

**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/RPI3+**

embedded PC

(Hardware and Software Manual)

Version 1.2

refers to product revision no.  
1.0

Title: emPC-A/RPI3+ embedded PC  
File: j:\as\entwicklung\projekte\hw\empc\empc-arp\doc>manual\manual\_empc-  
arpi3+.doc  
Pattern: \\ntserv5\janzgrp\as\entwicklung\formulare\entwicklung (ew)\ew\_manual.dot  
Created: Stefan Althöfer, 06.03.2015  
Last Update: Stefan Althöfer, 21.03.2022

© Janz Tec AG 2015-2020

Im Dörener Feld 8  
D-33100 Paderborn, Germany

Tel.: +49-5251-1550-0  
FAX: +49-5251-1550-190  
email: [support@janztec.com](mailto:support@janztec.com)  
Internet: [www.janztec.com](http://www.janztec.com)

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Features .....	5
1.2	Functional Overview .....	6
<b>2</b>	<b>Safety Instructions</b>	<b>7</b>
2.1	Installation and Maintenance .....	7
2.2	Ambient and Environmental Conditions.....	8
<b>3</b>	<b>Intended use</b>	<b>9</b>
3.1	Disposal and Recycling.....	9
3.1.1	Germany .....	9
<b>4</b>	<b>Installation</b>	<b>10</b>
4.1	Mounting .....	10
4.2	Connectors and Operators.....	11
4.2.1	POWER IN.....	11
4.2.2	Graphics connector (HDMI) .....	11
4.2.3	Ethernet Interface .....	12
4.2.4	USB host interfaces (USB1-USB4).....	12
4.2.5	Console (RS232) .....	13
4.2.6	Serial Port (RS232/RS485).....	13
4.2.7	CAN .....	14
4.2.8	Digital IO .....	14
4.2.9	Audio Video.....	15
<b>5</b>	<b>Maintenance</b>	<b>16</b>
5.1	Micro SDcard Replacement .....	17
5.2	Battery Replacement .....	17
<b>6</b>	<b>Operating System and Drivers</b>	<b>18</b>
6.1	Standard Raspbian Operating System .....	18
6.1.1	Installation .....	18
6.2	emPC-A/RPI3+ Driver Installation Script .....	19
6.2.1	Installation Script.....	19
6.2.2	CPU Frequency .....	19
6.2.3	Bluetooth.....	19
6.2.4	Driver Details .....	20
6.2.4.1	spi-bcm2835 : SPI driver .....	20
6.2.4.2	mcp2515 : CAN driver .....	20
6.2.4.3	sc16is7xx : Serial RS232/RS485 driver.....	20
6.2.4.4	rtc-ds1307 : RTC .....	20
6.3	Janz Tec AG Linux Image for emPC-A/RPI3+.....	22
6.4	General Information .....	22
6.5	Optional Software.....	22
6.5.1	CODESYS Control for Raspberry Pi SL Information .....	22
6.5.1.1	Information .....	23
6.5.1.2	License Information .....	23
<b>7</b>	<b>Programming Information</b>	<b>24</b>
7.1	User LEDs.....	24
7.2	I <sup>2</sup> C Bus .....	24
7.3	SPI Bus .....	24
7.4	CAN.....	25

7.5	Serial Port (RS232/RS485).....	25
7.6	Digital IO .....	25
7.7	Console RS232.....	25
7.8	Overclocking .....	26
<b>8</b>	<b>Appendices</b>	<b>27</b>
8.1	Technical Data .....	27
8.2	References .....	30
8.3	Dimensions .....	31
8.4	Raspberry GPIO Header.....	32
8.5	EMC Cable Installation .....	33
8.6	Product History.....	34
8.7	Manual History .....	34

## List of Figures

figure 1: emPC-A/RPI3+ block diagram .....	6
figure 2: emPC-A/RPI3+ mounting options: Desk-top, Wall and DIN Rail .....	10
figure 3: Multi connector for IO (Power Supply Pins highlighted) .....	11
figure 4: RS485 signaling .....	13
figure 5: digital IO detailed schematic .....	14
figure 6: emPC-A/RPI3+ case bottom part removal .....	16
figure 7: emPC-A/RPI3+ micro SDcard location .....	16
figure 8: Removing and replacing coin cell battery .....	17
figure 9: housing dimensions .....	31
figure 10: EMC Shielding .....	33

