems via TCP/IP. The device functions can be programmed via the Windows Embedded Compact 2013 operating system using .Net, C++ or C#. In addition, middleware functions can also be integrated on the device.

The devices must only be used as described in these instructions. Any other usage shall be considered improper and Turck shall not be held liable for any resulting damage.

3.2 General Safety Instructions

- The device may only be assembled, installed, operated and maintained by professionallytrained personnel.
- The device may only be used in accordance with applicable national and international regulations, standards and laws.
- The device only meets the EMC requirements for industrial areas and is not suitable for use in residential areas.



4 Product Description

The devices are executed in a fully encapsulated plastic housing with protection class IP67/ IP69K. Four RFID channels are available for connecting read/write heads. You can also connect sensors and actuators via eight digital I/O channels which you can freely configure as inputs and outputs. The connections for read/write heads and for digital I/Os are executed as M12 connectors. One M12 connector is available for the connection to the Ethernet.

4.1 Device Overview



Fig. 1: Dimensions

4.1.1 Display Elements

The devices have configurable multi-color LEDs.

- 4.2 Properties and Characteristics
 - Communication via TCP/IP
 - Freely programmable compact module based on Windows Embedded Compact 2013
 - Programming language: .Net, C++, C#
 - API available on request
 - 4 channels with M12 connection for RFID
 - 8 digital channels, configurable as PNP inputs and/or 2 A outputs
 - Integrated Ethernet switch allows line topology
 - Transmission rate: 10 Mbps/100 Mbps
 - Fully encapsulated module electronics
 - Protection classes IP65/IP67/IP69K
 - LEDs for status display

4.3 Functional Principle

The RFID interfaces connect the RFID system to other systems which communicate via TCP/IP (e.g. ERP systems). The interfaces have one Ethernet interface and multiple RFID interfaces. The RFID system is coupled with a third-party system, such as an ERP system, via the TCP/IP interface. The read/write heads are connected to the interfaces via the RFID interfaces. In addition, the interfaces can process signals from sensors and actuators via eight configurable digital channels.

4.4 Functions and Operating Modes

HF and UHF read/write heads can be connected to the RFID channels. It is also possible for HF and UHF read/write heads on one device to operate in parallel. The interface can perform control functions autonomously.

The device functions can be programmed via the Windows Embedded Compact 2013 operating system using .Net, C++ or C#. In addition, middleware functions can also be integrated on the device.

4.4.1 Functions in the Turck Service Tool

The device supports the following functions of the Turck Service Tool:

- Changing the IP address
- Reset to factory settings
- Carrying out a voltage reset
- Wink command
- Checking the firmware status



5 Mounting

The devices must be attached to a level, pre-drilled and grounded mounting surface.

➤ Attach the module to the mounting surface with two M6 screws. The maximum tightening torque for the screws is 1.5 Nm.



Fig. 2: Attaching the device to the mounting plate

5.1 Grounding the Device

5.1.1 Grounding and Shielding Concept

The grounding and shielding concept of the TBEN-S modules allows the fieldbus and I/O parts to be grounded separately.



Fig. 3: Replacement wiring diagram, shielding concept



Fig. 4: Grounding components

The grounding clip (1) on the M12 connectors for the fieldbus connection (P1, P2) connects the shield of the fieldbus lines.

The grounding ring (2) is attached below the grounding clip and connects the functional ground of the 7/8" connector (pin 3) for the power supply with the functional ground of the M12 connector (pin 5) for connecting the read/write heads, sensors, and actuators. The grounding screw (3) connects the device with the system's reference potential.



5.1.2 Grounding the Module (FE)

The grounding clip and the metal ring are connected to each other. A mounting screw through the bottom mounting hole in the module connects the shielding of the fieldbus lines to the functional ground of the power supply and the connected devices and to the reference potential of the system.

If a common reference potential is not required, remove the grounding clip to disconnect the fieldbus shield or attach the module with a plastic screw.

Removing the Grounding Clip

► Use a flat standard screwdriver to lever the grounding clip upwards and remove it.



Fig. 5: Removing the grounding clip

Mounting the Grounding Clip

- Insert the grounding clip between the fieldbus connectors (using a screwdriver if necessary) so that it makes contact with the metal housing of the connector.
- → The shield of the fieldbus lines lies flush to the grounding clip.



Fig. 6: Mounting the grounding clip

6 Connecting

6.1 Connecting the Modules to the Ethernet

For the connection to a TCP/IP system, the device has an integrated autocrossing switch with two 4-pin M12 Ethernet connectors. The maximum tightening torque is 0.6 Nm.



Fig. 7: M12 Ethernet connector for connection to a TCP/IP system

➤ Connect the device to a TCP/IP system according to the pin assignment shown below.



Fig. 8: Pin assignment for Ethernet connections



6.2 Connecting the Power Supply

For the connection to the power supply, the device has two 5-pin 7/8" connectors. V1 and V2 are galvanically isolated from one another. The maximum tightening torque is 0.8 Nm.



Fig. 9: 7/8" connector for the connection to the power supply

> Connect the device to the power supply according to the pin assignment shown below.

Pin assignment			
-	-(X1	Power feed
3 1 BK	= GND V2 3 = GND V1 2 3	X2	Continuation of the power to the next participant
4 3 GNYE = 5 1 4 BN	= FE 26004 = 24 VDC V1 1 5	V1	Power supply 1 (incl. supply to the electronics)
5 WH =	= 24 VDC V2 X2	V2	Power supply 2

Fig. 10: Pin assignment for the power supply connections

NOTE

The system voltage (V1) and the load voltage (V2) are fed in and monitored separately. If the permitted voltage is not reached, the slots are switched off according to the supply concept for the module type. If V2 is not reached, the PWR LED changes from green to red. If V1 is not reached, the PWR LED goes out.

6.3 Connecting the RFID Read/Write Heads

The device has four 5-pin M12 connectors for connecting RFID read/write heads. The maximum tightening torque is 0.8 Nm.



Fig. 11: M12 connector for connecting read/write heads

> Connect the read/write heads to the device according to the pin assignment shown below.

Pin assign	iment	Pin	
-(1	V _{AUX1}
2	1 = V _{aux} 1 2 = Data A	2	TX/RX-
10003	3 = GND 4 = Data B	3	GND (V1)
5 4	5 = FE/Shield	4	TX/RX+
		5	FE

Fig. 12: RS485 — pin assignment for read/write head connections

Pin assign	ment	Pin	
-(1	V _{AUX1}
2	1 = BN (+)	2	Data
10003	3 = BU (GND)	3	GND (V1)
5 4	4 = WH (Data) 5 = shield	4	Data
		5	FE

Fig. 13: Connection lines .../S2500 — pin assignment for read/write head connections

Pin assignment	Pin	
-(1	V _{AUX1}
2 = 1 = BN (+)	2	Data
$1 \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = 3 3 = BU (GND)$	3	GND (V1)
5 4 = BK (Data) 5 = shield	4	Data
	5	FE

Fig. 14: Connection lines .../S2501 — pin assignment for read/write head connections

Pin assignment	Pin	
-(1	V _{AUX1}
2 = 1 = RD (+)	2	Data
$1 \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} 3 = BK (GND)$	3	GND (V1)
5 4 5 = shield	4	Data
	5	FE

Fig. 15: Connection lines .../S2503 — pin assignment for read/write head connections



6.4 Connecting Digital Sensors and Actuators

The device has four 5-pin M12 connectors for connecting digital sensors and actuators. The maximum tightening torque is 0.8 Nm.



Fig. 16: M12 connector for connecting digital sensors and actuators

 Connect the sensors and actuators to the device according to the pin assignment shown below.

Pin assignm	ent	
-(2 1 000 3 5 4 C4C7	1 = V _{aux} 2 2 = Signal Out 3 = GND V2 4 = Signal Out 5 = FE	3 BU – 5 FE 4 BK 1 BN + 2 WH 3 BU – C C4C7

Fig. 17: Pin assignment for connecting digital sensors and actuators

7 Commissioning

7.1 Setting the IP Address

The IP address can be set via 2 decimal rotary coding switches and DIP switches on the device or via the Turck Service Tool.

7.1.1 Setting the IP Address via Switches on the Device

The IP address can be set via 2 decimal rotary coding switches and the "Mode" DIP switch on the device. The switches are located under a cover, along with the USB ports and the SET button.



Fig. 18: Switches for setting the IP address

> Open the cover over the switches.

- > Set the rotary coding switches to the desired position in accordance with the table below.
- > Set the "Mode" DIP switch to the desired position in accordance with the table below.
- ► Reset the voltage.
- ➤ ATTENTION! If the cover over the rotary coding switches is open, protection class IP67 or IP69K is not guaranteed. The device may be damaged by the ingress of foreign objects or liquids. Tightly close the cover over the switches.



Addressing Options

The IP address for the interfaces can be set in various ways. The following addressing options can be selected via the switches on the device. Changes to the settings are activated when the voltage is reset.

Setting option	"MODE" DIP switch	Rotary coding switch	Description
Default address	0	00	IP address: 192.168.1.100 Subnet mask: 255.255.255.0 Gateway: 192.168.1.1
Rotary mode	0	0199	In rotary mode, the last byte of the IP address can be set manually on the gateway. The other network settings are permanently stored in the gateway memory and cannot be changed in rotary mode. Addresses from 199 can be set.
DHCP mode	1	40	 In DHCP mode, the complete IP address is automatically assigned by a DHCP server in the network. The subnet mask allocated by the DHCP server and the default gateway address are permanently stored in the gateway memory. DHCP supports 3 types of IP address allocation: Automatic address assignment: The DHCP server assigns a permanent IP address to the client. Dynamic address assignment: The IP address assigned by the server is always only reserved for a specific period. Once this period has elapsed, or if it is explicitly released by a client, the IP address is reassigned. Manual address assignment: A network administrator allocates an IP address to the client. In this case, DHCP is only used to transfer the allocated IP address to the client.
PGM mode	1	50	In PGM mode, the complete IP address is assigned man- ually via the Turck Service Tool. In PGM mode, the set IP address and the subnet mask are stored in the gateway memory. All network settings (IP address, subnet mask, default gateway) are assumed by the module's internal EEPROM.
F_Reset	1	90	This mode resets all device settings to the default values and deletes all data in the device's internal flash memory. The following values are reset or deleted: – IP address and subnet mask – Parameter
Restore	1	00	IP address: 192.168.1.100 Network mask: 255.255.255.0 Gateway: 192.168.1.1

7.1.2 Setting the IP Address via the Turck Service Tool

The device is factory set to IP address 192.168.1.100. The IP address must be set via the Turck Service Tool. The Turck Service Tool is available to download free of charge from www.turck.com.

- ► Launch the Turck Service Tool.
- ► Click "Search" or press F5.

Your	Global Autom	ation Partner							TUI	ІСК
Search (F5)	Change (F2)	Wink (F3) Actic	ිූ [ons (F4) Clipl	Deard Langua	age Expert v	iew OFF	X Iose			
No. MAC a	address	Name	IP address	Netmask	Gateway	Mode	Device	Version	Adapter	Protocol

Fig. 19: Turck Service Tool – homescreen



← The Turck Service Tool displays the connected devices.

