

PROPRIETARY NOTICE

All rights reserved by Janz Tec AG.

No parts of this technical manual may be modified, copied or reproduced in any form or by any means for commercial use without the prior written permission of Janz Tec AG, Germany.

All instructions, information and specification contained in this manual are for reference only and remain subject to change without announcement.

emPC-A/RPI

embedded PC

(Hardware and Software Manual)

Version 1.5

refers to product revision no.

1.2

Title: emPC-A/RPI embedded PC
File: j:\as\entwicklung\projekte\hw\empc\empc-
arpi\doc\manual\empc_arpi_hardware_manual.doc
Pattern: \\ntserv5\janzgrp\as\entwicklung\formulare\entwicklung (ew)\ew_manual.dot
Created: Stefan Althöfer, 06.03.2015
Last Update: Stefan Althöfer, 24.02.2016

© Janz Tec AG 2015

Im Dörener Feld 8
D-33100 Paderborn, Germany

Tel.: +49-5251-1550-0
FAX: +49-5251-1550-190
email: support@janztec.com
Internet: www.janztec.com

Contents

1	Introduction	5
1.1	Features	5
1.2	Functional Overview	6
2	Safety Instructions	7
2.1	Installation and Maintenance	7
2.2	Ambient and Environmental Conditions.....	7
3	Installation	8
3.1	Mounting	8
3.2	Connectors and Operators.....	9
3.2.1	POWER IN.....	9
3.2.2	Graphics connector (HDMI)	9
3.2.3	Ethernet Interface	10
3.2.4	USB host interfaces (USB1-USB4).....	10
3.2.5	Console (RS232)	11
3.2.6	Serial Port (RS232/RS485).....	11
3.2.7	CAN	11
3.2.8	Digital IO	12
3.2.9	Audio Video.....	12
4	Maintenance	13
4.1	Micro SDcard Replacement	14
4.2	Battery Replacement	14
5	Operating System and Drivers	15
5.1	Standard Raspbian Operating System	15
5.1.1	Installation.....	15
5.2	emPC-A/RPI Driver Installation Script	15
5.2.1	Installation Instructions	16
5.2.2	Driver Details	16
5.2.2.1	spi-bcm2835 : SPI driver (optimized for emPC-A/RPI)	16
5.2.2.2	mcp2515 : CAN driver (optimized for emPC-A/RPI)	16
5.2.2.3	sc16is7xx : RS232/RS485 driver (optimized for emPC-A/RPI).....	17
5.2.2.4	rtc-ds1307 : RTC	17
5.3	Janz Tec AG Linux Image for emPC-A/RPI.....	18
5.4	General Information	18
5.5	Optional Software.....	18
5.5.1	CODESYS Control for Raspberry Pi SL Information	18
5.5.1.1	Information.....	19
5.5.1.2	License Information	19
6	Programming Information	20
6.1	User LEDs.....	20
6.2	I ² C Bus	20
6.3	SPI Bus	20
6.4	CAN.....	21
6.5	Serial Port (RS232/RS485).....	21
6.6	Digital IO	21
6.7	Console RS232.....	21
6.8	Overclocking	22
7	Appendices	23

7.1	Technical Data	23
7.2	References	25
7.3	Dimensions	26
7.4	Raspberry GPIO Header	27
7.5	EMC Cable Installation	28
7.6	Product History	29
7.7	Manual History	29

List of Figures

figure 1: emPC-A/RPI block diagram	6
figure 2: emPC-A/RPI mounting options: Desk-top, Wall and DIN Rail	8
figure 3: digital IO detailed schematic	12
figure 4: emPC-A/RPI case bottom part removal	13
figure 5: emPC-A/RPI micro SDcard location.....	13
figure 6: Removing and replacing coin cell battery	14
figure 7: housing dimensions.....	26
figure 8: EMC Shielding.....	28

List of Tables

table 1: Power Supply connector pins	9
table 2: Ethernet connector	10
table 3: USB host connector	10
table 4: Console (RS232) connector	11
table 5: Serial Port connector pins	11
table 6: CAN connector pins.....	11
table 7: Digital IO connector pins	12
Table 8: Internal I ² C devices	20

About this Manual






This is the hardware manual for the emPC-A/RPI embedded PC.

Conventions

If numbers are specified in this manual, they will be either decimal or hexadecimal. We use C-notation to identify hexadecimal numbers (the 0x prefix).

If we refer to low active signal names, they will be suffixed by a “#” character.

Some parts of the manual contains notices you have to observe to ensure your personal safety, or to prevent damage to property. These are visually marked with the following alert symbols:

	DANGER Indicates that death or severe personal injury <i>will</i> result if proper precautions are not taken.
	WARNING Indicates that death or severe personal injury <i>may</i> result if proper precautions are not taken.
	CAUTION Indicates that <i>minor</i> personal injury can result if proper precautions are not taken.
	NOTICE Indicates that damage to equipment can result if proper precautions are not taken.
	Indicates information that we think you should have read to save your time by avoiding common problems. Important suggestions that should be followed will also be marked with this sign.

Acronyms and Abbreviations

EMC	Electromagnetic capability.
ESD	Electrostatic discharge.
GND	System ground potential. Inside the product this is connected to the metal housing, which might be connected to protective earth by the installation. There exist some isolated reference grounds for communication interfaces or IO. These reference signals are referred to as GND-x, where x indicates function.
SELV	Safety extra low voltage.

1 Introduction

1.1 Features

- Raspberry PI 2 Model B powered
 - BCM2836 processor, up to 900MHz quad-core ARM Cortex-A7 CPU
 - 1GB LPDDR2 SDRAM
 - HDMI graphics output
 - 10/100 Mbit/s Ethernet
 - 4 x USB 2.0
 - Micro SDcard socket
- CAN, ISO/DIN 11898-2 (isolated from logic), jumper switchable 120 Ohm termination resistor
- Console, 9 pin DSUB
- Serial port, RS232 (4 wire) or RS485 (jumper selectable), jumper switchable 120 Ohm termination resistor for RS485
- RTC with battery backup
- 4 x digital OUT (0.5 A max.), 10..32V externally powered (isolated from logic)
- 4 x digital IN (isolated from logic)
-
-

1.2 Functional Overview

The processing core of the emPC-A/RPI is an unmodified Raspberry PI 2 B board. A Janz Tec specific add-on board implements the specific features. Refer to figure 1 for the block diagram.