## List of Tables

table 1: Power Supply connector pins .....	11
table 2: Ethernet connector .....	12
table 3: USB host connector .....	12
table 4: Console (RS232) connector .....	13
table 5: Serial Port connector pins .....	13
table 6: CAN connector pins .....	14
table 7: Digital IO connector pins .....	14
Table 8: Internal I <sup>2</sup> C devices .....	24

## About this Manual






This is the hardware manual for the emPC-A/RPI3+ 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:

	<b>DANGER</b> Indicates that death or severe personal injury <i>will</i> result if proper precautions are not taken.
	<b>WARNING</b> Indicates that death or severe personal injury <i>may</i> result if proper precautions are not taken.
	<b>CAUTION</b> Indicates that <i>minor</i> personal injury can result if proper precautions are not taken.
	<b>NOTICE</b> 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 3 Model B+ powered
  - BCM2837B0 processor, up to 1400MHz quad-core ARM 64bit ARMv8 CPU
  - 1GB LPDDR2 SDRAM
  - HDMI graphics output
  - 10/100/1000 Mbit/s Ethernet
  - 4 x USB 2.0
  - Micro SDcard socket
  - WLAN
  - Bluetooth
- 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)





## 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 (No AC power, no more than 32 V DC/DC or no non-SELV circuit)!

**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.

**CAUTION: Fire Risk**

The digital IO must be powered by a SELV power supply, that complies with the requirements of a limited energy source (LPS) (By using an appropriate supply or an external fuse).

**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.

**NOTICE: System overheating**

The cooling concept of the emPC-A/RPI3+ is not sufficient to allow full loading of the Raspberry PI 3+ Module processor without system overheating which leads to clock reduction. Refer to the technical data for details.

## 3 Intended use

The emPC-A/RPI3+ is designed for computing purposes in industrial environments. It is destined to be used indoor only.

### 3.1 Disposal and Recycling

Janz Tec products are manufactured to satisfy environmental protection requirements where possible. Some of the components used are capable of being recycled. Final disposal of this product after its service life must be accomplished in accordance with applicable country-, state-, or local-laws or -regulations.



This product contains a lithium battery. This should be removed before disposal and be returned separately

#### 3.1.1 Germany

All Janz Tec products are registered as B2B custom at the german EAR. Hence Janz Tec products **must not be** disposed to *public collection points for used electronic equipment*. Refer to §14 of General Terms & Conditions of Janz Tec AG for the details regarding the mutual obligations as to the disposal of Janz Tec products.

The Lithium battery included in the product is registered at “GRS Batterien” and can be disposed to *public collecting points for used batteries*.

## 4 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.

### 4.1 Mounting

The emPC-A/RPI3+ 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.