Ye O	our Global Aut	omation Q	Partner		1 1 4			-	TU		> IK
arch (F5) Change (F2) 1 MAC address 00:07:46:FF:A2:5A	Wink (F3) A Name	ctions (F4) Cli IP address 192.168.1.100	pboard Langu Netmask 255.255.255.0	age Expert Gateway 192.168.1.1	View ON Mode ROTARY	Start DHCP (F6) Configuratio Device TBEN-L5-4RFID-8DXP-WIN	vn (F7) AR Version 2.1.1.0	GEE (F8) Cl Adapter 192.168.1.93	ARGEE	Protocol Turck

Fig. 20: Turck Service Tool – found devices

- ► Select the device.
- ► Click "Change" or press F2.

Ŷ	our Global Aut	omation	n Partner						TU	RC	> IK
arch	(F5) Change (F2) MAC address 00:07:46:FF:A2:5A	Wink (F3) A Name	Actions (F4) Cli IP address 192.168.1.100	pboard Langu Netmask 255.255.255.0	age Expert Gateway 192.168.1.1	view ON Mode	Start DHCP (F6) Configuratio Device TBEN-L5-4RFID-8DXP-WIN	on (F7) AF Version 2.1.1.0	Adapter	ARGEE	Protocol Turck

Fig. 21: Turck Service Tool – select the device to be addressed.

- ➤ Change the IP address and if necessary the network mask and gateway.
- ► Accept the changes by clicking "Set in device".

P configuration (BOTA)	RY mode)
MAC address	IP address
00:07:46:FF:A2:5A	192.168.1 .100
Netmask	Gateway
255.255.255.0	192.168.1.1
itatus messages:	

Fig. 22: Turck Service Tool – changing the device configuration



7.2 Getting Started

7.2.1 Prerequisites

- Windows operating system (min. Windows 7)
- Visual Studio 2012/2013/2015
 - (https://www.visualstudio.com/de-de/products/visual-studio-express-vs.aspx)
- Application Builder (https://www.microsoft.com/en-us/download/details.aspx?id=38819)
- SDK (e.g. Turck_AM335x_RFID_SDKx.msi)

7.2.2 Creating Applications

Creating Applications in C/C++

- ➤ Make sure that the SDK is installed in the system (e.g. Turck_AM335x_RFID_SDKx.msi).
- ► Start Visual Studio.
- ► Select "New Project...".
- ➤ In the left-hand window area, under "Templates → Visual C++ -> Windows Embedded Compact", select "AM335x_Turck_RFID_SDKx".
- > In the central window area, select "Win32 Console Application".
- ► Enter a name in the lower area.
- Click "OK". Visual Studio starts in the Application Builder view with an automatically generated, simple console application.
- ► Add "#include "windows.h"".
- ► Add the following lines to the "wmain" function:

```
printf("Hello World\n");
Sleep(5000);
```

➤ Press F7 to compile the application.

Creating Applications in C#

- > Make sure that the SDK is installed in the system (e.g. Turck_AM335x_RFID_SDKx.msi).
- ► Start Visual Studio.
- ► Select "New Project...".
- ➤ In the left-hand window area, under "Templates → Other Languages -> Visual C++ -> Windows Embedded Compact", select "AM335x_Turck_RFID_SDKx".
- > In the central window area, select "Console Application".
- ► Enter a name in the lower area.
- Click "OK". Visual Studio starts in the Application Builder view with an automatically generated, simple console application.
- ➤ At the start of the "main cs" file, add "using System.Threading;" and "using System. Diagnostics;".
- ► Add the following lines in "Main Method":

```
Debug.WriteLine("Hello World");
Thread.Sleep(5000);
```

➤ Press F7 to compile the application.

7.3 Drivers

7.3.1 Ethernet

The Ethernet driver supports the Ethernet controller AM335x CPSW3G in switching mode.

The external ports of the Ethernet controller are mapped to the X_P1 and X_P2 plugs. The internal port of the Ethernet switch is available to the system as Ethernet device CPSW3G1.

The FTP server and the Telnet server are activated as default (with no authentication).

For details about WinSock, see

https://msdn.microsoft.com/EN-US/library/ee494651%28v=VS.80,d=hv.2%29.aspx.

7.3.2 NAND Flash

The NAND flash driver supports the AM335x GPMC controller and the inserted NAND flash card. The NAND flash driver is loaded automatically at the start and maps the FAT partition to the directory "/Mounted_Volume" transparently as standard.

The device directory is stored in the NAND flash file system permanently as standard.

7.3.3 USB Host

The USB host supports the Ethernet controller AM335x USB1 in host mode. The host is available on plug X25.

If the device is inserted and supported by the system, USB device drivers are loaded automatically. Human interface devices (MMIs) and USB mass storage are supported as standard.

7.3.4 USB OTG

In device mode, the USB OTG driver supports the Ethernet controller AM335x USB0. The USB OTG is available on plug X18.

Per default, the USB OTG device driver is configured to USB serial mode.



7.3.5 UART

The UART driver supports the UART1, UART2, UART3, and UART4 devices of the AM335x. The driver only supports RX/TX. Signals for a flow control are not available. The UARTs are achieved as COM1: - COM4: devices that use Win32 Serial Port API. For details, see https://msdn.microsoft.com/EN-US/library/ee488234%28v=VS.80,d=hv.2%29.aspx.

In order to correspond to the TBOX channels, the processor UARTs are mapped to the COM ports as follows:

UART1→ COM3:
UART2→ COM4:
UART3→ COM1:
UART4→ COM2:
By changing the value of the key
"HKEY_LOCAL_MACHINE\Drivers\BuiltIn\UARTX\RxFifoTriggerLevel" (possible values are 1-63), you can influence the receive FIFO of the UART.

7.3.6 GPIO

The GPIO driver supports the AM335x GPIOs as well as the PCA9506 GPIO expander and the XMC LED GPIOs.

The GPIO driver is available as a GIO1: device. It is available via the driver functions of the stream interface. For details, see

https://msdn.microsoft.com/EN-US/library/ee488234%28v=VS.80,d=hv.2%29.aspx.

The GPIOs must be identified via their GPIO ID. These are defined in the gpio_defines.h file which can be found in the SDK. The following GPIOs are supported by the BSP:

AM335x GPIO	GPIO ID	GPIO define
GPIO0_7	7	GPIO_7
GPIO0_12	12	GPIO_12
GPIO0_13	13	GPIO_13
GPIO0_19	19	GPIO_19
GPIO0_23	23	GPIO_23
GPIO1_15	47	GPIO_47
GPIO1_31	63	GPIO_63
GPIO2_22	86	GPIO_86
GPIO2_23	87	GPIO_87
GPIO2_24	88	GPIO_88
GPIO2_25	89	GPIO_89
GPIO3_14	110	GPIO_110
GPIO3_15	111	GPIO_111
GPIO3_16	112	GPIO_112
GPIO3_17	113	GPIO_113
GPIO3_18	114	GPIO_114
GPIO3_20	116	GPIO_116
GPIO3_21	117	GPIO_117

XMC GPIO	GPIO ID	GPIO define
P0.13	208	XGPIO_0
P0.12	209	XGPIO_1
P1.1	210	XGPIO_2
P1.0	211	XGPIO_3
P0.5	212	XGPIO_4
P0.4	213	XGPIO_5
P0.11	214	XGPIO_6
P0.10	215	XGPIO_7
P1.6	216	XGPIO_8
PCA GPIO	GPIO ID	GPIO define
PCA0_0[0]	128	EGPIO_0
•••		
PCA0_4[7]	167	EGPIO_39
PCA1_0[0]	168	EGPIO_40
PCA1_4[7]	207	EGPIO_79

The driver supports the following IOControl codes (defined in gpio_ioctls.h):

IOCTL_GPIO_SETBIT

Sets the corresponding GPIO to Level 1

Parameter	
lpInBuffer	Pointer to DWORD which contains the GPIO ID to
	be set

IOCTL_GPIO_CLRBIT

Sets the corresponding GPIO to Level 0

Parameter	
lpInBuffer	Pointer to DWORD which contains the GPIO ID to
	be set

IOCTL_GPIO_GETBIT

Reads from the level of the corresponding GPIO

Parameter	
pInBuffer	Pointer to DWORD which contains the GPIO ID to be read
lpOutBuffer	Pointer to DWORD which receives the current level



IOCTL_GPIO_SETMODE

Configures the mode of the corresponding GPIO

-		
Parameter		
pInBuffer		Pointer to an array of two DWORDs which contains the GPIO ID (array element 0) and the mode to be set (array element 1)
The following modes are su	pported (defined in	gpio_defines.h):
GPIO_DIR_OUTPUT	: configures the	GPIO as an output
GPIO DIR INPUT	: configures the	GPIO as an input

GPIO_DIR_INPUT	: configures the GPIO as an input
GPIO_INT_LOW_HIGH	: activation of the interrupt with a rising edge
GPIO_INT_HIGH_LOW	: activation of the interrupt with a falling edge
GPIO_INT_LOW	: activation of the low-level interrupt
GPIO_INT_HIGH	: activation of the high-level interrupt
GPIO_DEBOUNCE_ENABLE	: activation of the debounce

The modes GPIO_INT_LOW_HIGH, GPIO_INT_HIGH_LOW, GPIO_INT_LOW, GPIO_INT_HIGH, and GPIO_DEBOUNCE_ENABLE are not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XGPIO_0 – XGPIO_8).

IOCTL_GPIO_GETMODE

Returns the current mode of the corresponding GPIO

Parameter	
lpInBuffer	Pointer to DWORD which contains the GPIO ID
lpOutBuffer	Pointer to DWORD which receives the current mode

IOCTL_GPIO_CONFIGURE_PERIODIC_UPDATE

Sets the corresponding GPIO in toggle mode with 1 Hz or 2 Hz.

Parameter	
lpInBuffer	Pointer to a structure IOCTL_GPIO_PERIODIC_INFO (defined in gpio_ioctIs.h) which contains the GPIO ID and the periodic mode to be set:
	typedef struct { UINT uGpioID; UINT dwPeriMode; } IOCTL_GPIO_PERIODIC_INFO;

The following modes are supported (defined in gpio_defines.h):

GPIO_PERIODIC_DISABLE	: deactivates the periodic switching (toggling) of the GPIO
GPIO_PERIODIC_1HZ	: activation of the periodic switching of the GPIO with 1 Hz
GPIO_PERIODIC_2HZ	: activation of the periodic switching of the GPIO with 2Hz

This IOCONTROL function is not supported by XGPIO_8 GPIO.

IOCTL_GPIO_CONFIGURE_RXTX_UPDATE

Configures the corresponding GPIO for RX/TX signals

Parameter

IpInBufferPointer to DWORD which contains the GPIO ID to
be configured

The RX/TX signaling can only be deactivated by activating another mode via IOCTL_GPIO_CONFIGURE_PERIODIC_UPDATE, IOCTL_GPIO_SETBIT, or IOCTL_GPIO_CLRBIT.