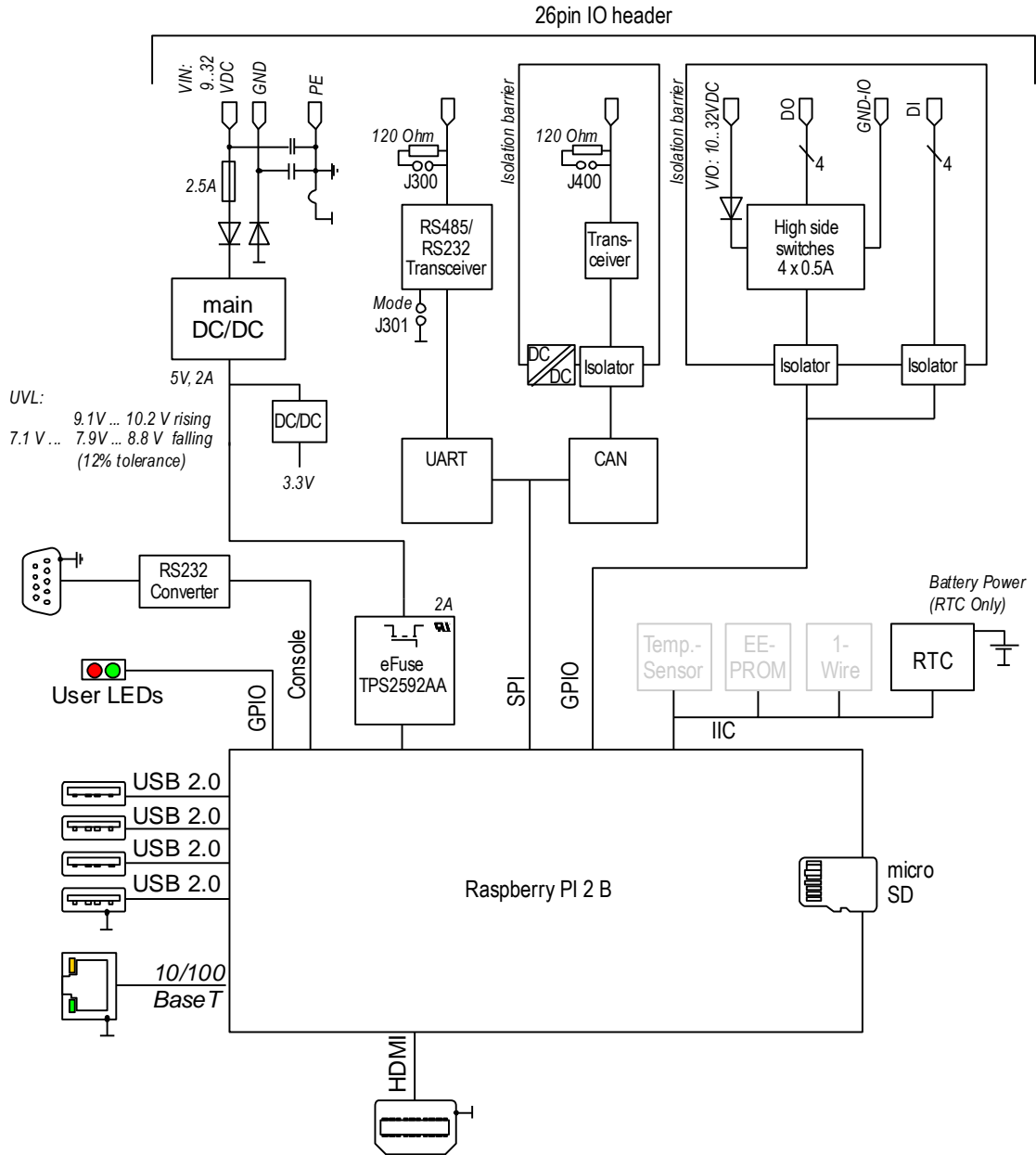


figure 1: emPC-A/RPI block diagram

2 Safety Instructions

Refer to page iv for explanation of the warning notice system.

The product described in this documentation may be operated only by personnel qualified for the specific task in accordance with the relevant documentation for the specific task, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products.

2.1 Installation and Maintenance

**DANGER: Electrical Shock**

Danger to life.
This product operates with 9.32 V DC SELV power supply. Do not connect this product to an improper power supply.

**DANGER: Electrical Shock**

Danger to life.
The IO interfaces (connectors) of the product are only suited to be connected to SELV circuits. Use interfaces (connectors) for their intended use only.

**CAUTION: Explosive Risk**

The installed computer board is equipped with a Lithium battery.
Danger of explosion if battery is incorrectly replaced. Replace only with battery of the same or equivalent type.

**WARNING: Burns Hazard**

The product generates considerable amount of heat. The housing transports this heat to the environment and therefore gets hot. **Caution when touching the housing, burns hazard!**

2.2 Ambient and Environmental Conditions

**CAUTION: Damage**

Do not operate the product beyond the specified ambient conditions

**DANGER: Explosive Risk**

Do not operate the product in potentially explosive atmosphere.

**NOTICE: EMI**

This product is a class A device. This product may cause radio interference. In this case the user must take adequate measures.

3 Installation



WARNING: Burns Hazard

The product generates considerable amount of heat. The housing transports this heat to the environment and therefore gets hot. **Caution when touching the housing, burns hazard!**

The product can be operated with DC power supply from 9 to 32 V.

3.1 Mounting

The emPC-A/RPI is intended for wall and 35mm DIN-rail mount, but can as well be used on the desk-top. Refer to figure 2 for the recommended mounting orientation.