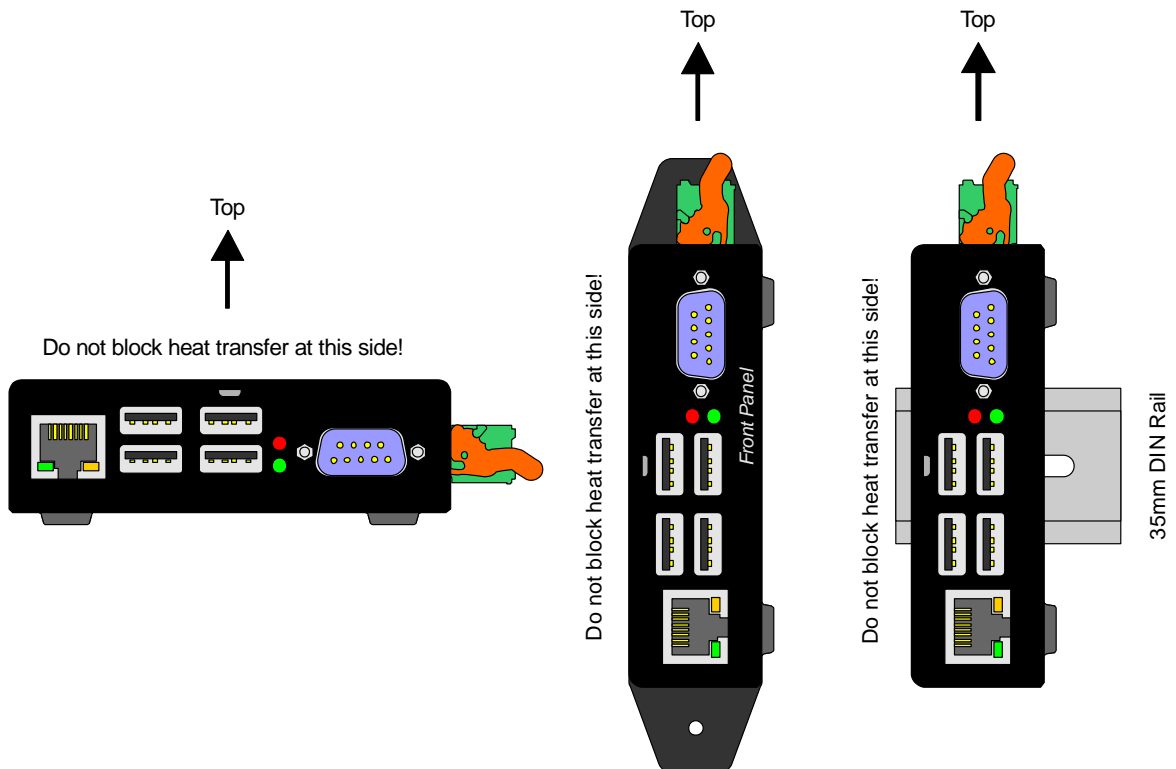


figure 2: emPC-A/RPI3+ mounting options: Desk-top, Wall and DIN Rail

## 4.2 Connectors and Operators

The 24 pin multi connector provides interface to many of the emPC-A/RPI3+ interfaces that are described later.

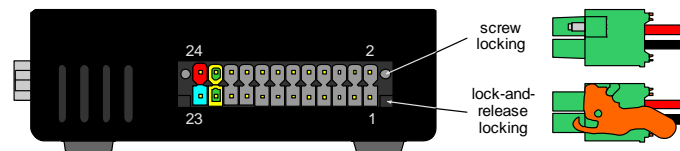


figure 3: Multi connector for IO (Power Supply Pins highlighted)

A suitable mating connector with lock-and-release levers is Phoenix Contact order number 1790580. A Model with screw locking is available with order number 1790399.

The plug uses spring-cage connection and accepts solid or flexible wires from 0.2 mm<sup>2</sup> to 1.5 mm<sup>2</sup>. Push in connection is possible, in case of flexible wires you have to use ferrules for this. If you use ferrules with plastic sleeve, the maximum wire size reduces to 0.75 mm<sup>2</sup>.

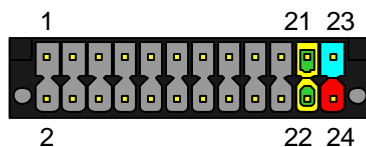


Stripping length for the wires is 10 mm.

If you use ferrules, obey that standard ferrules for 0.75 mm<sup>2</sup> have only 7 mm length.

### 4.2.1 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.

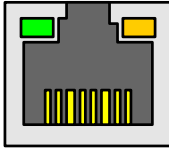
Pins 21 and 22 should be connected to EMI protective earth (PE). These contacts are internally connected to the DC power supply EMI filter and to the D-SUB shell of the console (RS-232). The signal is doubled to prevent false connection, it only requires one connection. In locations where no EMI noise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

### 4.2.2 Graphics connector (HDMI)

Provided by Raspberry PI board.

### 4.2.3 Ethernet Interface

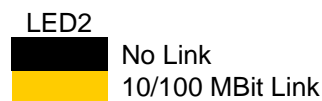
Ethernet interfaces of the Raspberry PI. The Ethernet physics is 10/100BaseT, available through the shielded modular jack. Twisted pair cable can be used to connect to this port.



1	MDI0+	5	MDI2+
2	MDI0-	6	MDI2-
3	MDI1+	7	MDI3+
4	MDI1-	8	MDI3-

table 2: Ethernet connector

The two LEDs indicate Ethernet status as follows:



#### NOTICE

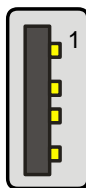
If your Ethernet cable is long (normative limit = 30 m) or leaves a building, then an additional protection against surges is required.

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.

### 4.2.4 USB host interfaces (USB1-USB4)

Two USB interfaces are available at the connector panel.



1	+5V, $I_{MAX} = 500 \text{ mA}$ , 500 mA for all ports
2	USB-
3	USB+
4	GND

table 3: USB host connector



#### NOTICE

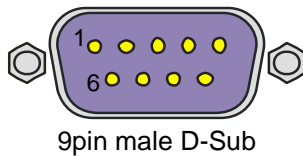
Although each port can deliver supply current of 500 mA, the overall load on USB1 to USB4 interfaces must be limited to 500 mA to prevent power supply from overheating and the +5V internal fuse from triggering.



