TE0715 Test Board
Table of Contents

Overview ................................................................. 5
Key Features ......................................................... 5
Revision History .................................................... 5
Release Notes and Know Issues .................................. 6
Requirements ........................................................ 6
  Software ........................................................... 6
  Hardware .......................................................... 6
Content ............................................................... 7
  Design Sources .................................................. 7
  Additional Sources .............................................. 7
  Prebuilt .......................................................... 7
  Download ........................................................ 8
Design Flow .......................................................... 9
Launch ....................................................................... 11
  Programming ...................................................... 11
    QSPI .............................................................. 11
    SD ................................................................. 11
    JTAG .............................................................. 11
  Usage .................................................................... 11
    Linux .............................................................. 12
    Vivado HW Manager .......................................... 12
System Design - Vivado ............................................. 13
  Block Design ....................................................... 13
    PS Interfaces .................................................... 13
  Constrains ........................................................ 14
    Basic module constrains .................................... 14
    Design specific constrains .................................. 14
Software Design - SDK/HSI ....................................... 15
  Application ........................................................ 15
    zynq_fsbl ....................................................... 15
    zynq_fsbl_flash ............................................. 15
    U-Boot .......................................................... 15
Software Design - PetaLinux ..................................... 16
  Config .................................................................. 16
  U-Boot .............................................................. 16
  Device Tree ...................................................... 17
  Kernel .............................................................. 18
  Rootfs ............................................................... 18
  Applications ...................................................... 18
    startup .......................................................... 18
Additional Software .................................................. 19
  SI5338 ............................................................. 19
Online version of this manual and other related documents can be found at https://wiki.trenz-electronic.de/display/PD/Trenz+Electronic+Documentation
Overview

Zynq Design PS with Linux and simple frequency counter to measure MGT Reference CLK with Vivado HW-Manager.

Key Features

- PetaLinux
- SD
- ETH
- USB
- I2C
- RTC
- FMeter
- SI5338 Initialisation with FSBL (optional)
- Special FSBL for QSPI Programming

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Vivado</th>
<th>Project Built</th>
<th>Authors</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-04-26</td>
<td>2017.4</td>
<td>TE0715-test_board-vivado_2017.4-build_07_20180426171530.zip&lt;br&gt;TE0715-test_board_noprebuild-vivado_2017.4-build_07_20180426171546.zip</td>
<td>John Hartfiel</td>
<td>new assembly variant</td>
</tr>
<tr>
<td>2018-03-27</td>
<td>2017.4</td>
<td>te0715-test_board-vivado_2017.4-build_07_20180327223552.zip&lt;br&gt;te0715-test_board_noprebuild-vivado_2017.4-build_07_20180327223606.zip</td>
<td>John Hartfiel</td>
<td>Board Part Bug fix with UART 1</td>
</tr>
<tr>
<td>2018-01-05</td>
<td>2017.4</td>
<td>te0715-test_board-vivado_2017.4-build_01_20180105195436.zip&lt;br&gt;te0715-test_board_noprebuild-vivado_2017.4-build_01_20180105195452.zip</td>
<td>John Hartfiel</td>
<td>No Design changes&lt;br&gt;Add FSBL for Flash Programming</td>
</tr>
<tr>
<td>2017-11-10</td>
<td>2017.2</td>
<td>te0715-test_board-vivado_2017.2-build_05_20171110134232.zip&lt;br&gt;te0715-test_board_noprebuild-vivado_2017.2-build_05_20171110134247.zip</td>
<td>John Hartfiel</td>
<td>New Web Link on Board Part Files&lt;br&gt;Add optional FSBL Code to reprogram SI5338</td>
</tr>
<tr>
<td>2017-10-19</td>
<td>2017.2</td>
<td>te0715-test_board-vivado_2017.2-build_04_20171019141808.zip&lt;br&gt;te0715-test_board_noprebuild-vivado_2017.2-build_04_20171019141825.zip</td>
<td>John Hartfiel</td>
<td>changed Flash typ on TE0715_board_files.csv&lt;br&gt;(older one is not supported on Vivado 2017.2)</td>
</tr>
<tr>
<td>2017-09-22</td>
<td>2017.2</td>
<td>te0715-test_board-vivado_2017.2-build_02_20170927143412.zip&lt;br&gt;te0715-test_board_noprebuild-vivado_2017.2-build_02_20170927143427.zip</td>
<td>John Hartfiel</td>
<td>initial release</td>
</tr>
</tbody>
</table>
Release Notes and Know Issues