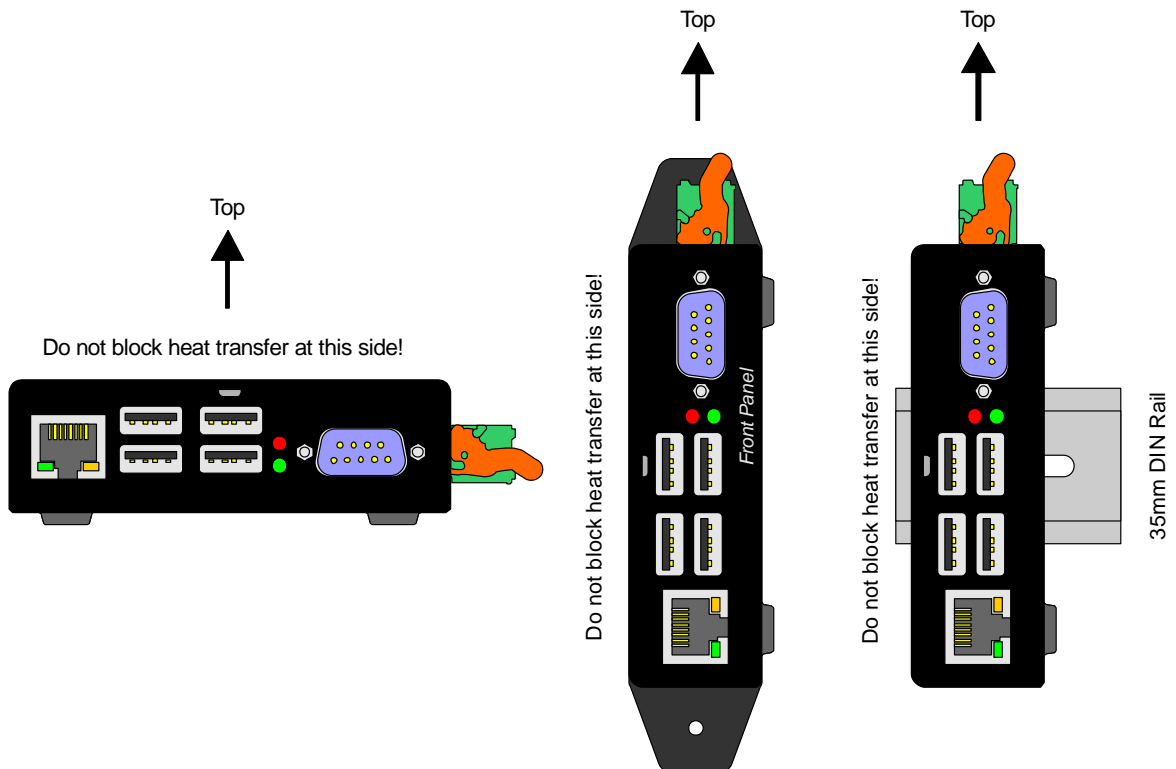


figure 2: emPC-A/RPI mounting options: Desk-top, Wall and DIN Rail

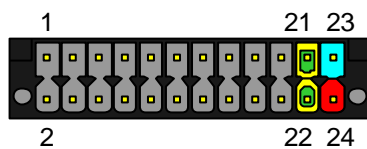
3.2 Connectors and Operators

The 24 pin multi connector provides interface to many of the emPC-A/RPI interfaces that are described later.

A suitable mating connector is Phoenix Contact order number 1790580. A Model witch screw latching is available with order number 1790399.

3.2.1 POWER IN

The system power supply is located on the multi connector.



21	PE	22	PE
23	GND	24	VIN (9..32 VDC)

table 1: Power Supply connector pins



DANGER

The product may only be operated with power supplies which can be considered SELV circuits.

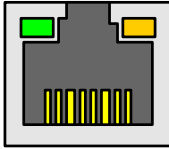
Pins 21 and 22 should be connected to EMI protective earth (PE). These contacts are internally connected to the DC power supply EMI filter and to the D-SUB shell of the console (RS-232). The signal is doubled to prevent false connection, it only requires one connection. In locations where no EMI noise is present, the PE connection is not required. The PE has no safety function.

3.2.2 Graphics connector (HDMI)

Provided by Raspberry PI board.

3.2.3 Ethernet Interface

Ethernet interfaces of the Raspberry PI. The Ethernet physics is 10/100BaseT, available through the shielded modular jack. Twisted pair cable can be used to connect to this port.

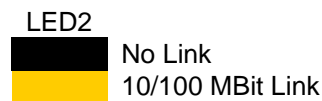


1	RD+	5	n.c.
2	RD-	6	TD-
3	TD+	7	n.c.
4	n.c.	8	n.c.

1

table 2: Ethernet connector

The two LEDs indicate Ethernet status as follows:



NOTICE

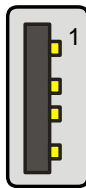
If your Ethernet cable is long (normative limit = 30 m) or leaves a building, then an additional protection against surges is required.

Add an external surge protector in such cases. E.g. ground the Ethernet cable shield where it enters the building or your cabinet.

If you just connect to a nearby switch, such countermeasures are not required.

3.2.4 USB host interfaces (USB1-USB4)

Two USB interfaces are available at the connector panel.



1	+5V, I _{MAX} = 500 mA, 1000 mA for all ports
2	USB-
3	USB+
4	GND

table 3: USB host connector



NOTICE

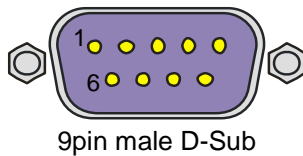
Although each port can deliver supply current of 500 mA, the overall load on USB1 to USB4 interfaces must be limited to 1000 mA to prevent power supply from overheating.



If you draw excessive power from USB ports or short circuit USB power supply outputs, then the Raspberry PI board will temporarily shut off the USB power supply and retry every 2 seconds.

3.2.5 Console (RS232)

The RS232 (Console) serial interface is provided by Raspberry PI board. The connector is a standard 9 pin D-SUB plug, but the interface is connected with two wires only, table 4 shows the pin out.

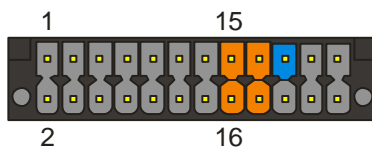


1	n.c.	6	n.c.
2	RxD	7	n.c.
3	TxD	8	n.c.
4	n.c.	9	n.c.
5	GND		

table 4: Console (RS232) connector

3.2.6 Serial Port (RS232/RS485)

The Serial Port is implemented by SC16IS740 UART. A multiprotocol transceiver is utilized to support both RS232 and RS485. The interface signals are located on the multi connector, refer to table 5.



15	TxD , B / D+	16	RTS , A / D-
17	RxD	18	CTS
19	GND	20	

table 5: Serial Port connector pins

RS232 mode is selected by shorting jumper J301, else RS485 mode is selected. The case has to be opened to access this jumper.

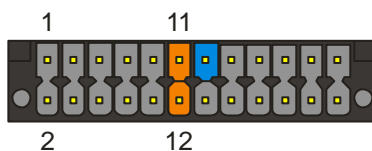
In RS232 mode, the signals TxD (out), RxD (in), RTS (out) and CTS (in) are provided.

In the RS485 mode the bidirectional differential signals D+ and D- are provided. When the transmitter is enabled and no data is send (MARK=1), then D+ > D-. Refer to section 6 (Programming Information) for more information about RS485 control.

An internal 120 Ohm termination resistor can be enabled between D+ and D- by shorting jumper J300. The case has to be opened to access this jumper.