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.

#### 4.2.5 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.



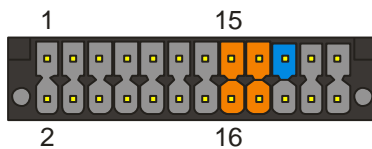
9pin male D-Sub

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

#### 4.2.6 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 , D-	16	RTS , 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 (1=MARK), then  $D+ > D-$  (refer to figure 4).

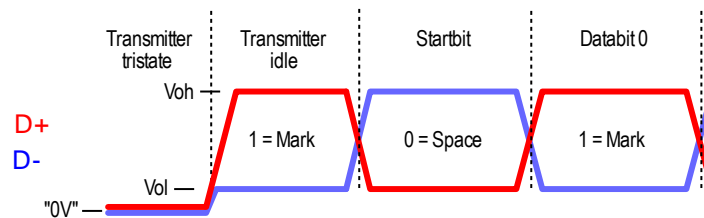


figure 4: RS485 signaling

Note that D+ corresponds to pin "A" of the transceiver chip. However this is signal B according to the EIA-485 or Profibus standards.

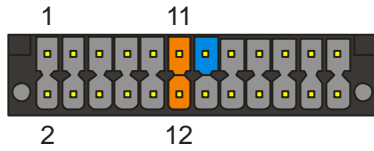
The receiver has the fail safe input feature, which ensures that no data is received when D+ and D- are floating or are shorted (e.g. by the termination resistor).

Refer to section 7 (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.

### 4.2.7 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, as result it is always required to connect 13 GND-CAN to your CAN ground.



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.

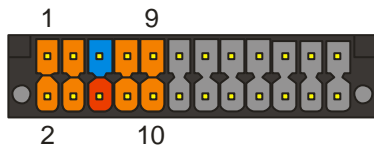


#### CAUTION

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

### 4.2.8 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



#### CAUTION

The digital IO must be powered by a SELV power supply, that complies with the requirements of a limited energy source (LPS) (By using an appropriate supply or an external fuse).

The digital inputs and outputs are isolated from system power supply. Refer to figure 4 for detailed schematic.

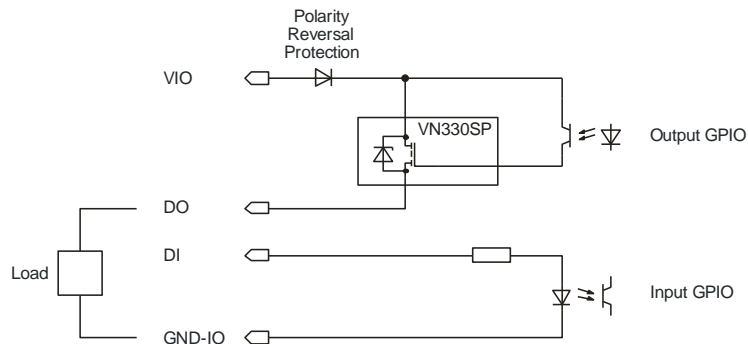


figure 5: digital IO detailed schematic



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 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 VIN. 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.

**4.2.9 Audio Video**

The combined audio and analogue video connector is not accessible on the emPC-A/RPI3+.

## 5 Maintenance



### NOTICE

Always follow common ESD practice when you service the product!

To open the housing, follow these steps (Refer to figure 6):

- 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 $\mu$ SD-card			Remove DIN rail clip and/or wall mount bracket
Change jumper settings			Remove case bottom part
Replace Battery			Complete disassembly

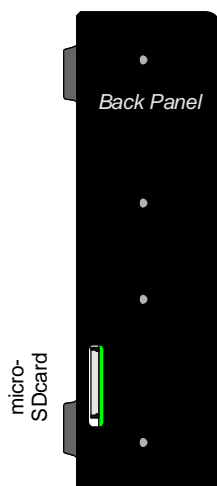


figure 7: emPC-A/RPI3+ micro SDcard location

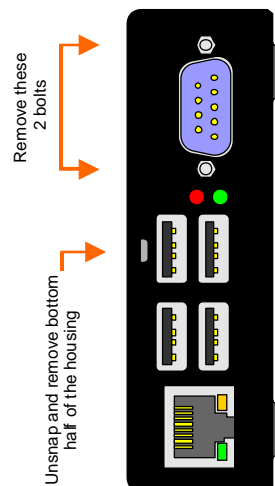


figure 6: emPC-A/RPI3+ case bottom part removal



### NOTICE

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

## 5.1 Micro SDcard Replacement



### NOTICE

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

Refer to figure 6 to identify the location of the Micro SDcard. Remove the DIN rail clip and/or the wall mount bracket if necessary.