This IOCONTROL is supported by GPIOs XGPIO_0 to XGPIO_8.

IOCTL_GPIO_CONFIGURE_AUTO_RECOVERY

Configures the automatic recovery of the GPIO

Parameter

lpInBuffer

Pointer to a structure IOCTL_OC_INFO (defined in gpio_ioctls.h) which contains the GPIO ID for activating the automatic recovery, the excess current detection of the GPIO, and the mode (automatic or manual recovery):

```
typedef struct {
UINT uGpioID;
UINT uOCGpioID;
BYTE bOCLevel;
BYTE bMode;
```

For the parameter bOCLevel (defined in gpio_defines.h), the following conditions of the trigger level are supported:

- GPIO_AUTOREC_OC_LOWLEV: the excess current condition of uGpioID is triggered by a lowlevel interrupt at uOCGpioID
- GPIO_AUTOREC_OC_HIGHLEV: the excess current condition of uGpioID is triggered by a highlevel interrupt at uOCGpioID
- GPIO_AUTOREC_OC_SAMELEV: the excess current condition of uGpioID is triggered when uOCGpioID is on the same level
- GPIO_AUTOREC_OC_DIFFLEV: the excess current condition of uGpioID is triggered when uOCGpioID is on the additional level

For the parameter bMode, the following modes (defined in gpio_defines.h) are supported: GPIO AUTOREC OC DISABLED: excess current detection deactivated

- GPIO_AUTOREC_OC_SHUTDOWN: the excess current condition is handled with a shutdown (manual recovery)
- GPIO_AUTOREC_OC_AUTOREC: the excess current condition is handled with a shutdown and after a safe period, with an automatic recovery



IOCTL_GPIO_GETIRQ

Returns the IRQ of the corresponding GPIO

Parameter	
lpInBuffer	Pointer to DWORD which contains the GPIO ID
lpOutBuffer	Pointer to DWORD which receives the IRQ number

This IOCTL function is not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XG-PIO_0 – XGPIO_8).

IOCTL_GPIO_SET_DEBOUNCE_TIME

Sets the debounce time of the GPIO (bank)

Parameter	
lpInBuffer	Pointer to a structure IOCTL_GPIO_SET_DE- BOUNCE_TIME_IN (defined in gpio_ioctls.h) which contains the GPIO ID and the debounce time to be set:
	<pre>typedef struct { UINT gpioId; UINT debounceTime; } IOCTL_GPIO_SET_DEBOUNCE_TIME_IN;</pre>

The debounce time is calculated as follows:

Debounce time = (DEBOUNCETIME + 1) \times 31 μ s. The debounce time is valid globally for all GPIOs of the same bank.

This IOCTL function is not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XG-PIO_0 – XGPIO_8).

IOCTL_GPIO_GET_DEBOUNCE_TIME

Returns the debounce time of the GPIO (bank)

Parameter	
lpInBuffer	Pointer to DWORD which contains the GPIO ID
lpOutBuffer	Pointer to DWORD which receives the debounce time

This IOCTL function is not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XG-PIO_0 – XGPIO_8).

IOCTL_GPIO_INIT_INTERRUPT

Initializes the interrupt for the GPIO

cture IOCTL_GPIO_INIT_INTER-
'inea in gpio_loctIs.h):
<pre>uct { ioID; ysIntrID; ent;</pre>

uGpioID must be set to the GPIO ID and hEvent to an event handle. The SysIntr used is returned in the element dwSysIntrID.

This IOCTL function is not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XG-PIO_0 – XGPIO_8).

IOCTL_GPIO_ACK_INTERRUPT

Acknowledges a GPIO interrupt

Parameter	
lpInBuffer	

Pointer to a structure IOCTL_GPIO_INTERRUPT_ INFO (defined in gpio_ioctls.h):

```
typedef struct {
UINT uGpioID;
DWORD dwSysIntrID;
} IOCTL_GPIO_INTERRUPT_INFO;
```

This IOCTL function is not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XG-PIO_0 – XGPIO_8).

IOCTL_GPIO_DISABLE_INTERRUPT

Deactivates the interrupt of a GPIO

Parameter		
lpInBuffer	Pointer to a structure IOCTL_GPIO_INTERRUPT_INFO	

This IOCTL function is not supported by the PCA (EGPIO_0 – EGPIO_79) and XMC GPIOs (XG-PIO_0 – XGPIO_8).



7.3.7 SPI

The SPI driver supports the AM335x MCSPI0 interface. The SPI0 interface is available as an SPI1: device.

The driver supports the following functions (defined in sdk_spi.h):

HANDLE SPIOpen(LPCTSTR pSpiName)

Opens the driver for later use	
Parameter	
pSpiName	String with the device name ("SPI1:")

Return value

Handle on driver

VOID SPIClose(HANDLE hContext)

Ends the driver after use	
Parameter	
hContext	Handle returned via SPIOpen()

BOOL SPILockController(HANDLE hContext, DWORD dwTimeout)

Locks the access to the driver for the current thread	
Parameter	
hContext	Handle returned by SPIOpen()
dwTimeout	Timeout for activation of the lock
Return value	
TRUE	Successful
FALSE	Not successful

BOOL SPIUnLockController(HANDLE hContext)

Unlocks the access to the driver. Locks the access to the driver for the current thread	
Parameter	
hContext	Handle returned by SPIOpen()
Return value	
TRUE	Successful
FALSE	Not successful

BOOL SPIConfigure(HANDLE hContext, DWORD address, DWORD config)

ndle returned by SPIOpen()
pselect number (only CS0 is supported)
/ORD which contains the desired configuration. e configuration must take place in accordance h the description of the register MCSPI_ 0CONF in the manual of the AM335x (Technical ference Manual).

Configures the SPI device for further actions

Return value	
TRUE	Successful
FALSE	Not successful

BOOL SPIEnableChannel(HANDLE hContext)

Activates the channel configured by the address parameter SPIConfigure() and therefore also the corresponding chipselect

Parameter	
hContext	Handle returned by SPIOpen()
Return value	
TRUE	Successful
FALSE	Not successful

BOOL SPIDisableChannel(HANDLE hContext)

Deactivates the channel previous activated by SPIEnableChannel()

Parameter	
hContext	Handle returned by SPIOpen()
Return value	
TRUE	Successful
FALSE	Not successful

BOOL SPISetSlaveMode(HANDLE hContext)

Configures the SPI controller for slave mode	
Parameter	
hContext	Handle returned by SPIOpen()
Poture value	
neturii value	
TRUE	Successful
FALSE	Not successful



DWORD SPIRead(HANDLE hContext, DWORD size, VOID *pBuffer)

Reads from the SPI bus	
Parameter	
hContext	Handle returned by SPIOpen()
size	Number of bytes to be read
pBuffer	Pointer to receivebuffer
Return value	
Number of bytes actually read	

DWORD SPIWrite(HANDLE hContext, DWORD size, VOID *pBuffer)

Writes to the SPI bus	
Parameter	
hContext	Handle returned by SPIOpen()
size	Number of bytes to be written
pBuffer	Pointer to sendbuffer
Return value	
Number of bytes actually written	

DWORD SPIWriteRead(HANDLE hContext, DWORD size, VOID *pOutBuffer, VOID *pInBuffer)

Reads/writes simultaneously from/to the SPI bus

Parameter	
hContext	Handle returned by SPIOpen()
size	Number of bytes to be read/written
pOutBuffer	Pointer to sendbuffer
pInBuffer	Pointer to receivebuffer

Return value

Number of bytes actually read/written

DWORD SPIAsyncWriteRead(HANDLE hContext, DWORD size, VOID *pOutBuffer, VOID *pInBuffer)

Reads/writes simultaneously from/to the SPI bus via DMA

Parameter	
hContext	Handle returned by SPIOpen()
size	Number of bytes to be read/written
pOutBuffer	Pointer to sendbuffer
pInBuffer	Not used, set to NULL

Return value

Value of the size parameter

DWORD SPIWaitForAsyncWriteReadComplet(HANDLE hContext, DWORD size, VOID *pOutBuffer)

Waiting for completion of the DMA transfer

Parameter	
hContext	Handle returned by SPIOpen()
size	Number of bytes to be written
pOutBuffer	Pointer to receivebuffer
Return value	
Value of the size parameter	

7.3.8 I2C

The I2C proxy driver (available in user mode) supports the AM335x-I2C0 interface. The I2C0 interface is available as an I2C1: device.

The I2C proxy driver is available via the file API (CreateFile(), ReadFile(), WriteFile(), SetFilePointer()).

To select the base subaddress used in the subsequent ReadFile()- WriteFile() calls, SetFilePointer() is used. For the selection of the address and the baud rate of the I2C device, the following IOControl codes are available (defined in i2cproxy.h):

IOCTL_I2C_SET_SLAVE_ADDRESS

Sets the slave address of the I2C device to be addressed

Parameter	
lpInBuffer	Pointer to DWORD which contains the slave address

IOCTL_I2C_SET_SUBADDRESS_MODE

Sets the subaddress mode

Parameter	
lpInBuffer	Pointer to DWORD which contains the desired sub- address mode. The following modes are available (defined in sdk_i2c.h): -I2C_SUBADDRESS_MODE_0: no device subaddresses -I2C_SUBADDRESS_MODE_8: 1-byte subaddresses -I2C_SUBADDRESS_MODE_16: 2-byte subaddresses -I2C_SUBADDRESS_MODE_24: 3-byte subaddresses -I2C_SUBADDRESS_MODE_32: 4-byte subaddresses



IOCTL_I2C_SET_BAUD_INDEX

Sets the baud rate of the I2C	
Parameter	
lpInBuffer	 Pointer to DWORD which contains the desired baud rate. The following baud rates are available (defined in sdk_i2c.h): SLOWSPEED_MODE: 100 KHz FULLSPEED_MODE: 400 KHz HIGHSPEED_MODE_1P16: 1.6 MHz HIGHSPEED_MODE_2P4: 2.4 MHz HIGHSPEED_MODE_3P2: 3.2 MHz

7.3.9 RTC

The OAL supports the onboard RTC. The RTC is used automatically by the system (OAL). Therefore, the time is read out and set via, for example, the functions GetSystemTime() and SetSystemTime() or via console functions for the date and time. The RTC is synchronized at system start (read) and when the value is changed (write).

7.3.10 Using the Application

To ensure that the device monitors the Application Builder deploy/debug requirements, components on the device have to be activated manually once the device has been started. This can be done via Telnet, for example:

➤ Connect the device via Telnet (default address IP 192.168.1.1).

➤ Run "start conmanclient3 & cmaccept3" in the command line of the device.

Once the device has been prepared for the connection with Application Builder, the device can be used and run on the device by pressing F5 (or via "Debug \rightarrow Start Debugging").

Application Builder should switch to the debug view, transfer the application to the device and then run it. When the example application referred to above is run, the string "Hello World" should be visible in the output window (debug) of Application Builder.

7.3.11 Debugging the Application

To insert a breakpoint in a specific line of the source code, click the vertical gray bar in front of the line. The breakpoint is identified with a red dot.