3.2.7 CAN

The CAN interface is implemented with a MCP2515 controller chip. The interfaces is located on the multi connector, refer to table 6. The CAN interface signals support ISO 11898-2 (high speed) and are isolated from system logic.



11	CAN-H	12	CAN-L
13	GND-CAN	14	n.c.

table 6: CAN connector pins

An internal 120 Ohm CAN termination resistor is enabled by shorted jumper J400. To disable the termination, the case has to be opened to remove this jumper.

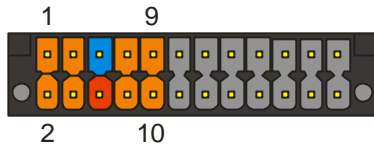


CAUTION

Factory setting: Jumper J400 is **not** set by default, and therefore the 120 Ohm termination resistor is disabled!

3.2.8 Digital IO

Digital input (DI1-DI4) and output (DO1-DO4) ports are provided on the multi IO connector (refer to table 7).



1	DO4	2	DO3
3	DO2	4	DO1
5	GND-IO	6	VIO (10..32 VDC)
7	DI4	8	DI3
9	DI2	10	DI1

table 7: Digital IO connector pins

The digital inputs and outputs are isolated from system power supply. Refer to figure 3 for detailed schematic.

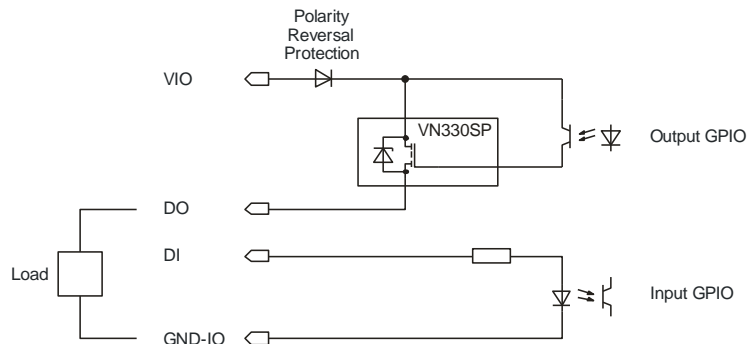


figure 3: digital IO detailed schematic

The digital input signals (DI1-DI4) refer to GND-IO and have an input impedance of about 5 kOhm. Switching level for *low* is $\leq 4V$ and *high* $\geq 9V$. The digital inputs do not require VIO to be present.

The digital outputs drive VIO power to the output pins (high side switch). The load must be connected between the output pin and the return of the VIO power supply.

The output driver supports resistive and inductive loads.



NOTICE

When using inductive loads, the output driver has internal clamping to demagnetize the load. This clamps the output pin to about 55 V below V_{IN} . When using this feature, very carefully evaluate that the demagnetization energy does not overheat the output driver. Parameters to check are the switching frequency, the load current, the load inductance and VIO voltage. Refer to the VN330SP datasheet for more information. If in doubt, add external clamping circuits.

3.2.9 Audio Video

The combined audio and analogue video connector is not accessible on the emPC-A/RPI.

4 Maintenance



NOTICE

Always follow common ESD practice when you service the product!

To open the housing, follow these steps (Refer to figure 4):

- Disconnect all connectors
- Remove both bolts on the front RS232 DSUB connector
- Remove all screws on the backside (DIN rain clip or mounting plate)
- Insert the tip of a small screwdriver in to the hole above the USB ports and unclip the two case parts
- Pull the lower case part away from the upper case

Task			Remarks
Replace μ SD-card			Remove DIN rail clip and/or wall mount bracket
Change jumper settings			Remove case bottom part
Replace Battery			Complete disassembly

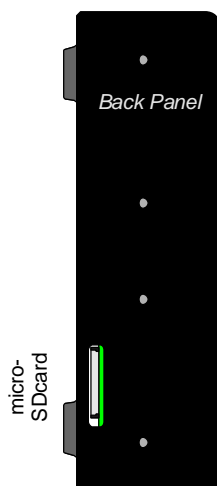


figure 5: emPC-A/RPI micro SDcard location

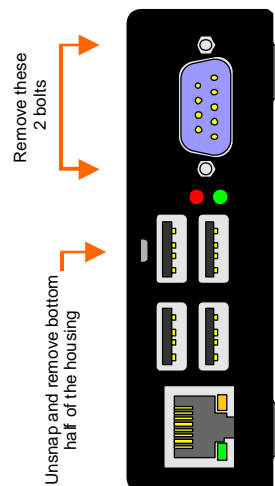


figure 4: emPC-A/RPI case bottom part removal



NOTICE

Remove the micro SDcard before you attempt to remove the Raspberry PI module from the housing.

4.1 Micro SDcard Replacement



NOTICE

Power must be turned off before removing or inserting the micro SDcard

Refer to figure 4 to identify the location of the Micro SDcard. Remove the DIN rail clip and/or the wall mount bracket if necessary.

4.2 Battery Replacement



CAUTION

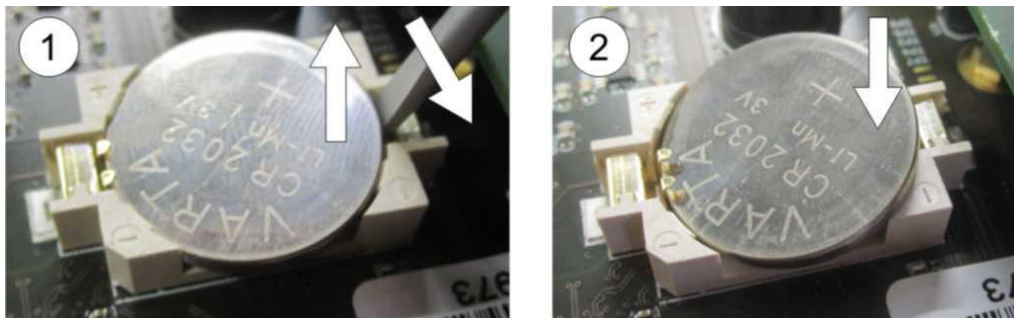
The installed computer board is equipped with a Lithium battery. Danger of explosion if battery is incorrectly replaced. Replace only with battery of the same or equivalent type (3-volt lithium coin cell battery).

- Do not attempt to recharge the battery.
- Do not disassemble, crush, puncture, short external contacts, or dispose of in fire or water.

Compatible battery type: CR2032 (3 Volt lithium coin cell battery)

The battery is used for backing up the system time when the power supply is removed.