## 5.2 Battery Replacement



### CAUTION

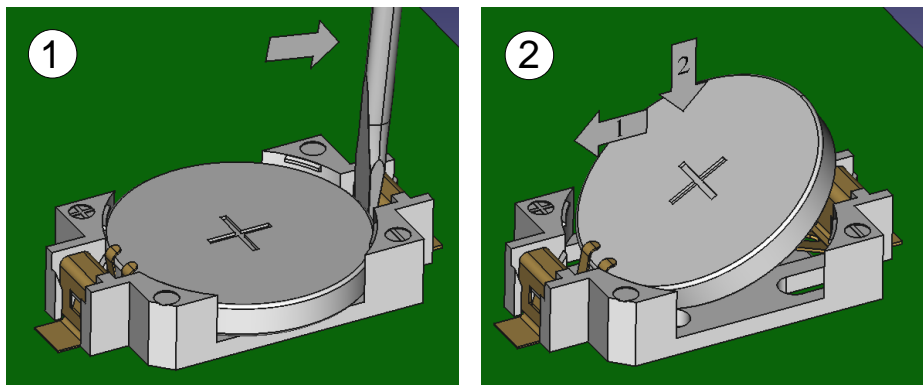
The installed computer board is equipped with a Lithium battery. Danger of explosion if battery is incorrectly replaced. Replace only with battery of the same or equivalent type (3-volt lithium coin cell battery).

- 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 5) and locate the battery on the Janztec board.
4. Remove the battery from the holder (See figure 8)
5. Insert the new battery (See figure 8)
6. Re-assemble the housing
7. See chapter 6.2.4.4 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 (below the hooks) and gently press down on the right side of the battery until the battery snaps into the holder.

figure 8: Removing and replacing coin cell battery

## 6 Operating System and Drivers

You can order the emPC-A/RPI3+ with pre-installed software (Raspbian operating system). Then our default Janz Tec AG Linux image is pre-installed on the  $\mu$ SD-card and the emPC-A/RPI3+ driver package is also pre-installed and fully configured. For more information, refer to 6.2.

Of course, you can order the emPC-A/RPI3+ without software, so you can choose a Raspberry Pi 3 B compatible operating system of your choice and install it manually on a  $\mu$ SD card. (Raspbian recommended)

Refer to 6.1.1 on how to install an image from the official Raspberry Pi website.

### 6.1 Standard Raspbian Operating System

If the additional interfaces of the emPC-A/RPI3+ (RTC, CAN and RS232/RS485) are not required, then the latest official Raspbian operating system is recommended for emPC-A/RPI3+ 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/RPI3+ features, refer to chapter 6.2 and 6.3.

#### 6.1.1 Installation

For more information on how to prepare a  $\mu$ 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 Buster with Desktop image from **2019-07-10** with Kernel 4.19.

(SHA-256: **6a1a5f20329e580d5161a0255b3d4163db6f56c3997e1c3b36bdd51140bd768e**)

#### Download:

Latest image based on RASBIAN Buster:

<http://downloads.raspberrypi.org/raspbian/images/raspbian-2019-07-12/2019-07-10-raspbian-buster.zip>

#### Legacy Download

Legacy images based on RASPBIAN Stretch:

<https://downloads.raspberrypi.org/raspbian/images/raspbian-2019-04-09/2019-04-08-raspbian-stretch.zip>

#### Default log in credentials:

Username: **pi**  
Password: **raspberrypi**

(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/>

## 6.2 emPC-A/RPI3+ Driver Installation Script

**NOTICE**

Always create a backup of the  $\mu$ 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/RPI3+ (RTC, CAN and RS232/RS485), we provide an installation script and installation instructions available at:

<https://github.com/janztec/empc-arpi3-linux-drivers>

(All source code modifications by Janz Tec AG are also included in this repository)

### 6.2.1 Installation Script

**Prerequisites:**

The installation script requires that the emPC-A/RPI3+ has booted successfully from the Raspbian image on the  $\mu$ SD-card prepared in chapter 6.1.1 and that you have created a backup copy of your  $\mu$ SD-card.

**Installation Instructions:**

Follow the instructions on <https://github.com/janztec/empc-arpi3-linux-drivers> to use our installation script to install all required drivers and settings automatically.