Fig. 23: Debugging the application

The next time the application is used and when it is run on the target device, it stops at the preselected breakpoint(s). This is indicated by a yellow arrow on top of the breakpoint.

A remote debugging via Visual Studio is very similar to the local debugging of an application, including the individual steps, callstack display (call list), memory display, etc.

7.3.12 Using a Network Socket in C#

You can implement a network communication in C# via the socket class. The following example code can be used to create and open a socket for a server at 192.168.1.1 on Port 80. For more information about the communication via the socket, see https://msdn.microsoft.com/en-us/library/system.net.sockets.socket_members(v=vs.90).

```
using System;
using System.Collections.Generic;
using System.Ling;
using System.Text;
using System. Threading;
using System. Diagnostics;
using System.Net;
using System.Net.Sockets;
namespace TurckWinSock
{
    class Program
    {
        static void Main(string[] args)
        {
            Debug.WriteLine("Hello World");
            //Create an endpoint with the server IP and Port 80
            IPAddress ip = new IPAddress(new byte[] {192, 168, 4,
80});
            IPEndPoint IpEnd = new IPEndPoint(ip, 80);
            //create a socket object
            Socket TestSocket = new Socket(IpEnd.AddressFamily,
SocketType.Stream, ProtocolType.Tcp);
```



```
try
            {
                //Connect the socket to the server
                TestSocket.Connect(IpEnd);
                //Check the connection
                if (TestSocket.Connected)
                {
                    Debug.WriteLine("socket connected");
                }
                else
                {
                    Debug.WriteLine("socket connection failed");
                }
                //Do something with the socket
                Thread.Sleep(5000);
                //Close the socket
                TestSocket.Shutdown(SocketShutdown.Both);
                TestSocket.Close();
            }
            catch (Exception e)
            {
                Debug.WriteLine("exception while connecting
socket");
            }
        }
    }
```

}

7.3.13 Using the TBOX API Library

An API library (TBOX API) is available for extracting functions of the DXPs and the COM ports. An overview and description of the functions of the API library are available in the header file TBOX_API.h.

Procedure with a C/C++ Application

To use the API library in a C/C++ application, integrate the header file TBOX_API.h and create a static link between TBOX_API_LIB.lib and the application.

Copy TBOX_API.h and TBOX_API_LIB.lib to your application directory and insert the library to the linker as an additional dependency:

- ► Select "PROJECT \rightarrow Project Properties" (Alt + F7).
- > In the left-hand window area, select "Configuration Properties \rightarrow Linker \rightarrow Input".
- ➤ In the right-hand window area, click the "Additional Dependencies" dropdown list and select "<Edit...>".

(neuro(bebug)	Platform: Active(AM335x_T	「urck_RFID_SDK5) ▼	Configuration Manager
Common Properties	Additional Dependencies	coredll.lib;%(AdditionalDependencies)	
Configuration Properties	Ignore All Default Libraries	<edit></edit>	
General	Ignore Specific Default Libraries		
Deployment	Module Definition File		
Debugging	Add Module to Assembly		
VC++ Directories	Embed Managed Resource File		
⊳ C/C++	Force Symbol References		
⊿ Linker	Delay Loaded Dils		
General	Assembly Link Resource		
Input	Assembly Eline Resource		
Manifest File			
Debugging			
System			
Optimization			
Embedded IDL			
Windows Metadata			
Advanced			
All Options			
Command Line			
Resources			
▷ Resources ▷ XML Document Generator			
 Resources XML Document Generator Browse Information 			
 Resources XML Document Generator Browse Information Build Events 			
 Resources XML Document Generator Browse Information Build Events Custom Build Step 	Additional Dependencies		
 Resources XML Document Generator Browse Information Build Events Custom Build Step Code Analysis 	Additional Dependencies Specifies additional items to add to the lin	nk command line [i.e. kernel32.lib]	

Fig. 24: Property pages



|--|

Additional Dependencies	? ×
TBOX_API_LIB.Iib	*
	v
•	•
Inherited values:	
coredll.lib	*
	-
Inherit from parent or project defaults	Macros>>
	OK Cancel

Fig. 25: Additional dependencies

- At the start of the main source file, integrate the header file TBOX_API.h via "#include "TBOX_API.h";".
- The application can use the TBOX API. Before using further API functions, "TBOX_Init();" must first be called. This resets all settings to the standard values and puts the hardware in a defined state.

Example 1: Have the green LED app flash:

TBOX_SYS_LED_GreenSet(LED_APP, LED_2HZ);

The setting LED_TxD_RxD_SIGNALING is supported only for LEDS assigned to a COM port.

Example 2: Import the value of the rotary switch:

```
DWORD dwVal = TBOX_SYS_SWITCH_Get(0);
printf("Rotary Switches Value: %d\n", dwVal);
```

Example 3: Configure the DXP as an input and read its current value:

TBOX_DXP_Init(8, FALSE, FALSE); DWORD dwVal = TBOX_DXP_Get(8); printf("DXP 8 Level: %d\n", dwVal2);

DXP0 to DXP7 are not used on this platform because they are occupied by the COM ports.

Example 4: Read all DXP inputs immediately:

```
dwVal = TBOX_DXP_Get(-1);
for(unsigned int i=8; i < 16; i++)
    printf("DXP %d Level: %d\n", i, (dwVal & (1 << i)) >> i);
```

The values of all DXPs are read out if "-1" is used as the DxpNo parameter. The return value is available via a bit field whose bits correspond to the number of the DXP level (e.g. bit0 for DXP 0, bit1 for DXP1, etc.).

Example 5: Configure a COM with the settings RS485 mode, Swap Lines, No Bias and No Termination. Also activate RX and TX signals for the corresponding LEDs and switch on the power supply VAUX for the port:

TBOX COM HardwareInit(TBOX COM1,

0, TBOX_COM_RS485_MODE, TBOX_COM_SWAP_AB_MODE, TBOX_COM_BIAS_OFF_MODE, TBOX_COM_TERM_OFF_MODE); TBOX_SYS_LED_GreenSet(LED_COM1_TX, LED_TxD_RxD_SIGNALING); TBOX_SYS_LED_GreenSet(LED_COM1_RX, LED_TxD_RxD_SIGNALING); TBOX_SYS_VAUX_Set(TBOX_COM1, TBOX_COM_POWER_24V_MODE);

The baud setting is ignored after the COM port has been set (standard value is 115200 baud). Only the RS485 mode is supported.

7.3.14 Procedure with a C# Application

To use the API in C# applications, the API is also available as a DLL (TBOX_API_DLL.dll). The DLL is not part of the OS standard image and must therefore be copied to the device. Before a function from the DLL can be used, the DLL must be imported to your C# application. Example:

```
[DllImport("TBOX_API_DLL.dll")]
public extern static int TBOX_Init();
[DllImport("TBOX_API_DLL.dll")]
public extern static void TBOX_SYS_LED_GreenSet(int LedNo, int
State);
```

After the import, the DLL can be called up within the C# application. Example:

TBOX_Init();
TBOX_SYS_LED_GreenSet(4, 0xCCCC);



7.4 Specific Settings / Implementations

7.4.1 Autostart Application

The default registry contains a reference to a specific application in the flash storage to autostart. Therefore it is possible to let your own application autostart on boot by placing it into "\ Mounted_Volume\" and naming it "autostart.exe".

7.4.2 Image Version Readout

To read out the current WEC image version, include "bsp_ioctls.h" from the SDK and use the following IOControl Code with KernelloControl: IOCTL_HAL_GET_BSP_VERSION Parameter lpOutBuffer: pointer to a IOCTL_HAL_GET_BSP_VERSION_OUT struct.

IOCTL_HAL_GET_BSP_VERSION_OUT is defined as follows:

typedef struct {
 DWORD dwVersionMajor;
 DWORD dwVersionMinor;
DWORD dwVersionQFES;
 DWORD dwVersionIncremental;
} IOCTL_HAL_GET_BSP_VERSION_OUT;

7.4.3 Addressing Mode Readout

In order to read out the addressing mode determined at boottime according to the pushbutton, dip- and rotary-switches, the following IOControl Code (defined in "bsp_ioctls.h") can be used with KernelloControl: IOCTL_HAL_GET_ADDRMODE

Parameter lpOutBuffer: pointer to DWORD receiving the addressing mode.

The Ethernet IP Settings are automatically modified at WEC startup according to the detected mode.

The addressing modes are defined on p. 21.

8 Operation

8.1 LED Indicators

The devices have freely programmable multi-color LEDs for displaying information about:

- Power supply
- Collective and bus faults
- Status

Diagnostics					
LED PWR	Meaning				
Off	No power or undervoltage	No power or undervoltage at V1			
Lights up in green	Power at V1 ok	Power at V1 ok			
Lights up in red	No power or undervoltage at V2				
LED BUS	Meaning	Meaning			
Flashing red	Wink command active				
COM LEDs (RFID channels)					
TXD LED	RXD LED	Meaning			
Flashes green	Off	Data being sent			
Off	Flashes green	Data being received			
Flashing red	Flashing red	Short circuit in the power supply			
	Illuminated in red	Memory overflow			
Illuminated red/green	Illuminated red/green	Incorrect configuration			
DXP LEDs (digital channels, LEDs 815)					
Green LED	Red LED	Meaning			
Off	Off	No I/O signal present			
Illuminated	Off	I/O signal present			
Off	Illuminated	Overload at output			
Flashing	Flashing	Overload of the auxiliary supply			



9 Troubleshooting

If the device does not work as expected, first check whether ambient interference is present. If there is no ambient interference, check the device connections for faults. If there are no faults, there is a device malfunction. In this case, decommission the device and replace it with a new device of the same type.

10 Maintenance

Perform regular checks to ensure that the connections and cables are always in good condition. The devices are maintenance-free. If necessary, clean in a dry state.

10.1 Carrying out a Firmware Update

The firmware for the device can be updated using the PuTTY and WinSCP tools.



CAUTION

Interruption of the power supply during the firmware update **Device damage due to faulty firmware update**

- > Do not interrupt the power supply to the device during the firmware update.
- > Do not reset the power supply during the firmware update.

Example: Updating Firmware

The example uses the following settings:

- The device is connected to the power supply via the X1 connection.
- The device is connected to the Ethernet via the ETH2 connection.
- The rotary coding switches on the device are in position 00 (default). The device's IP address depends on the firmware status:

Firmware status	IP address
Version 2.1.0.0 or earlier	192.168.1.1
Version 2.1.1.0 or later	192.168.1.100

The device is connected to the host PC.

The PuTTY and WinSCP software tools have been installed.

- ► Open PuTTY.
- ► Enter the following settings in PuTTY:
 - Host name: IP address of the device
 - Port: 23
- Optional: Assign a name for the current session (here: TBEN-Lx-WINCE). The session can be loaded via "Load" for later repetitions.
- ➤ With saved sessions: Select TBEN-Lx-WINCE and confirm with "Load".
- ► Click "Open".

NOTE

If a connection cannot be established, check the IP address of the host PC.