1. Turn off the computer properly through the operating system, then turn off any external devices.
2. Disconnect the power supply from the power inlet and disconnect any external devices.
3. Disassemble the housing (refer chapter 4) and locate the battery on the Janztec board.
4. Remove the battery from the holder (See figure 6)
5. Insert the new battery (See figure 6)
6. Re-assemble the housing
7. See chapter 5.2.2.4 for important software information



1. Removal: Insert screwdriver at right side and bend so that the battery pops out. Use only gentle force, otherwise the battery holder might be damaged.
The use a plastic tool is preferred to avoid shorting the battery
2. Insertion: Align new battery to the left side of the holder and gently press down on the right side of the battery until the battery snaps into the holder.

figure 6: Removing and replacing coin cell battery

5 Operating System and Drivers

When you order the emPC-A/RPI with pre-installed software (Raspbian operating system), then our default Janz Tec image is pre-installed on the μ SD-card and the emPC-A/RPI driver package is already installed and fully configured. For more information, refer to 5.2.

Of course you can order the emPC-A/RPI without software, so you can choose a Raspberry Pi 2 B compatible operating system of your choice and install it manually on a μ SD card. (Raspbian recommended)

Refer to 5.1.1 on how to install an image from the official Raspberry Pi website.

5.1 Standard Raspbian Operating System

If the additional interfaces of the emPC-A/RPI (RTC, CAN and RS232/RS485) are not required, then the latest official Raspbian operating system is recommend for emPC-A/RPI systems. (Digital IO and LEDs are controllable by standard GPIO commands and console (RS232) is supported by Raspbian by default). To use all of the emPC-A/RPI features, refer to chapter 5.2 and 5.3.

5.1.1 Installation

For more information on how to prepare a μ SD-card to run the Raspbian operating system visit:

<https://www.raspberrypi.org/documentation/installation/installing-images/README.md>

Our Janz Tec image is based on the official RASPBIAN JESSIE LITE image from **2015-11-21** with kernel version **4.1**. (SHA-1: **97888fcd9bfbbae2a359b0f1d199850852bf0104**)

Download at: https://downloads.raspberrypi.org/raspbian_lite/images/ raspbian_lite-2015-11-24/

Default log in credentials:

Username: **pi**
Password: **raspberry**

(depending on your keyboard layout language, the last “y” might be “z” on your keyboard)

For more information visit the official Raspberry quick start guide:

<https://www.raspberrypi.org/help/quick-start-guide/>

5.2 emPC-A/RPI Driver Installation Script

**NOTICE**

Always create a backup of the μ SD card before installing this script!

**NOTICE**

This script is only compatible to Raspbian based installations!

To automatically install all configuration entries and drivers to enable the additional features of the emPC-A/RPI (RTC, CAN and RS232/RS485), we provide an installation script and installation instructions on: <https://github.com/janztec/empc-arpi-linux-drivers>

All source code modifications made by Janz Tec AG are publicly available at <https://github.com/janztec/empc-arpi-linux>

5.2.1 Installation Instructions

The installation requires that the emPC-A/RPI has booted successfully from the Raspbian image on the µSD-card prepared in chapter 5.1.1 and that you have created a backup copy of the µSD-card. For more installation instructions visit:

<https://github.com/janztec/empc-arpi-linux-drivers/blob/master/README.md>

5.2.2 Driver Details

The following drivers are automatically installed by our driver installation script:

5.2.2.1 *spi-bcm2835* : SPI driver (optimized for emPC-A/RPI)

CAN and RS232/RS485 controllers are connected by SPI to the Raspberry PI board and we have optimized the default Raspberry SPI (*spi-bcm2835*) driver for better performance.

5.2.2.2 *mcp2515* : CAN driver (optimized for emPC-A/RPI)

This driver installation includes a SocketCan compatible driver. In the configuration file `/etc/network/interfaces` the CAN device with the Linux device name **can0** is enabled by default (bus-on) and is configured to 500Kbit (bitrate 500000). To change the default bitrate, edit file `/etc/network/interfaces`, change the bitrate parameter and restart your system.

Here are some can-utils examples:

Print incoming can frames:

```
pi@raspberrypi ~$ candump can0
```

Send random can frames:

```
pi@raspberrypi ~$ cangen can0
```

Show can bus load information:

```
pi@raspberrypi ~$ canbusload can0@5000000 -r -t -b -c
```



NOTICE

On high CAN-bus utilization, incoming CAN frames might get dropped because of an overrun in the MCP2515 FIFO. To reduce the possibility of such overruns, we have highly optimized the CAN and SPI drivers to receive incoming CAN frames significant faster than the default *mcp2515* Linux driver. As a result, even with high bus utilization, baudrates up to 500Kbit/s can be used with (almost) no CAN frame drops. Because of the limitations of the MCP2515 controller and the interrupt latency of Linux, the 1Mbit/s baudrate is not recommend, when CAN frame drops cannot be accepted.

5.2.2.3 *sc16is7xx* : RS232/RS485 driver (optimized for emPC-A/RPI)

By default the emPC-A/RPI is configured to RS232 mode (Jumper J301 is set).

If you want to use RS485 mode, open the case as described in chapter 4 and remove the jumper J301 and check the following configuration:

```
pi@raspberrypi ~$ sudo nano /boot/cmdline.txt
```

When parameter RS485 is set to 2 (RS485=2), then 'automatic mode' is enabled. Starting with hardware revision 1.1, this parameter sets RS232 or RS485 mode based on the jumper J301 setting and no further manual software configuration is required. For older hardware revisions, manually configuration as shown below is required.

Parameter RS485=1 enables the automatic RS485 RTS control mode and automatically controls the RS485 transceiver direction for sending or receiving.

Parameter RS485=0 disabled the RS485 mode and enables normal RS232 operation.

5.2.2.4 *rtc-ds1307* : RTC

MCP7940N RTC is compatible with the *rtc-ds1307* Linux driver (supported from kernel 3.19). The emPC-A/RPI installation installs the RTC service, so that the RTC automatically sets the system time on every boot.

Reading the RTC time:

```
pi@raspberrypi ~$ sudo hwclock -r
```

Writing the system time to the RTC

```
pi@raspberrypi ~$ sudo hwclock --systohc -D --noadjfile --utc-r
```

After replacing the RTC battery, call the following command lines to set the RTC oscillator trim register to value 0 persistently. (Otherwise the RTC will be less precise than it could be)

```
pi@raspberrypi ~$ sudo bash  
pi@raspberrypi ~$ modprobe i2c-dev  
pi@raspberrypi ~$ mknod /dev/i2c-1 c 89 1 2>/dev/null  
pi@raspberrypi ~$ i2cset -f -y 1 0x6f 8 0x00 b
```

5.3 Janz Tec AG Linux Image for emPC-A/RPI

Our pre-installed Janz Tec AG Image is optionally available and contains the combination of the official Raspbian operating system as well as our emPC-A/RPI drivers. File `/root/imageversion.txt` contains the image name and version of the installed Janz Tec image.