### 6.2.2 CPU Frequency

CPU frequency is set to fixed 600MHz in file `/boot/config.txt`. See 8.1 “Environmental Specifications” for more information:

```
root@raspberrypi ~$ cat /boot/config.txt
arm_freq=600
```

### 6.2.3 Bluetooth

Bluetooth is not available after installing this driver package.

The reason is, that the Raspberry PI 3B module has internally two serial ports. Normally the Raspbian image has the following configuration:

**Serial console (uart1):** this uses the core clock and when the CPU core clock changes, the baud rate will change also

**Bluetooth (uart0):** this UART has larger FIFOs and a dedicated clock, so the baud rate is not dependent on the CPU core clock

In favour of the serial console, we have configured that the two serial ports are switched. Serial console then uses `uart0` and Bluetooth uses `uart1`. This is configured in file `/boot/config.txt`.

With this configuration setting, serial console works the same as on the Raspberry PI 2B, but Bluetooth does not work correctly.

```
root@raspberrypi ~$ nano /boot/config.txt
dtoverlay=pi3-miniuart-bt
```

## 6.2.4 Driver Details

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

### 6.2.4.1 *spi-bcm2835* : SPI driver

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.

### 6.2.4.2 *mcp2515* : CAN driver

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@500000 -r -t -b -c
```



#### NOTICE

On high CAN-bus utilization, incoming CAN frames might get dropped because of an overrun in the MCP2515 FIFO using higher CAN baud rates like 500000 or 1000000 baud. To reduce the possibility of such overruns, we patch the default *mcp2515* Linux driver with our optimizations before compilation to achieve better performance.

### 6.2.4.3 *sc16is7xx* : Serial RS232/RS485 driver

By default, the emPC-A/RPI3+ is configured to RS232 mode (Jumper J301 is set).

If you want to use this serial port in RS485 mode, open the case as described in chapter 5, and remove the jumper J301.

The jumper state is checked every system start up (`/etc/rc.local`) and if no jumper is detected, tool "tty-auto-rs485" switches the Linux device `/dev/ttySC0` to RS485 mode. As a result, your application can use Linux device `/dev/ttySC0` identically, regardless of configured RS232 or RS485 mode.

### 6.2.4.4 *rtc-ds1307* : RTC

MCP7940N RTC is compatible with the *rtc-ds1307* Linux driver (supported from kernel 3.19). The emPC-A/RPI3+ installation installs the RTC service, so that the RTC automatically sets the system time on every boot during the `initramfs` execution.

Reading the RTC time:

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

Writing the system time to the RTC

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

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
```

## 6.3 Janz Tec AG Linux Image for emPC-A/RPI3+

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/RPI3+ 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.

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

## 6.4 General Information

- To reduce the risk of losing information stored in files on the  $\mu$ 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  $\mu$ SD card as read-only, in combination with a RAM-file system overlay.

## 6.5 Optional Software

### 6.5.1 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/RPI3+!

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/RPI3+ system is ideal for testing and educational applications, because the control can make use of the additional interfaces the emPC-A/RPI3+ 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/RPI3+ 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/RPI3+ 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.



<b>CODESYS Features</b>	<b>Standard Raspbian</b>	<b>Raspbian + emPC-A/RPI3+ driver package</b>
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"> <li>• RTC / system time</li> <li>• (network time server also possible)</li> </ul>

\*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

#### 6.5.1.1 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>

#### 6.5.1.2 License Information

	<p><b>NOTICE</b></p> <p>We strongly recommend, that you create a backup copy of an activated license!</p>
---	---

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“ and all \*.WibuCmRaU files from the Raspberry Pi (accessible in CODESYS via double-click on the device under the tab "Files") to a safe external storage.

## 7 Programming Information

This chapter gives technical information on how to program, or adapt existing Linux drivers to the emPC-A/RPI3+ hardware and is intended for users familiar with creating own Linux kernels and drivers.

For standard Linux users, optionally our emPC-A/RPI3+ driver package (chapter 6.2) is available and this already includes all required drivers, therefore no driver programming is needed to use the emPC-A/RPI3+ hardware and it's interfaces.

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



#### NOTICE

When the emPC-A/RPI3+ driver package (chapter 6.2) has been installed, the green LED is configured to show the  $\mu$ 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.

### 7.2 I<sup>2</sup>C Bus

The emPC-A/RPI3+ 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 I<sup>2</sup>C devices

### 7.3 SPI Bus

The emPC-A/RPI3+ 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)
------	-----------	------	----------------

## 7.4 CAN

The CAN controller is clocked by a 16 MHz crystal.