<table>
<thead>
<tr>
<th>Issues</th>
<th>Description</th>
<th>Workaround</th>
<th>To be fixed version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timing problems with Frequency counter</td>
<td>can be ignored</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

Requirements

Software

<table>
<thead>
<tr>
<th>Software</th>
<th>Version</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vivado</td>
<td>2017.4</td>
<td>needed</td>
</tr>
<tr>
<td>SDK</td>
<td>2017.4</td>
<td>needed</td>
</tr>
<tr>
<td>PetaLinux</td>
<td>2017.4</td>
<td>needed</td>
</tr>
</tbody>
</table>

Hardware

Basic description of TE Board Part Files is available on [TE Board Part Files](http://www.trenz-electronic.de).

Complete List is available on <design name>/board_files/*_board_files.csv

Design supports following modules:

<table>
<thead>
<tr>
<th>Module Model</th>
<th>Board Short Name</th>
<th>PCB Revision Support</th>
<th>DDR</th>
<th>QSPI Flash</th>
<th>Others</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>TE0715-03-15-1C</td>
<td>03_15_1c</td>
<td>REV01,02,03</td>
<td>1GB</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-03-15-1I</td>
<td>03_15_1i</td>
<td>REV01,02,03</td>
<td>1GB</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-03-15-2I</td>
<td>03_15_2i</td>
<td>REV01,02,03</td>
<td>1GB</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-03-30-1C</td>
<td>03_30_1c</td>
<td>REV01,02,03</td>
<td>1GB</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-03-30-1I</td>
<td>03_30_1i</td>
<td>REV01,02,03</td>
<td>1GB</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-03-30-3E</td>
<td>03_30_3e</td>
<td>REV01,02,03</td>
<td>1GB</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-15-1C</td>
<td>04_15_1c</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-15-1I</td>
<td>04_15_1i</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-15-2I</td>
<td>04_15_2i</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-30-1C</td>
<td>04_30_1c</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-30-1I</td>
<td>04_30_1i</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-30-3E</td>
<td>04_30_3e</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-12s-1C</td>
<td>12s</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TE0715-04-30-11A</td>
<td>04_30_1i</td>
<td>REV04</td>
<td>1GB_L</td>
<td>32</td>
<td>Micron instead of Spansion Flash</td>
<td></td>
</tr>
</tbody>
</table>
Design supports following carriers:

<table>
<thead>
<tr>
<th>Carrier Model</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>TE0701</td>
<td></td>
</tr>
<tr>
<td>TE0703</td>
<td>used as reference carrier</td>
</tr>
<tr>
<td>TE0705</td>
<td></td>
</tr>
<tr>
<td>TE0706</td>
<td></td>
</tr>
<tr>
<td>TEBA0841</td>
<td></td>
</tr>
</tbody>
</table>

Additional HW Requirements:

<table>
<thead>
<tr>
<th>Additional Hardware</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB Cable for JTAG/UART</td>
<td>Check Carrier Board and Programmer for correct typ</td>
</tr>
<tr>
<td>XMOD Programmer</td>
<td>Carrier Board dependent, only if carrier has no own FTDI</td>
</tr>
</tbody>
</table>

Content

For general structure and of the reference design, see Project Delivery

Design Sources

<table>
<thead>
<tr>
<th>Type</th>
<th>Location</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vivado</td>
<td>&lt;design name&gt;/block_design</td>
<td>Vivado Project will be generated by TE Scripts</td>
</tr>
<tr>
<td></td>
<td>&lt;design name&gt;/constraints</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;design name&gt;/ip_lib</td>
<td></td>
</tr>
<tr>
<td>SDK/HSI</td>
<td>&lt;design name&gt;/sw_lib</td>
<td>Additional Software Template for SDK/HSI and apps_list.csv with settings for HSI</td>
</tr>
<tr>
<td>PetaLinux</td>
<td>&lt;design name&gt;/os/petalinux</td>
<td>PetaLinux template with current configuration</td>
</tr>
</tbody>
</table>

Additional Sources

<table>
<thead>
<tr>
<th>Type</th>
<th>Location</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SI5338 Project</td>
<td>&lt;design_name&gt;/misc/si5338</td>
<td>---</td>
</tr>
</tbody>
</table>

Prebuilt

Only on ZIP file with Prebuilt content.