You can start the `raspi-config` Raspberry PI configuration tool with “`sudo raspi-config`”, which offers important configuration settings in a convenient menu.



NOTICE

To use the full size of your μ SD-card, it is important that you choose the menu entry “**1 Expand Filesystem**” to expand the Linux partition on the μ SD card to use all the available space for the operating system.

```

Raspberry Pi Software Configuration Tool (raspi-config)
Setup Options
1 Expand Filesystem      Ensures that all of the SD card storage is available to the OS
2 Change User Password  Change password for the default user (pi)
3 Enable Boot to Desktop/Scratch  Choose whether to boot into a desktop environment, Scratch, or the command-line
4 Internationalisation Options  Set up language and regional settings to match your location
5 Enable Camera          Enable this Pi to work with the Raspberry Pi Camera
6 Add to Rastrack        Add this Pi to the online Raspberry Pi Map (Rastrack)
7 Overclock              Configure overclocking for your Pi
8 Advanced Options       Configure advanced settings
9 About raspi-config     Information about this configuration tool

                <Select>                                <Finish>
  
```

<https://www.raspberrypi.org/documentation/configuration/raspi-config.md>

5.4 General Information

- To reduce the risk of losing information stored in files on the μ SD-card, especially when writing to the card during a power failure event, consider the use of an external UPS (uninterruptable power supply). Alternatively, mount the μ SD card as read-only, in combination with a RAM-file system overlay.

5.5 Optional Software

5.5.1 CODESYS Control for Raspberry Pi SL Information



NOTICE

Without further agreements, Janz Tec AG does not provide support for configuring and running CODESYS Control for Raspberry PI SL on the emPC-A/RPI!

CODESYS control runtime system is a soft PLC created by the company 3S-Smart Software Solutions GmbH and is available for Raspberry Pi based devices. Using this IEC 61131-3 programming system in combination with our emPC-A/RPI system is ideal for testing and educational applications, because the control can make use of the additional interfaces the emPC-A/RPI offers (digital IO, CAN, RTC and RS232/RS485).

To download a time limited evaluation version, or buy a full license visit: <http://store.codesys.com/>

**NOTICE**

Always consult the official “CODESYS Control for Raspberry Pi SL” documentation, product description, features and licensing terms.

The following summary of features is only for your information and might not be accurate.

Even with the standard Raspbian operating system without any modifications, the CODESYS Control for Raspberry Pi SL can control all digital IOs on the emPC-A/RPI by using the GPIO device already included in the CODESYS device repository. (Red and green LED also controllable by this GPIO device). The serial console (RS232) is also support by default. *2

When the optionally available emPC-A/RPI driver package is installed, the CODESYS control can also access the CAN port for CAN/CANopen as well as the serial console (RS232) and the RS232/RS485 ports for serial and Modbus RTU communications. The RTC support is especially useful when you do not have a network time server connected, but you want to use date and time in e.g. log files.

CODESYS Features	Standard Raspbian	Raspbian + emPC-A/RPI driver package
CAN/CANopen *1	not supported	supported (using SocketCan)
Modbus RTU Master *2	console (RS232, 2wire)	console (RS232, 2wire) serial port RS232 (4wire) serial port RS485
GPIO device	digital IO (4in, 4out) red and green LEDs	digital IO (4in, 4out) red and green LEDs
Date / Time	requires a time server in the network	<ul style="list-style-type: none"> • RTC / system time • (network time server also possible)

*1: In the current version: 2.1.1.0 of the CODESYS control package, CAN/CANopen is included in the evaluation and paid version, but not mentioned in the product description

*2: May require manual configuration in the CODESYS configuration files

5.5.1.1 Information

If you are interested in CODESYS runtime systems with real-time support, more Ethernet ports, or more fieldbus ports, our embedded PC series emPC and panel PC series emVIEW offer the whole range of industrial PCs in many performance classes.

<https://www.janztec.com/en/embedded-computing.html>

5.5.1.2 License Information

**NOTICE**

We strongly recommend, that you create a backup copy of an activated license!

In case that the contents of the card will become corrupt, the license will be lost! We strongly recommend, that your first step is to create a backup copy of the activated license.

To create a backup copy of the license, copy the folder “backup” with the file „3SLicenseInfo.tar“ from the Raspberry Pi (accessible in CODESYS via double-click on the device under the tab “Files”) to a safe external storage.

6 Programming Information

This chapter gives technical information on how to program, or adapt existing Linux drivers to the emPC-A/RPI hardware and is intended for users familiar with creating own Linux kernels and drivers.

For standard Linux users, optionally our emPC-A/RPI driver package (chapter 5.2) is available and this already includes all required drivers, therefore no driver programming is needed to use the emPC-A/RPI hardware and it's interfaces.

6.1 User LEDs

The two front panel LEDs of the CPU module are controllable by the user. They are controlled by Raspberry GPIOs.

IO	GPIO
GREEN	GPIO5 (Pin 29)
RED	GPIO12 (Pin 32)

LEDs are turned on by setting the GPIO to low level.

Example:

```
root@raspberrypi ~$ echo "12" > /sys/class/gpio/export
root @raspberrypi ~$ echo "out" > /sys/class/gpio/gpio12/direction
root @raspberrypi ~$ echo "1" > /sys/class/gpio/gpio12/value
root @raspberrypi ~$ echo "0" > /sys/class/gpio/gpio12/value
```



NOTICE

When the emPC-A/RPI driver package (chapter 5.2) has been installed, the green LED is configured to show the μ SD-card activity. To control this LED by from the command line, uncomment the entry "dtparam=act_led_gpio=5" from the configuration file /boot/config.txt and remove all entries starting with "bcm2708.disk_*" from the configuration file /boot/cmdline.txt and reboot the system and reboot the system. After this, setting the output value of GPIO 5 controls the green LED.

6.2 I²C Bus

The emPC-A/RPI utilizes the Raspberry I2C bus to connect to several peripheral devices.

Type	Device	Address	Notes
EEPROM	-	0b1010000	Optional
Thermal Sensor	DS75	0b1001000	Optional
RTC	MCP7940	0b1101111	

Table 8: Internal I²C devices

6.3 SPI Bus

The emPC-A/RPI utilizes the Raspberry PI SPI bus to connector CAN and an additional UART.