## 7.5 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.

## 7.6 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](http://elinux.org/RPi_Low-level_peripherals#Shell)

## 7.7 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)

## 7.8 Overclocking

The thermal design of the emPC-A/RPI3+ 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.

## 8 Appendices

### 8.1 Technical Data

Only Raspberry features usable on emPC-A/RPI3+ are listed.

#### Processing Core

CPU BCM2837B0, up to 1400MHz quad-core 64bit ARMv8 Cortex A53

#### 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 10/100/1000 Mbit/s Ethernet (LAN7515, USB based)

WLAN As defined by Raspberry PI 3 Module

Bluetooth As defined by Raspberry PI 3 Module

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 (UL-VO)

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

Internal 3.0 A eFuse in +5V supply of Pi 3 module

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

Inrush Current (max) TBD

Power Dissipation Without external load or expansion cards

4 x stress-ng matrix-copy 128 @ 600 MHz : 4.2 W (typ)

1 x stress-ng matrix-copy 128 @ 1400 MHz : 4.3 W (typ)

4 x stress-ng matrix-copy 128 @ 1400 MHz : **9.5 W = not supported**

**External Load Capabilities**

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

**Environmental Specifications**

To specify an ambient operating temperature for reliable 24/7 industrial operation, the CPU and GPU load are two very important factors. If all CPU cores are idle all the time, then the maximum specified ambient temperature can be much higher than in a maximum CPU utilization application.

The new BCM2837B0 system-on-chip has much more computation power than its predecessors, but also gets far, far hotter. In high CPU load scenarios, the clock is dynamically increased from 600 MHz to 1400 MHz and then produces so much heat, that even at room temperature the clock would automatically throttle down to 600 MHz after several minutes to prevent overheating.

Because of the thermal limitations of the used standard Raspberry Pi 3 Model B single board computer, the form factor of the casing and the economic limitations in this entry level system, no extensive passive cooling solution can be used. The internal baseboard designed by us providing the power supply and the I/Os and two fieldbuses to the Raspberry Pi board, would allow an ambient operating temperature from -25 .. 60 C° but the Raspberry Pi board is the limiting factor.

The emPC-A/RPI3 uses a passive heat sink on the BCM2837B0 chip and therefore is it not possible to use all of the computation performance the BCM2837B0 provides. The system power consumption ranges from 3.0 W (average when idle) to 9.5 W (maximum under stress). To prevent CPU overheating and throttling, the maximum system power consumption should not be higher than about 4 W. In most real world use cases of this embedded computer, the average power consumption will be lower than this limit, but if constant CPU performance is important for your application, then limiting the maximum CPU frequency to 600 MHz in configuration file `/boot/config.txt` is one possibility to achieve this. Without limiting the maximum CPU frequency, the CPU will dynamically change the frequency depending on the CPU temperature (see commands: `vcgencmd measure_temp` and `vcgencmd measure_clock arm`).

Ambient Temperature operating 0 .. +40 °C (vertical orientation / wall and DIN rail mounting)  
0 .. +35 °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.

the above temperature ratings are defined under the following CPU load:

- No GPU load
- 4 cores loaded at 600 MHz or 1 core loaded at 1400 MHz, where the load per core is comparable to that of the Linux stress-ng utility
- The above numbers are without safety margins to compensate process variations of the Broadcom SoC
- More load will bring the CPU Junction temperature above 80 °C and the CPU will reduce clock frequency towards 600 MHz. The system will not overheat under such conditions.

Temperature storage -20 .. +75 °C<sup>2)</sup>  
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

## 8.2 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](http://www.janztec.com)



## 8.3 Dimensions

Refer to figure 9 for the housing dimensions.