<table>
<thead>
<tr>
<th>File</th>
<th>File-Extension</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIF-File</td>
<td>*.bif</td>
<td>File with description to generate Bin-File</td>
</tr>
<tr>
<td>BIN-File</td>
<td>*.bin</td>
<td>Flash Configuration File with Boot-Image (Zynq-FPGAs)</td>
</tr>
<tr>
<td>BIT-File</td>
<td>*.bit</td>
<td>FPGA (PL Part) Configuration File</td>
</tr>
<tr>
<td>File</td>
<td>File-Extension</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DebugProbes-File</td>
<td>*.ltx</td>
<td>Definition File for Vivado/Vivado Labtools Debugging Interface</td>
</tr>
<tr>
<td>Diverse Reports</td>
<td>---</td>
<td>Report files in different formats</td>
</tr>
<tr>
<td>Hardware-Platform-Specification-Files</td>
<td>*.hdf</td>
<td>Exported Vivado Hardware Specification for SDK/HSI and PetaLinux</td>
</tr>
<tr>
<td>LabTools Project-File</td>
<td>*.lpr</td>
<td>Vivado Labtools Project File</td>
</tr>
<tr>
<td>OS-Image</td>
<td>*.ub</td>
<td>Image with Linux Kernel (On Petalinux optional with Devicetree and RAM-Disk)</td>
</tr>
<tr>
<td>Software-Application-File</td>
<td>*.elf</td>
<td>Software Application for Zynq or MicroBlaze Processor Systems</td>
</tr>
</tbody>
</table>

**Download**

Reference Design is only usable with the specified Vivado/SDK/PetaLinux/SDx version. Do never use different Versions of Xilinx Software for the same Project.

Reference Design is available on:

- TE0715 "Test Board" Reference Design
Design Flow

Reference Design is available with and without prebuilt files. It's recommended to use TE prebuilt files for first lunch.

Trenz Electronic provides a tcl based built environment based on Xilinx Design Flow.

See also:

- Vivado/SDK/SDSoC#XilinxSoftware-BasicUserGuides
- Vivado Projects
- Project Delivery.

The Trenz Electronic FPGA Reference Designs are TCL-script based project. Command files for execution will be generated with "_create_win_setup.cmd" on Windows OS and "_create_linux_setup.sh" on Linux OS.

TE Scripts are only needed to generate the vivado project, all other additional steps are optional and can also executed by Xilinx Vivado/SDK GUI. For currently Scripts limitations on Win and Linux OS see: Project Delivery Currently limitations of functionality

1. Run _create_win_setup.cmd/_create_linux_setup.sh and follow instructions on shell:

2. Press 0 and enter for minimum setup

3. (optional Win OS) Generate Virtual Drive or use short directory for the reference design (for example x:\<design name>)

4. Create Project
   a. Select correct device and Xilinx install path on "design_basic_settings.cmd" and create Vivado project with "vivado_create_project_gui_mode.cmd"
   Note: Select correct one, see TE Board Part Files
5. Create HDF and export to prebuilt folder
   a. Run on Vivado TCL: `TE::hw_build_design -export_prebuilt`
      Note: Script generate design and export files into \prebuilt\hardware\<short dir>. Use GUI is the same, except file export to prebuilt folder

6. Create Linux (uboot.elf and image.ub) with exported HDF
   a. HDF is exported to "prebuilt\hardware\<short name>"
      Note: HW Export from Vivado GUI create another path as default workspace.
   b. Create Linux images on VM, see PetaLinux KICKstart
      i. Use TE Template from \os\petalinux
      Note: run init_config.sh before you start petalinux config. This will set correct temporary path variable.

7. Add Linux files (uboot.elf and image.ub) to prebuilt folder
   a. "prebuilt\os\petalinux\default" or "prebuilt\os\petalinux\<short name>"
      Notes: Scripts select "prebuilt\os\petalinux\<short name>", if exist, otherwise "prebuilt\os\petalinux\default"

8. Generate Programming Files with HSI/SDK
   a. Run on Vivado TCL: `TE::sw_run_hsi`
      Note: Scripts generate applications and bootable files, which are defined in "sw_lib\apps_list.csv"
   b. (alternative) Start SDK with Vivado GUI or start with TE Scripts on Vivado TCL: `TE::sw_run_sdk`
      Note: See SDK Projects
Launch

⚠️ Check Module and Carrier TRMs for proper HW configuration before you try any design.