Type	Device	Chip Select	Interrupt
CAN	MCP2515	CE0#	GPIO25 (Pin22)
UART	SC16IS740	CE1#	GPIO17 (Pin11)

6.4 CAN

The CAN controller is clocked by a 16 MHz crystal.

6.5 Serial Port (RS232/RS485)

The UART is clocked by a 1.8432 MHz oscillator.

When in RS485 mode, the transmitter is controlled by the UART's RTS# signal. A low enables the transmitter.

The receiver is disabled whenever the transmitter is enabled, hence transmitted characters are not received. It is possible to modify the hardware such that the receiver is always enabled, to allow verification of transmitted characters. Ask for details if you need to make such a modification.

6.6 Digital IO

The digital inputs and outputs are controlled by Raspberry PI GPIO signals.

IO	GPIO
DI1	GPIO6 (Pin 31)
DI2	GPIO13 (Pin 33)
DI3	GPIO16 (Pin 36)
DI4	GPIO19 (Pin 35)
DO1	GPIO23 (Pin16)
DO2	GPIO22 (Pin 15)
DO3	GPIO27 (Pin 13)
DO4	GPIO18 (Pin 12)

To turn digital output on (drive VIO to the output), set the GPIO to high state. All other states (low, high-Z or weak pullup) will turn the digital output off.

Digital input GPIOs must be configured as input. A high level will be detected if sufficient voltage is detected on the digital input pin.

Programming the digital inputs and outputs is the same as on a standard Raspberry PI 2 B, so you can get more information on http://elinux.org/RPi_Low-level_peripherals#Shell

6.7 Console RS232

On Raspbian based installations, by default this port is configured as a serial Linux console and you can log into the Raspberry using a standard serial (null modem) cable and a terminal program like "Putty". Use the following settings to log in: Speed (baud rate): 115200, Bits: 8, Parity: None, Stop Bits: 1, Flow Control: None

To permanently disable the serial console and use this port for RS232 (2 wire) connections with the device name `/dev/ttyAMA0`, the tool "raspi-config" allows the configuration in menu, 8 Advanced Options, A7 Serial and "No". (if you need 4-wire RS232, use the RS232/RS485 port, see 3.2.5)

6.8 Overclocking

The thermal design of the emPC-A/RPI does not support CPU overclocking. Changing the CPU parameters (CPU MHz, Voltage) is not recommended as it will permanently set a fuse in your SoC and void your warranty.

7 Appendices

7.1 Technical Data

Only Raspberry features usable on emPC-A/RPI are listed.

Processing Core

CPU BCM2836, up to 900MHz quad-core ARM Cortex-A7 CPU

Memory

Main Memory 1GB LPDDR2 SDRAM

Storage

Micro SDcard 1 x internal socket

External Interfaces

Video 1 x HDMI (Maximum resolution 1920x1080)

Ethernet 1 x 10/100 Mbit/s Ethernet (LAN9512, USB based)

USB 4 x USB2.0

CAN MCP2515 controller

- ISO/DIS 11898-2 (isolated from logic),
- jumper switchable 120 Ohm termination resistor

Serial Port Console (RPI defined)

- RS232 (2 wire)

SC16IS740 controller

- RS232 (4 wire)

- RS485

Digital IO

- 4 x digital OUT (0.5 A max.), 10..32V

- 4 x digital IN

- Isolated from system ground

Indicators and Switches

User LEDs 2 programmable LEDs (red, green)

System

Housing Plastics

RTC MCP7940

Battery CR 2032, for real time clock

Power Requirements

Power Supply DC power, 9 .. 32 V

Inv.-pol. protection Yes

Fuse Internal 2.5 A melting fuse in DC in, GND is unfused

potential separation No, GND is connected to connector shells and PE contact

Inrush Current (max) TBD

Power Dissipation Without external load or expansion cards
TBD

External Load Capabilities

+5V (USB) Max. 500 mA per USB port, max 1000 mA for all USB ports

Environmental Specifications

Ambient Temperature	0 .. +45 °C (vertical orientation / wall and DIN rail mounting)
operating	0 .. +40 °C (horizontal orientation / desk-top mounting)
	at sea level, derated by 1 °C per 300 m above sea level to a maximum of 2000 m.
Temperature storage	-20..+75 °C ²⁾
Humidity	5%..95% r.H., non condensing
Protection Class	IP20

Physical Dimensions

Size (WxHxD)	99.8 x 30.0 x 96.7 mm
Weight	200g

7.2 References

These references direct you to manuals and specifications that you might need to know when you attempt to program the product. Most of the documents can be downloaded from the Internet. Look for the WWW servers of the component/chip manufacturers.

- [1] <https://www.raspberrypi.org/downloads/>
- [2] <https://www.raspberrypi.org/documentation/>

WWW-References

Janz Tec AG

www.janztec.com

7.3 Dimensions

Refer to figure 7 for the housing dimensions.