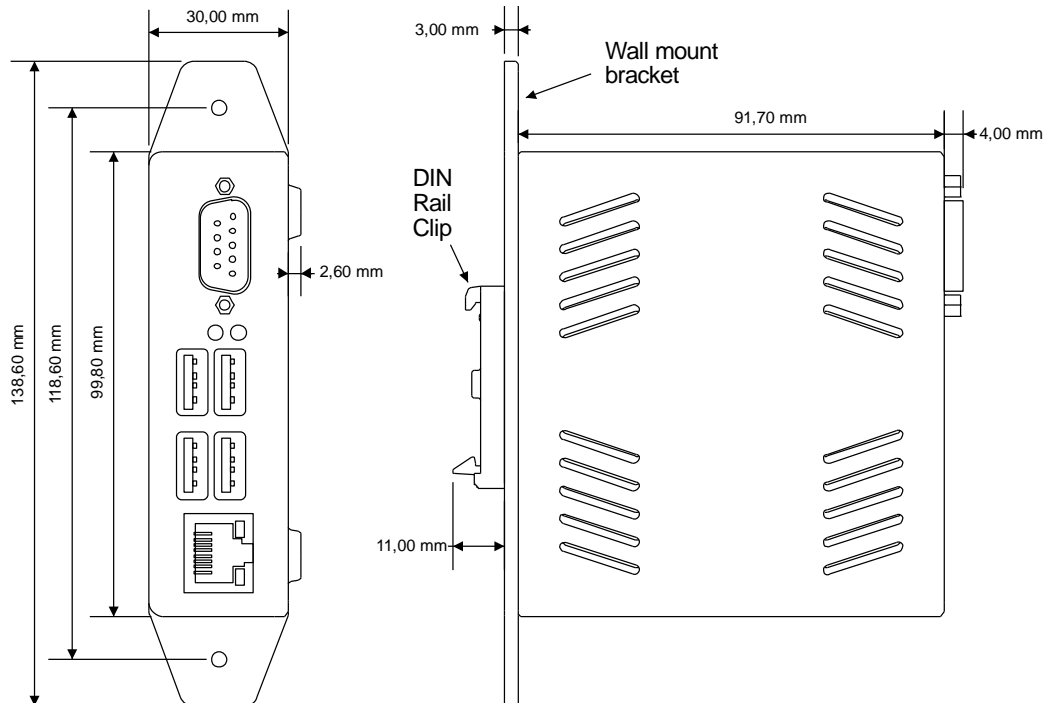


figure 9: housing dimensions

The emPC-A/RPI3+ 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.

## 8.4 Raspberry GPIO Header

Pin Number	Raspberry	emPC-A/RPI3+	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	

## 8.5 EMC Cable Installation

In figure 10 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.

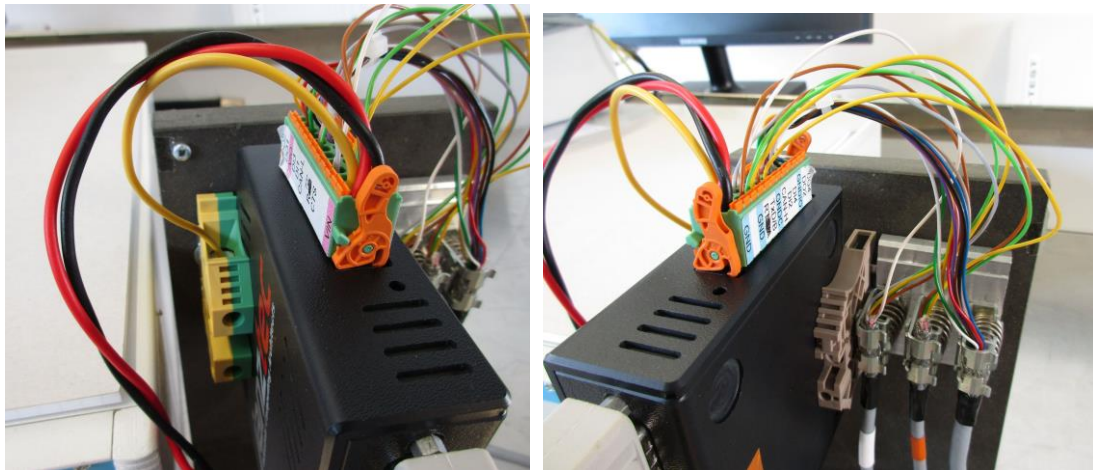


figure 10: EMC Shielding

## 8.6 Product History

Version	Release Date	Name	Changes
1.0	2018-09-21	as	<ul style="list-style-type: none"><li>• Initial release</li></ul>
			<ul style="list-style-type: none"><li>•</li></ul>

## 8.7 Manual History

Version	Release Date	Name	Changes
1.0	2018-08-28		<ul style="list-style-type: none"><li>• Initial release</li></ul>
1.1	2019-09-19	ama	<ul style="list-style-type: none"><li>• Updated chapter "Operating System and Drivers"</li></ul>
1.2	2022-03-21	as	<ul style="list-style-type: none"><li>• Added chapter 3 (Intended Use)A</li><li>• Fixed ethernet speed feature in several places</li></ul>
			<ul style="list-style-type: none"><li>•</li></ul>