R PuTTY Configuration			? ×
Category:			
Session	<u> </u>	Basic options for your PuTTY ses	sion
Logging	Sp	ecify the destination you want to connec	t to
- Keyboard	Ho	ost <u>N</u> ame (or IP address)	Port
Bell	19	92.168.1.1	23
Features	Co	nnection type: Ra <u>w</u>	🔘 Se <u>r</u> ial
···· Appearance ···· Behaviour ···· Translation ···· Selection	E Los Sa Ti	ad, save or delete a stored session wed Sessions BEN-Lx_WINCE	
Colours Connection Data Proxy Telnet		efault Settings erial BEN-Lx BEN-Lx_WINCE BEN-Lx_WINCE_Debug BEN-Lx_alteFW	Load Sa <u>v</u> e Delete
	T Cic	_ ose window on e <u>x</u> it: Always ⊚ Never ⊚ Only on cle	an exit
<u>A</u> bout <u>H</u>	<u>l</u> elp	<u>O</u> pen	<u>C</u> ancel

Fig. 26: PuTTY configuration



192.168.1.3	1 - PuTTY				
\>	-				*
\>					
$\langle \rangle$					
\rightarrow dir					
Direct	tory of				
12/12/12	04:00a	<dir></dir>		Mounted Volume	
01/01/06	04:00a	<dir></dir>		My Documents	
01/01/06	04:00a	<dir></dir>		Temp	
01/01/06	04:00a	<dir></dir>		Windows	
Found 1 Dir	4 file (s) 1243	(s). Total size (346368 bytes free) bytes.		
<pre>\> cd wind</pre>	lows				
\windows>	dir				
L					
Direct	tory of	\windows			
01/01/06	12:00p		89	FTPD.tmp	
01/01/06	04:00a			initobj.dat	
01/01/06	04:00a	<dir></dir>		Кеуз	
10/21/15	06:06a		77824	cmd.exe	
10/28/16	04:29a		40960	timezones.dll	
10/21/15	06:08a		28672	ping.exe	
10/21/15 10/21/15	06:00a		28672	ndisconfig exe	
10/21/15	06:06a		45056	route.exe	
01/20/17	02:37a		36864	netstat.exe	
10/21/15	06:08a		28672	tracert.exe	
10/21/15	06:06a		36864	ipv6.exe	
01/20/17	02:36a		11614	ceconfig.h	
02/17/16	04:51a		9932	firmware.bin	
02/17/16	04:51a		65136	clientshutdown3.exe	
02/17/16	04:51a		67160	CMAccept3.exe	
02/17/16	04:51a		113256	Conmancilents.exe	
02/17/16	04.51a		112208	eDbgTL3 dll	
02/17/16	04:51a		92272	TcpConnectionA3.dll	
02/17/16	04:51a		214960	vsdebugeng.impl.resources.dll	=
02/17/16	04:51a		154176	edm3.exe	1
02/17/16	04:51a		660368	MSDIA110.dll	
02/17/16	04:51a		96144	msvsmon.exe	
02/17/16	04:51a		364	msvsmon.exe.config	
02/17/16	04:51a		1144216	VSDebugEng.dll	
02/17/16	04:51a		594848	vsdebugeng.impl.dll	
02/14/17	05:27a		24576	update.exe	
02/14/17	05:27a		20480	XMC_Load.exe	
04/19/16	12:23a		9366	XMC1x ASCLoader.bex	
09/06/16	10:15p		25722	83000xxx V0.1.7.0.hex	
Found	32 fil	e(s). Total size	3894719 byte	28.	
I DIr	(3) 1243	940300 bytes free			
\windows>					Ŧ

► Try to open the directory of the device with cd \windows.

Fig. 27: Directory \windows in PuTTY

Files can be exchanged between the host PC and the device via WinSCP.

- ► Start WinSCP.
- ► Log onto the device in WinSCP.
- ► Activate the "Anonymous" checkbox.

File protocol: Encryption: FTP No encryption Host name: Port number: 192.168.1.1 21 User name: Password: anonymous ••••••••• Edit Advanced	File protocol: Encryption: FTP No encryption Host name: Port number: 192.168.1.1 21 User name: Password: anonymous •••••••••• Edit Advanced	ず New Site	Ses	sion		
FTP No encryption Host name: Port number: 192.168.1.1 21 User name: Password: anonymous •••••••••••••••• Edit Advanced	FTP No encryption Host name: Port number: 192.168.1.1 21 User name: Password: anonymous •••••••••• Edit Advanced	TBEN-Lx-WinCE	Eile	protocol:	Encryption:	
Host name: Port number: 192.168.1.1 21 User name: Password: anonymous ••••••••••••••••••••••••••••••••••••	Host name: Port number: 192.168.1.1 21 User name: Password: anonymous ••••••••••••••••••••••••••••••••••••		FT	Р	No encryption	
192, 168, 1, 1 21 User name: Password: anonymous ••••••••••• Edit Advanced	192, 168, 1, 1 21 User name: Password: anonymous ••••••••••• Edit Advanced		Ho	st name:		Port number:
User name: Password: anonymous ••••••••••••••••••••••••••••••••••••	User name: Password: anonymous ••••••••••••••••••••••••••••••••••••		19	2.168.1.1		21
anonymous Edit Advanced	Edit Advanced		Use	er name:	Password:	
Edit Advanced	Edit Advanced		an	ionymous	•••••	•••••
				Edit		Advanced 🖙

Fig. 28: Logging on in WinSCP



In WinSCP, select the folders on the host PC and on the device between which you want to exchange files.