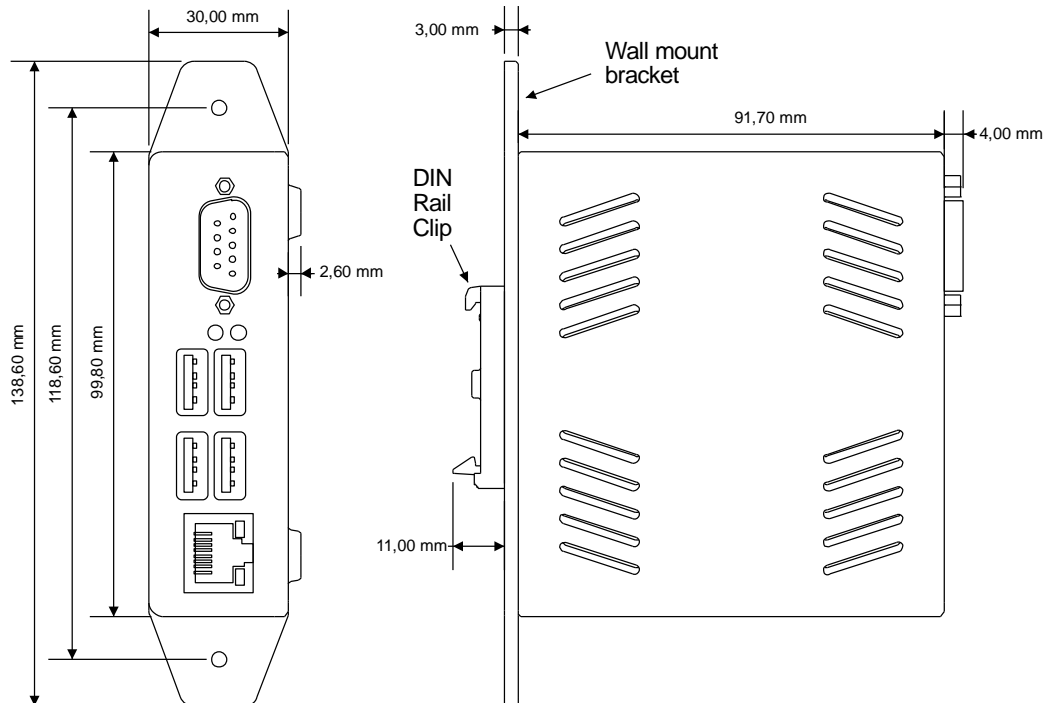


figure 7: housing dimensions

The emPC-A/RPI is delivered with the following mounting options:

- DIN rail clip (not assembled)
- Wall mount bracket (not assembled)
- Self-adhesive rubber bumpers (not assembled)

Note, that you will not normally use both the DIN rail clip and the wall mount bracket at the same time.

7.4 Raspberry GPIO Header

Pin Number	Raspberry	emPC-A/RPI	Direction
1	+3.3V	NC	
2	+5V		
3	GPIO2	I2C-SDA	Bidir
4	+5V		
5	GPIO3	I2C-SCL	Bidir
6	GND		
7	GPIO4	EED_WP	Out
8	GPIO14	Console TxD	Out
9	GND		
10	GPIO15	Console RxD	In
11	GPIO17	INT_UART#	In
12	GPIO18	DO4	Out
13	GPIO27	DO3	Out
14	GND		
15	GPIO22	DO2	Out
16	GPIO23	DO1	Out
17	+3.3V	NC	
18	GPIO24	RS485_MODE_DETECT (0=RS232, 1=RS485)	In
19	GPIO10	SPI_MOSI	Out
20	GND		
21	GPIO9	SPI_MISO	In
22	GPIO25	INT_CAN#	In
23	GPIO11	SPI_CLK	Out
24	GPIO8	SPI_CE_CAN	Out
25	GND		
26	GPIO7	SPI_CE_UART	Out
27		NC	
28		NC	
29	GPIO5	LED green	Out
30	GND		
31	GPIO6	DI1	In
32	GPIO12	LED red	Out
33	GPIO13	DI2	In
34	GND		
35	GPIO19	DI4	In
36	GPIO16	DI3	In
37	GPIO26	NC	
38	GPIO20	NC	
39	GND		
40	GPIO21	NC	

7.5 EMC Cable Installation

In figure 8 we show the EMC shield (PE) connection that we utilized during EMC compatibility tests to pass industrial level immunity.

- Connect PE from the multi connector to the EMC shield (DIN rail)
- Connect cable shields for signals running to the multi connector to the EMC shield (DIN rail).

The terminals/clamps utilized are just examples; other models might work equally good.

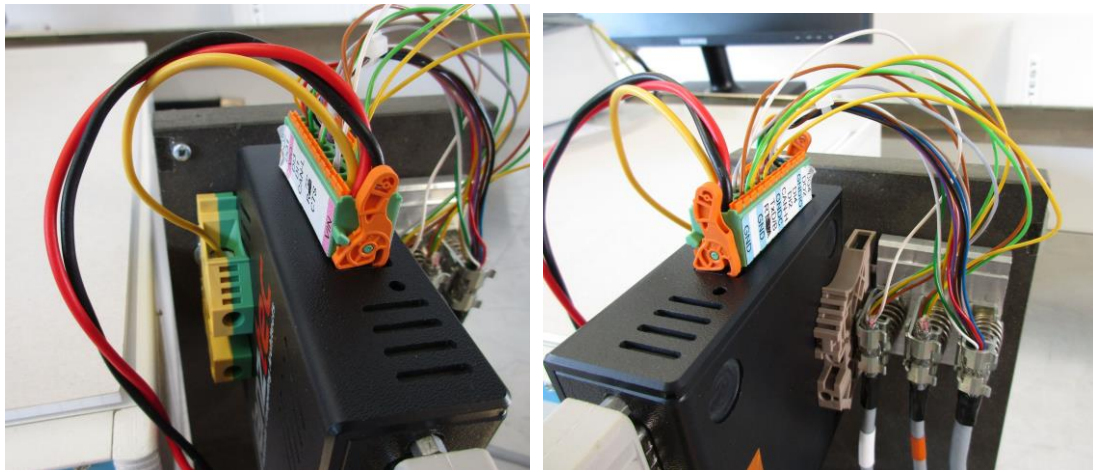


figure 8: EMC Shielding

7.6 Product History

Version	Release Date	Name	Changes
0.9	2015-05-06	as	<ul style="list-style-type: none"> Prototypes
1.0	2015-05-26	ama	<ul style="list-style-type: none"> Prototypes
1.1	2015-08-10	as	<ul style="list-style-type: none"> Added PE connection to multi connector (EMC filter) Some Manufacturing improvements Note: Change of LED makes V1.1 Boards incompatible with V1.0 housings. Added RS485_MODE_DETECT signal to RPI connector Reworked digital inputs circuits might lead to slightly different switching levels and input current
1.2	2015-09-14	As	<ul style="list-style-type: none"> Improved surge handling at DC input.
			<ul style="list-style-type: none">

7.7 Manual History

Version	Release Date	Name	Changes
0.9	2015-05-06	as	<ul style="list-style-type: none"> Initial preliminary release
1.0	2015-05-19	ama	<ul style="list-style-type: none"> Preliminary release
1.1	2015-05-26	ama	<ul style="list-style-type: none"> Operating System and Drivers” chapter added
1.2	2015-08-10	ama	<ul style="list-style-type: none"> Updated chapter 5 for hardware version 1.1
1.3	2015-08-26	ama	<ul style="list-style-type: none"> Updated LED activity deactivation Updated emPC-A/RPI driver package information to v2x3
1.4	2015-09-14	as	<ul style="list-style-type: none"> Updated multi connector pin description with PE location (only V1.1 of the emPC-A/RPI) Added some information regarding EMI compliance Updated chapter 5 Updated block diagram
1.5	2016-02-10	as ama	<ul style="list-style-type: none"> Fixed some spelling errors Updated chapter 5. Changed from driver package to driver installation script Switching level of digital inputs specified