Intel Stat Freighen Asiot door of 131 Image: State	😺 I 🕞 👪 = I			Upo	dateWinCe2110customer	\$				- 🗆 🛛
store	Datei Start Freigel	ben Ansi	cht							~ 🤅
update_win_21.1.0 - TBEN-Lix-WinCE - WinSCP Image: Section Options Remote Help Image: Synchronic Image: Synchronic Image: Section Options Remote Help Image: Synchronic Image: Section Options Remote Help Image: Synchronic Image: Synchronic Image: Section Options Remote Help Image: Synchronic Image: Section Options Remote Help Image: Synchronic Image: Synchronic Image: Section Options Remote Help Image: Synchronic Image: Section Options Remote Help Image: Synchronic Image: Section Options Remote Help Image: Section Options Remote Help Image: Synchronic Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help Image: Section Options Remote Help	ndows CE Teln (🔿 - 🛧 🌗 Di	ieser PC → E	okumente → TBEN-L	x-4RFID-8DXP-WIN → Up	dateWinCe2110customer			v ¢ "∪;	odateWinCe2110cr	ustomer" 🔎
Local Mark Files Commands Session Options Remote Help Some Synchronice Source Transfer Settings Default TERN-Low Mut C C we Session C Lubicad - C Laft R C Propertie R C C C C Anged Rights Owner C Lubicad - C Laft R C C C C C C C C C C C C C C C C C C			u	pdate_win_2.1.1.0 - TI	BEN-Lx-WinCE - WinSCP					□ ×
Image: Synchronize Image: Sy	Local Mark Files Commands Session Option	is Remote H	Help							
Image: Serie Type Changed Image: Serie Type Changed Image: Serie Type Changed Image: Serie Type Changed Image: Serie Type Changed Image: Serie Type Changed Image: Serie Type Serie Type Image: Serie Type Changed Image: Serie Type Changed Image: Serie Type Serie T	🖶 🕃 📮 Synchronize 🗾 🧬 💽 🍥	🗿 Queue 👻	Transfer Setting	s Default	• 🦪 •					
My documents My documents <td< td=""><td>TBEN-Lx-WinCE I New Session</td><td></td><td>20000000000</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	TBEN-Lx-WinCE I New Session		20000000000							
Lubacat	My documents	- 10 0	n 2 %		🚺 Windows 🔹 🤗 🔽	7 4 -		Find Files	2	
Cu., Władki, Schnabel Document, VIBEN-Li-4RIPO BUN-WINUJGATEWING 221 IOCutomer/update.win, 21.110 Windows Name Size Type Changed Parent directory 21.07.2017 09:1105 Beboch bat 168 Windows-Batchau. 220.2017 11:46:32 Size Size 25.K8 NBO-Datei 19.06.2017 12:43:22 Size Size 118 Windows-Batchau. 220.2017 11:46:44 Size Size 12.K8 30000cc, V0.14.0.hex 25.K8 Size Size 18.0000cc, V0.14.0.hex 25.K8 25.K8 Size 18.0000cc, V0.14.0.hex 25.K8 18.0000cc, V0.14.0.hex Size 18.0000cc, V0.14.0.hex 18.0000cc, V0.14.0.hex 18.0000cc, V0.14.0.hex Size 18.0000ccc, V0.14.0.hex 18.0000cc, V0.14.0.hex	🕞 Unload 🖌 📝 Edit 🚽 🌱 📥 Propertie				E Download + D'Edt	- × _/			1. V -	
Name Size Type Changed Name Beborthundernoverlautering 21.07.2017 09:11:05 1KB Windows-Batchda 22.02.2017 11:46:48 Size Size	C) \ Hendrik Schnabel\ Decuments\ TPEN, Ly, 4PEI		UndateWinCo2110cur	tomorlundate win 2110	Windows	· • •				
Name Jose Overal Name State Name Bebootbat 1158 Vindows-Batchda 220/2017 1091105 220/2017 104648 220/2017 114648 Sebootbat 256 KB NBO-Datei 19.062017 124452 200000,V0.14.0.hex 25 KB 20.062017 114648 So OS, MK.bat 11KB Vindows-Batchda 22.02.017 1147.14 20.02.017 1145.44 200.000,V0.14.0.hex 25 KB 20.06.001 124452 Windows-Batchda 22.02.017 1145.44 19.06.2017 124452 ConmarClient3.ee 111 KB 17.02.2016 0451 Windows-Batchda 22.02.017 1145.44 19.06.2017 124452 ConmarClient3.ee 111 KB 17.02.2016 0451 Windows-Batchda 22.02.017 1145.44 19.06.2017 124452 ConmarClient3.ee 111 KB 17.02.2016 0451 Windows-Batchda 22.02.017 1145.44 19.06.2017 124452 ConmarClient3.ee 111 KB 17.02.2016 0451 Windows-Batchda 22.02.017 1145.44 19.06.2017 124452 ConmarClient3.ee 111 KB 17.02.2016 0451 Windows-Batchda 22.02.017 1145.44 19.06.2017 124532 ConmarClient3.ee 111 KB 17.02.2016 0451 Windows-Batchda 10.02.017 124532 ConmarClient3.ee 111 KB 17.02.2016 0451 11.02.015 0606	Name	Cia-	Ture	Changed	Nama	Circ	Channed	Dishts	Ourse	•
image: construction image: construction<		SIZE	Devest disectory	21.07.2017.00.11.05	Ndrie 1	5126	Changed	Rights	Owner	
Betorthola 1.68 Windows-Batcha 22.02.017 11.443.52 DK.hob 32.768.KB NBO-Datei 19.06.2017 12.443.52 ccconfig.h 12.KB 0.06.2016 03:48 NK.hob 32.768.KB NBO-Datei 19.06.2017 11.471.424 Ccconfig.h 12.KB 0.06.2016 03:48 NK.hob 32.768.KB NBO-Datei 19.06.2017 11.471.424 Ccconfig.h 12.KB 0.06.2016 03:48 Mdmand.block 1.KB Windows-Batcha 22.02.2017 11.471.44 Cleintshutdown3.exe 64 KB 17.02.2016 04:51 Windows-Batcha 22.02.2017 11.451.44 CommanClient3.exe 11 KB 17.02.2016 04:51 Windows-Batcha 22.02.2017 11.451.44 CommanClient3.exe 11 KB 17.02.2016 04:51 Windows-Batcha 22.02.2017 11.451.44 10 KB 17.02.2016 04:51 CommanClient3.exe 11 KB 17.02.2016 04:51 Windows-Batcha 22.02.2017 11.451.44 10 KB 17.02.2016 04:51 CommanClient3.exe 11 KB 17.02.2016 04:51 Windows-Batcha FtPD.tmp 1 KB S0.01.20015 01:10		1 1/12	Windows Patchda	22.07.2017 09:11:05	Sa		20 01 2006 17:01			
1900-Unit 1900-Unit 1900-	EPOOTND ab0	256 VD	MPO Datai	10.06.2017 12:42:52	02000eer V0140 her	25 KD	29.01.2000 17:01			
100000 100000111111111 100000111111111 100000111111111 0005_NK.bat 11.KB Windows-Batchda 22.02.2017 11.47.14 44 KB 1B_LOCK-Datei 19.06.2017 12.43.52 III.CML down3.exe 66 KB 17.02.2016 04.51 3.xldmand.bat 1 KB Windows-Batchda 22.02.2017 11.45.44 III.KB CommanClient3.see 11.11 KB 11	NK pb0	32 768 KB	NB0-Datei	19.06.2017 12:45:32	ooootx_vo.1.4.0.nex	12 KB	03 06 2016 03:48			
Construction 1 KB Kindbard 1 KB Exclusion 1 results include in the interview of the i		1 KB	Windows-Batchda	22 02 2017 11.47.14	Clientsbutdown3 eve	64 KB	17.02 2016 04-51			
Addmand.back 1 KB Windows-Batchde 22.02.2017 11:45:44 Image: CommanClient3.exe 11 KB 17.02.2016 04:51 DeviceAgentTranspor 79 KB 17.02.2016 04:51 DeviceAgentTranspor 79 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 Disposition 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 Disposition 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 Disposition 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 Disposition 10 KB 17.02.2016 04:51 DeviceAgentTranspor 10 KB 17.02.2016 04:51 Disposition	vidmand 1block	AA KB	1BLOCK-Datei	19.06.2017 12:43:52	CMAccent3 eve	66 KB	17.02.2016.04:51			
Ind Index Station Excess in Figure Total Index Station Ind Index Station Excess in Figure Total Index Station Ind Index Station Excess in Figure Total Index Station Ind Index Station Index Station Index Station Index Station Ind Index Station Index Station Index Station Index Station Ind Index Station Index Station Index Station Index Station Ind Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station Index Station <td>vidmand hat</td> <td>1 KB</td> <td>Windows-Batchda</td> <td>22 02 2017 11:45:44</td> <td>Cmd eve</td> <td>76 KB</td> <td>21 10 2015 06:06</td> <td></td> <td></td> <td></td>	vidmand hat	1 KB	Windows-Batchda	22 02 2017 11:45:44	Cmd eve	76 KB	21 10 2015 06:06			
Composition OpericeAgentTransport 79 KB 17.02.2016 04:51 Composition ObgTL3.dll 110 KB 17.02.2016 04:51 Composition Office OpericeAgentTransport 79 KB 17.02.2016 04:51 Composition Office OpericeAgentTransport 10 KB 17.02.2016 04:51 Composition Office OpericeAgentTransport 10 KB 17.02.2016 04:51 Composition Office OpericeAgentTransport 10 KB 17.02.2016 04:51 Composition OpericeAgentTransport OpericeAgentTransport 10 KB 17.02.2016 04:51 Composition Operi		1 KD	Windows Datenda	ELICECTIT THESH	ConmanClient3.exe	111 KB	17.02.2016 04:51			
 					DeviceAgentTranspor	79 KB	17.02.2016 04:51			
Image: Section of States 151 KB 17.02.2016 04:51 Image: Image					eDbaTI 3.dll	110 KB	17.02.2016 04:51			
 firmware.bin 10 KB 17.02.2016 04:51 FTPD.tmp 1KB 20.0206 17:01 initobj.dat 0 KB 29.01.2006 17:01 iport.exe 40 KB 21.10.2015 06:08 iport.exe 30.62 XEB initobj.dat KB 17.02.2016 04:51 FireD.tmp KB 21.10.2015 06:08 MSDIA110.dll 645 KB 17.02.2016 04:51 msvsmon.exe.config 1KB 17.02.2016 04:51 msvsmon.exe.config KB 17.02.2016 04:51 msvsmon.exe.config msvsmon.exe.config msvsmon.exe.config KB 17.02.2016 04:51 msvsmon.exe.config msvsmon.exe.config msvsmon.exe.config KB 17.02.2016 04:51 msvsmon.exe.config KB Tor.2016 04:51 time.exe 44 KB 17.02.2016 04:51<					edm3.exe	151 KB	17.02.2016 04:51			
Image: State of the state					firmware.bin	10 KB	17.02.2016 04:51			
Initedbj.dat 0 KB 29.01.2006 17:01 Initedbj.dat 0 KB 29.01.2001 06:08 Initedbj.dat 0 KB 21.02.015 06:06 Initedbj.dat 10 KB 17.02.2016 04:51 Initedbj.dat 10 KB 17.02.2016 04:51 Insvismon.exe 94 KB 17.02.2016 04:51 Insvismon.exe 28 KB 21.10.2015 06:08 Intestat.exe 36 KB 23.060 KB Intestat.exe 44 KB 21.10.2015 06:08 Intestat.exe 44 KB 21.10.2015 06:08 Intestat.exe 44 KB 21.10.2015 06:06 Intestat.exe 44 KB 21.10.2015 06:06 Intestat.exe 44 KB 21.10.2015 06:06 Integration construction 3.dll 91 KB 17.02.2016 04:51 Integration construction 3.dll 91 KB 17.02.2016 04:51 Integration construction 3.dll 91 KB 17.02.2016 04:51 Integration constreation constreation construction 3.dlll					FTPD.tmp	1 KB	30.01.2006 01:28			
Importing.exe 40 KB 21.10.2015 06:08 Importing.exe 36 KB 21.10.2015 06:06 Importing.exe 36 KB 21.10.2015 06:06 Importing.exe 94 KB 17.02.2016 04:51 Immovime.exe 94 KB 17.02.2016 04:51 Immovime.exe 94 KB 17.02.2016 04:51 Importing.exe 28 KB 21.10.2015 06:08 Importing.exe 28 KB 21.10.2015 06:08 Importe.exe 36 KB 21.10.2015 06:08 Importe.exe 44 KB 21.10.2015 06:08 Importe.exe 44 KB 21.10.2015 06:08 Importe.exe 44 KB 21.10.2015 06:06 Importe.exe					initobi.dat	0 KB	29.01.2006 17:01			
Image: Specific and Specif					ipconfig.exe	40 KB	21.10.2015 06:08			
MSDIA110.dll 645 KB 17.02.2016 04:51 msvsmon.exe 94 KB 17.02.2016 04:51 msvsmon.exe 94 KB 17.02.2016 04:51 msvsmon.exe 28 KB 17.02.2016 04:51 msvsmon.exe 28 KB 21.10.2015 06:08 metstat.exe 36 KB 03.06.2016 03:50 mping.exe 28 KB 21.10.2015 06:08 motic.exe 44 KB 21.10.2015 06:06 Topto-nectionA3.dll 91 KB 17.02.2016 04:51 TopconectionA3.dll 91 KB 17.02.2016 04:51 TopconectionA3.dll 36 KB 21.00.1015 06:06 TopConectionA3.dll 36 KB 12.10.2015 06:06 TopConectionA3.dll 36 KB 17.02.2016 04:51 TopConectionA3.dll 36 KB 10.0.015 06:06 TopConectionA3.dll 36 KB 10.0.015 06:06 *					іруб.ехе	36 KB	21.10.2015 06:06			
Immsvsmon.exe 94 KB 17.02.2016 04:51 Immsvsmon.exe.config 1 KB 17.02.2016 04:51 Immsvsmon.exe.config 1 KB 17.02.2016 04:51 Immsvsmon.exe.config 28 KB 21.10.2015 06:08 Immsvsmon.exe. 28 KB 21.10.2015 06:08 Immsvsmon.exe. 28 KB 21.10.2015 06:08 Immsvsmon.exe. 44 KB 21.10.2015 06:08 Immsvsmon.exe. 44 KB 21.10.2015 06:06 Immsvsmon.exe.exe					MSDIA110.dll	645 KB	17.02.2016 04:51			
Important msvsmon.exe.config 1 KB 17.02.2016 04:51 Implement ndisconfig.exe 28 KB 21.10.2015 06:08 Implement 44 KB 21.10.2015 06:06 Implement 91 KB 17.02.2016 04:51 Implement 36 KB 21.10.2015 06:06 Implement 0 B of 3.780 KB in 0 of 31					msvsmon.exe	94 KB	17.02.2016 04:51			
Image: minisconfig.exe 28 KB 21.10.2015 06:08 Image: minisconfig.exe 36 KB 03.06.2016 03:50 Image: minisconfig.exe 28 KB 21.10.2015 06:08 Image: minisconfig.exe 28 KB 21.10.2015 06:06 Image: minisconfig.exe 28 KB 21.10.2015 06:06 Image: minisconfig.exe 24 KB 21.10.2015 06:06 Image: minisconfig.exe 44 KB 21.10.2015 06:06 Image: minisconfig.exe 17.02.2016 04:51 Image: minisconfig.exe 08 of 3.780 KB in 0 of 31					msvsmon.exe.config	1 KB	17.02.2016 04:51			
Image: metstat.exe 36 KB 03.06.2016 03:50 Image: metstat.exe 28 KB 21.10.2015 06:08 Image: metstat.exe 28 KB 21.10.2015 06:06 Image: metstat.exe 44 KB 21.10.2015 06:06 Image: metstat.exe 44 KB 21.10.2015 06:06 Image: metstat.exe 44 KB 17.02.2016 04:51 Image: metstat.exe 36 KB 17.02.2015 06:06 Image: metstat.exe 36 KB 21.02015 06:06 Image: metstat.exe 0 B of 3.780 KB in 0 of 31 V					ndisconfig.exe	28 KB	21.10.2015 06:08			
▲ 28 KB 21.10.2015 06:08 ■ route.exe 44 KB 21.10.2015 06:06 ▲ TcpConnectionA3.dll 91 KB 17.02.2016 04:51 ▲ ■ 17.02.2016 04:51 > ■ ■ 17.02.2016 04:51 > ■ ■ 17.02.2016 04:51 > ■ ■ 0 B of 3.780 KB in 0 of 31 >					netstat.exe	36 KB	03.06.2016 03:50			
Image: Contraction A3.dll 91 KB 17.02.2016 06:06 C Stimezones.dll 36 KB 21.10.2015 06:06 Stimezones.dll 36 KB 21.10.2015 06:06 ✓					ing.exe	28 KB	21.10.2015 06:08			
Image: Construction A3.dll 91 KB 17.02.2016 04:51 Image: Construction A3.dll 91 KB 17.02.2016 06:06 ✓ 13.069 KB of 33.069 KB in 6 of 6 0 B of 3.780 KB in 0 of 31 ✓ ✓					Toute.exe	44 KB	21.10.2015 06:06			
Image: Solution of the					StepConnectionA3.dll	91 KB	17.02.2016 04:51			
33.069 KB of 33.069 KB in 6 of 6 0 B of 3.780 KB in 0 of 31	<			>	timezones.dll	36 KB	21.10.2015 06:06			~
	33.069 KB of 33.069 KB in 6 of 6				0 B of 3.780 KB in 0 of 31					

Fig. 29: WinSCP – selecting folders

here ===================================	BEN-Lx-WinCE - WinSCP –
Local Mark Files Commands Session Options Remote Help	
🛺 😤 🔁 Synchronize 🥃 🧬 🗃 🖓 🖓 Queue 🗸 🛛 Transfer Settings Default	• 1 🚑 •
	Windows 🔹 🤗 🗔 👍 🗸 🚓 🖓 🔂 🖓 🖓 Find Filer 🦻
Upload - 📝 Edit - 🗙 🔂 La Properties 📑 🕼 🕂 🖃 🕅	I 🔐 Download 🔹 🕼 Edit 🔹 🗶 🕼 L 🖉 Properties 📑 🛅 🕼 I 🛨 🖃 🕅
Upload TBEN-Lx-4 RFID-8DXP-WIN\UpdateWinCe2110customer\update_win_2.1.1	/Windows
Delta della	Name Size Changed Rights Owner
directory 21.07.2017 09:11:05	🛓
Upload selected local file(s) to vs-Batchda 22.02.2017 11:46:48	📙 Keys 29.01.2006 17:01
EBOOTND.nb0 19.06.2017 12:43:52	83000xxx_V0.1.4.0.hex 25 KB 20.04.2016 05:18
NK.nb0 32.768 KB NB0-Datei 19.06.2017 12:44:22	ceconfig.h 12 KB 03.06.2016 03:48
SOS_NK.bat 1 KB Windows-Batchda 22.02.2017 11:47:14	E clientshutdown3.exe 64 KB 17.02.2016 04:51
xldmand.1block 44 KB 1BLOCK-Datei 19.06.2017 12:43:52	CMAccept3.exe 66 KB 17.02.2016 04:51
Skildmand.bat 1 KB Windows-Batchda 22.02.2017 11:45:44	E cmd.exe 76 KB 21.10.2015 06:06
	ConmanClient3.exe 111 KB 17.02.2016 04:51
	DeviceAgentTranspor 79 KB 17.02.2016 04:51
	BeDbgTL3.dll 110 KB 17.02.2016 04:51
	edm3.exe 151 KB 17.02.2016 04:51
	firmware.bin 10 KB 17.02.2016 04:51
	FTPD.tmp 1 KB 30.01.2006 01:28
	initobj.dat 0 KB 29.01.2006 17:01
	ipconfig.exe 40 KB 21.10.2015 06:08
	■ ipv6.exe 36 KB 21.10.2015 06:06
	MSDIA110.dll 645 KB 17.02.2016 04:51
	msvsmon.exe 94 KB 17.02.2016 04:51
	msvsmon.exe.config 1 KB 17.02.2016 04:51
	andiscontig.exe 28 KB 21.10.2015 00:08
	netstat.exe 50 KB 05.00.2010 05:30
	Pinglexe 20 ND 21.10.2015 06:06
	TrpConnection A2 dll 01 KP 17.02.2016 04:51
<	timezones dll 36 KB 21 10 2015 06:06
33,069 KB of 33,069 KB in 6 of 6	0 B of 3.780 KB in 0 of 31
	A FTP 0.03:50

► Click "Upload" to transfer the files to the device.

Fig. 30: Transferring files to the device



► Update the view in PuTTY.

🛃 192.168.1.1	1 - PuTTY		
02/17/16	04:51a	154176 edm3	3.exe
02/17/16	04:51a	660368 MSD1	IA110.dll
02/17/16	04:51a	96144 msvs	smon.exe
02/17/16	04:51a	364 msvs	smon.exe.config
02/17/16	04:51a	1144216 VSDe	ebugEng.dll
02/17/16	04:51a	594848 vsde	ebugeng.impl.dll
02/14/17	05:27a	24576 upda	ate.exe
02/14/17	05:27a	32768 XMC	Load.exe
02/14/17	05:27a	20480 XMC	_CheckVersion.exe
04/19/16	12:23a	9366 XMC1	1x_ASCLoader.hex
09/06/16	10:15p	25722 8300	00xxx_V0.1.7.0.hex
Found	32 fil	e(s). Total size 3894719 bytes.	
1 Dir	(s) 124	346368 bytes free	
\windows>	dir		
Direct	cory of	\windows	
01/01/06	12.10-	22554422 38	n h0
01/01/06	12:100	53554432 NK.I	NK bat
01/01/06	12:150	/3 US_1	NK.Dat
01/01/06	12:100	196 FIFE	b. cmp
01/01/06	04:00a		cobj.dat
10/21/15	04:004	<dir> Reys</dir>	3
10/21/15	04:20-	//824 Cmd.	.exe
10/20/10	04:298	40960 cime	ezones.dii
10/21/15	06:00a	20072 ping	g.exe
10/21/15	06.000	10500 ipc	acopfig ave
10/21/15	06:064	45056 rout	te eve
10/21/13 01/20/17	00.004	36864 pete	stat ava
10/21/15	06:08a	28672 trad	cert eve
10/21/15	06:06a	36864 invi	6.exe
$\frac{10}{21}$	02:36a	11614 cec	onfig h
02/17/16	04:51a	9932 firm	mware bin
02/17/16	04.51a	65136 clia	entshutdown3 eve
02/17/16	04.51a	67160 CMA	ccent3 eve
02/17/16	04:51a	113256 Con	manClient3.exe
02/17/16	04:51a	80520 Devi	iceAgentTransport3.dll
02/17/16	04:51a	112208 eDbo	gTL3.dll
02/17/16	04:51a	92272 Top	ConnectionA3.dll
02/17/16	04:51a	214960 vsd	ebugeng.impl.resources.dll
02/17/16	04:51a	154176 edm3	3.exe
02/17/16	04:51a	660368 MSD1	TA110.d11
02/17/16	04:51a	96144 msvs	smon.exe
02/17/16	04:51a	364 msv	smon.exe.config
02/17/16	04:51a	1144216 VSD	ebuqEng.dll
02/17/16	04:51a	594848 vsde	ebugeng.impl.dll
02/14/17	05:27a	24576 upda	ate.exe
02/14/17	05:27a	32768 XMC	Load.exe
02/14/17	05:27a	20480 XMC	
04/19/16	12:23a	9366 XMC1	1x ASCLoader.hex
09/06/16	10:15p	25722 8300	00xxx V0.1.7.0.hex
			-
Found	34 fil	e(s). Total size 37449631 bytes.	
1 Dir	(s) 103	010304 bytes free	
\windows>			-

Fig. 31: PuTTY – updated view

► Start the firmware update: Enter OS_NK.bat and confirm with ENTER.



NOTE

Instead of OS_NK.bat, the firmware update can also be started using the commands xldrnand.bat and eboot.bat.

192.168.1.	1 - PuTTY		~					X
02/17/16	04.51a	the state of the	154176	edm3 eve		-		
02/17/16	04:51a		660368	MSDIA110 dl	1			
02/17/16	04:51a		96144	msvsmon.exe	-			
02/17/16	04:51a		364	msvsmon.exe	.config			
02/17/16	04:51a		1144216	VSDebugEng.	d11			
02/17/16	04:51a		594848	vsdebugeng.	impl.dll			
02/14/17	05:27a		24576	update.exe				
02/14/17	05:27a		32768	XMC Load.exe	2			
02/14/17	05:27a		20480	XMC CheckVer	rsion.exe	2		
04/19/16	12:23a		9366	XMC1x_ASCLos	ader.hex			
09/06/16	10:15p		25722	83000 xxx_ V0	.1.7.0.he	2X		
Found	32 file(s)	. Total size 389	4719 byte	23.				
1 Dir	(s) 1243463	68 bytes free						
\windows>	dir							
Direct	tory of \wi	ndows						
01/01/06	12.16p		33554432	NK nb0				
01/01/06	12:10p		73	OS NK.bat				
01/01/06	12:160		496	FTPD.tmp				
01/01/06	04:00a		0	initobi.dat				
01/01/06	04:00a	<dir></dir>		Kevs				
10/21/15	06:06a		77824	cmd.exe				
10/28/16	04:29a		40960	timezones.d	11			
10/21/15	06:08a		28672	ping.exe				
10/21/15	06:08a		40960	ipconfig.exe	2			
10/21/15	06:08a		28672	ndisconfig.	exe			
10/21/15	06:06a		45056	route.exe				
01/20/17	02:37a		36864	netstat.exe				
10/21/15	06:08a		28672	tracert.exe				
10/21/15	06:06a		36864	ipv6.exe				
01/20/17	02:36a		11614	ceconfig.h				
02/17/16	04:51a		9932	firmware.bin				
02/17/16	04:51a		65136	clientshutdo	own3.exe			
02/17/16	04:51a		67160	CMAccept3.ex	xe			
02/17/16	04:51a		113256	ConmanClient	t3.exe			
02/17/16	04:51a		80520	DeviceAgent:	Fransport	t3.dII		
02/17/16	04:51a		112208	ebbgil3.dll	- 10 -111			
02/17/16	04:51a		92272	1cpConnect10	onA3.dll		-17.7	
02/17/16	04:51a		214960	vsdebugeng.	impi.resc	ources	. 011	
02/17/16	04:51a		1541/6	edma.exe				
02/17/16	04:514		060300	msbiailo.di.	L			
02/17/16	04:514		26144	mayamon ava	config			
02/17/16	04:51a		1144216	VSDebugEng (.conrig			=
02/17/16	04:51a		594848	vsdebugeng	impl dll			
02/14/17	05:27a		24576	update.exe	- aptruct			
02/14/17	05:27a		32768	XMC Load.ex	-			
02/14/17	05:27a		20480	XMC CheckVe	rsion.exe	2		
04/19/16	12:23a		9366	XMC1x ASCLO	ader.hex			
09/06/16	10:15p		25722	83000xxx V0	.1.7.0.he	2x		
Found 1 Dir	34 file(s)	. Total size 374	49631 byt	tes.				
1 011	(5) 1050105	or bytes free						
\windows>	OS_NK.bat							+

Fig. 32: Starting the firmware update



The update in progress is shown with "update NK...".

🛃 192.168.1.:	1 - PuTTY	the beauties have be		-		x	
02/17/16	04:51a	1144216	VSDebugEng.	d 1 1		~	
02/17/16	04:51a	594848	vsdebugeng.	impl.dll			
02/14/17	05:27a	24576	update.exe				
02/14/17	05:27a	32768	XMC_Load.ex				
02/14/17	05 : 27a	20480	XMC_CheckVe:	rsion.exe			
04/19/16	12:23a	9366	XMC1x_ASCLo	ader.hex			
09/06/16	10:15p	25722	83000xxx_V0	.1.7.0.hex			
Found 1 Dir	32 file(s) (s) 1243463). Total size 3894719 byt 368 bytes free	es.				
\windows>	dir						
Direct	tory of \w	indows					
01/01/06	12:16p	33554432	NK.nb0				
01/01/06	12:15p	73	OS_NK.bat				
01/01/06	12:16p	496	FTPD.tmp				
01/01/06	04:00a	(initobj.dat				
01/01/06	04:00a	<dir></dir>	Keys				
10/21/15	06:06a	77824	cmd.exe				
10/28/16	04:29a	40960	timezones.d	11			
10/21/15	06:08a	28672	ping.exe				
10/21/15	06:08a	40960	ipconfig.ex	2			
10/21/15	06:08a	28672	ndisconfig.	exe			
10/21/15	06:06a	45056	route.exe				
01/20/17	02:37a	36864	netstat.exe				
10/21/15	06:08a	28672	tracert.exe				
10/21/15	06:06a	36864	ipv6.exe				
01/20/17	02:36a	11614	ceconfig.h				
02/17/16	04:51a	9932	firmware.bi	n			
02/17/16	04:51a	65136	clientshutd	own3.exe			
02/17/16	04:51a	6/16	CMACCept3.e	xe			
02/17/16	04:51a	113256	ConmanClien	c3.exe			
02/17/16	04:514	11220	DeviceAgent.	ransports.	111		
02/17/16	04:518	112200	EDDG1L3.dll	116 2/10			
02/17/16	04:51a	92272	Tepconnectio	impl pogour			
02/17/16	04:514	214900	vsaebugeng.	Impi.resourc	Ses.dll		
02/17/16	04:51a	1541/6	MEDIATIO AT	1			
02/17/16	04.51a	96144	matramon ava	L			
02/17/16	04:51a	364	mawamon exe	config			
02/17/16	04:51a	1144216	VSDebugEng				
02/17/16	04:51a	594848	vsdebugeng	impl.dll			
02/14/17	05:27a	24576	undate.exe	Impirali			
02/14/17	05:27a	3276	XMC Load.ex	-			
02/14/17	05:27a	20480	XMC CheckVe	- rsion.exe		≡	
04/19/16	12:23a	9366	XMC1x ASCLO	ader.hex			
09/06/16	10:15p	25722	83000xxx V0	.1.7.0.hex			
Found 1 Dir	34 file(s) (s) 1030103). Total size 37449631 by 304 bytes free	rtes.				
\windows>	OS NK.bat						
\windows>	rem Updat	e OS WIN-CE im Verzeichni	s windows				
\windows>	update.ex	e i \windows\NK.nb0					
update NK							
						-	

Fig. 33: Firmware update in progress

🛃 192.168.1.1	L - PuTTY	time to manufact house in		
02/14/17	05:27a	24576	update.exe	~
02/14/17	05:27a	32768	XMC_Load.exe	
02/14/17	05:27a	20480	XMC CheckVersion.exe	
04/19/16	12:23a	9366	XMC1x_ASCLoader.hex	
09/06/16	10:15p	25722	83000xxx_V0.1.7.0.hex	
Found	22 filo(a)	Total gize 2804710 but		
1 Dir	(s) 1243463	68 bytes free		
\windows>	dir			
Direct	ory of \wi	ndows		
01/01/06	12:16p	33554432	NK.nb0	
01/01/06	12:15p	73	OS NK.bat	1
01/01/06	12:16p	496	FTPD.tmp	
01/01/06	04:00a		initobj.dat	
01/01/06	04:00a	<dir></dir>	Кеуз	
10/21/15	06:06a	77824	cmd.exe	
10/28/16	04:29a	40960	timezones.dll	
10/21/15	06:08a	28672	ping.exe	
10/21/15	06:08a	40960	ipconfig.exe	
10/21/15	06:08a	28672	ndisconfig.exe	
10/21/15	06:06a	45056	route.exe	
01/20/17	02:37a	36864	netstat.exe	
10/21/15	06:08a	28672	tracert.exe	
10/21/15	06:06a	36864	ipv6.exe	
01/20/17	02:36a	11614	ceconfig.h	
02/17/16	04:51a	9932	firmware.bin	
02/17/16	04:51a	65136	clientshutdown3.exe	
02/17/16	04:51a	67160	CMAccept3.exe	
02/17/16	04:51a	113256	ConmanClient3.exe	
02/17/16	04:51a	80520	DeviceAgentTransport3.dll	
02/17/16	04:51a	112208	eDbgTL3.dll	
02/17/16	04:51a	92272	TcpConnectionA3.dll	
02/17/16	04:51a	214960	vsdebugeng.impl.resources.dll	
02/17/16	04:51a	154176	edm3.exe	
02/17/16	04:51a	660368	MSDIA110.dll	
02/17/16	04:51a	96144	msvsmon.exe	
02/17/16	04:51a	364	msvsmon.exe.config	
02/17/16	04:51a	1144216	VSDebugEng.dll	
02/17/16	04:51a	594848	vsdebugeng.impl.dll	
02/14/17	05:27a	24576	update.exe	
02/14/17	05:27a	32768	XMC_Load.exe	
02/14/17	05 : 27a	20480	XMC_CheckVersion.exe	
04/19/16	12:23a	9366	XMC1x_ASCLoader.hex	
09/06/16	10:15p	25722	83000xxx_V0.1.7.0.hex	-
Found 1 Dir	34 file(s) (s) 1030103	. Total size 37449631 by 04 bytes free	tes.	
	0.0 100			
\windows>	OS_NK.bat			
\windows>	rem Update	OS WIN-CE im Verzeichni	5 WINDOWS	
\windows>	update.exe	1 \windows\NK.nb0		
update NK.				
done				
(Windows>				
\windows>				Ψ.

The firmware update is complete when the "done" message appears.

Fig. 34: Firmware update complete

► Complete the firmware update: Reset the device's voltage.



Checking the Firmware Status

From version 2.1.0.0, the firmware status can be displayed via the Turck Service Tool.

- ► Launch the Turck Service Tool.
- ► Click "Search".
- The device is displayed. The firmware status is listed under "Version".

Turck S	ervice Tool, Vers. 3.0.	0		-	-	-		1				
Yo	ur Global Auto	omation Pa	rtner									
		Sinacioni a										
Q	\square	(j) (j)	් ද	<u>۲</u>	DE .			EIP	5		٢	
Suchen (F5) Ändern (F2) Wink (F3) Aktionen (F4) Zwischenablage Sprache Expertenmodus AN Starte DHCP (F6) Konfiguration (F7) ARGEE (F8) Schließen												
Nr.	MAC-Adresse	Name	IP-Adresse	Netzmaske	Gateway	Modus	Gerätetyp	Version	Adapter	ARGEE	Protokoll	
= 1	00:07:46:FF:A2:5B		<u>192.168.1.100</u>	255.255.255.0	192.168.1.1	ROTARY	TBEN-L5-4RFID-8DXP-WIN	2.1.1.0	192.168.1.60	-	Turck	



Repairs 11

The device is not intended for repair by the user. If the device is faulty, please take it out of operation. If you are returning the device to Turck, please note our return terms and conditions.

Returning Devices 11.1

If a device has to be returned, bear in mind that only devices with a decontamination declaration will be accepted. This is available at

http://www.turck.de/en/retoure-service-6079.php

and must be filled in completely and affixed to the outside of the packaging such that it is secure and cannot be impaired by adverse weather.

Disposal 12



The device must be properly disposed of, not in general household waste.

13 Technical Data

Technical data	
Supply	
Power supply	24 VDC
Admissible range	1830 VDC
Total current	V1 max. 8 A, V2 max. 9 A at 70 °C per module
RFID supply	2 A per channel at 70 °C
Sensor/actuator supply	2 A per slot at 70 °C
Potential separation	V1 and V2 voltage groups galvanically isolated
Dielectric strength	Up to 500 VDC V1 and V2 over Ethernet
Power loss	Typically ≤ 5 W
System description	
Processor	Cortex A8 800 MHz
Memory	256 MB Flash ROM; 512 MB DDR3 RAM
Real-time clock	Yes
Operating system	Windows Embedded Compact 2013
System data	
Transmission rate	Ethernet 10 Mbps/100 Mbps
Connectivity	2 × M12, 4-pin, D-coded
RFID	
Number of channels	4
Connectivity	M12, 5-pin
Supply	2 A per channel at 70 °C, short-circuit proof
Digital inputs	
Number of channels	8
Connectivity	M12, 5-pin
Input type	PNP
Type of input diagnostics	Channel diagnostics
Switching threshold	EN 61131-2 Type 3, PNP
Low-level signal voltage	< 5 V
High-level signal voltage	> 11 V
Low-level signal current	< 1.5 mA
High-level signal current	> 2 mA
Potential separation	Galvanic isolation to P1/P2
Dielectric strength	Up to 500 VDC (V1 and V1 over Ethernet)
Cable length	Max. 50 m



Technical data	
Digital outputs	
Number of channels	8
Connectivity — outputs	M12, 5-pin
Output type	PNP
Type of output diagnostics	Channel diagnostics
Output voltage	24 VDC from potential group
Output current per channel	2.0 A, short-circuit proof, max. 4.0 A per port
Simultaneity factor	0.56
Load type	Resistive, inductive, lamp load
Short-circuit protection	Yes
Potential separation	Galvanic isolation to P1/P2
Dielectric strength	Up to 500 VDC (V1 and V1 over Ethernet)
Standard/directive conformity	
Vibration test	Acc. to EN 60068-2-6
Acceleration	Up to 20 g
Shock test	Acc. to EN 60068-2-27
Drop and topple	Acc. to IEC 60068-2-31/IEC 60068-2-32
Electromagnetic compatibility	Acc. to EN 61131-2
Approvals and certificates	CE
UL conditions	cULus LISTED 21 W2, Encl.Type 1 IND.CONT.EQ.
General information	
Dimensions (W x L x H)	$60.4 \times 230.4 \times 39 \text{ mm}$
Operating temperature	-40 °C to +70 °C
Storage temperature	-40 °C to +70 °C
Operating altitude	Max. 5000 m
Protection class	IP65/IP67/IP69K
MTTF	
Housing material	PA6-GF30
Housing color	Black
Window material	Lexan
Screw material	303 stainless steel
Halogen-free	Yes
Mounting	2 mounting holes, Ø 6.3 mm







www.turck.com