Reference Design is also available with prebuilt files. It's recommended to use TE prebuilt files for first lunch.

TE0715-0x-30-xx only: HP IO Banks max power supply voltage is 1.8V.

Programming

Xilinx documentation for programming and debugging: Vivado/SDK/SDSoC-Xilinx Software Programming and Debugging

QSPI

Optional for Boot.bin on QSPI Flash and image.ub on SD.

1. Connect JTAG and power on carrier with module
2. Open Vivado Project with "vivado_open_existing_project_guimode.cmd" or if not created, create with "vivado_create_project_guimode.cmd"
3. Type on Vivado TCL Console: TE::pr_program_flash_binfile -swapp u-boot
   Note: To program with SDK/Vivado GUI, use special FSBL (zynq_fsbl_flash) on setup
4. Copy image.ub on SD-Card
5. Insert SD-Card

SD

1. Copy image.ub and Boot.bin on SD-Card.
   • For correct prebuilt file location, see <design_name>/prebuilt/readme_file_location.txt
2. Set Boot Mode to SD-Boot.
   • Depends on Carrier, see carrier TRM.
3. Insert SD-Card in SD-Slot.

JTAG

Not used on this Example.

Usage

1. Prepare HW like described on section Programming
2. Connect UART USB (most cases same as JTAG)
3. Select SD Card as Boot Mode (or QSPI - depending on step 1)
   Note: See TRM of the Carrier, which is used.
4. Power On PCB  
   Note: 1. Zynq Boot ROM loads FSBL from SD into OCM, 2. FSBL loads U-boot from SD into DDR, 3. U-boot load Linux from SD into DDR

**Linux**

1. Open Serial Console (e.g. putty)  
   a. Speed: 115200  
   b. COM Port: Win OS, see device manager, Linux OS see dmesg |grep tty (UART is *USB1)

2. Linux Console:  
   Note: Wait until Linux boot finished For Linux Login use:  
   a. User Name: root  
   b. Password: root

3. You can use Linux shell now.  
   a. I2C 1 Bus type: i2cdetect -y -r 1  
   b. RTC check: dmesg | grep rtc  
   c. ETH0 works with udhcpc

**Vivado HW Manager**

MGT Reference CLK Counter:  

1. Open Vivado HW-Manager and add VIO signal to dashboard (*.ltx located on prebuilt folder).  
   a. Set radix from VIO signals to unsigned integer.  
   Note: Frequency Counter is inaccurate and displayed unit is Hz

MGT CLK is configured to 125MHz by default, FCLK is not configured by default (optional possible see FSBL description).
System Design - Vivado

Block Design

---

PS Interfaces

Activated interfaces:

<table>
<thead>
<tr>
<th>Type</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDR</td>
<td>--</td>
</tr>
<tr>
<td>QSPI</td>
<td>MIO</td>
</tr>
<tr>
<td>I2C0</td>
<td>EMIO-NC</td>
</tr>
<tr>
<td>I2C1</td>
<td>MIO</td>
</tr>
<tr>
<td>UART0</td>
<td>MIO</td>
</tr>
<tr>
<td>GPIO</td>
<td>MIO</td>
</tr>
<tr>
<td>SD0</td>
<td>MIO</td>
</tr>
</tbody>
</table>
Constrains

Basic module constrains

```plaintext
_i_bitgen_common.xdc

set_property BITSTREAM.GENERAL.COMPRESS TRUE [current_design]
set_property CONFIG_VOLTAGE 3.3 [current_design]
set_property CFGBVS VCCO [current_design]

set_property BITSTREAM.CONFIG.USR_ACCESS TIMESTAMP [current_design]
```

```plaintext
_i_unused_io.xdc

set_property BITSTREAM.CONFIG.UNUSEDPIN PULLNONE [current_design]
```

Design specific constrain

```plaintext
_i_io.xdc

set_property PACKAGE_PIN K2 [get_ports {fclk[0]}]
set_property IOSTANDARD LVCMOS18 [get_ports {fclk[0]}]
set_property CLOCK_DEDICATED_ROUTE FALSE [get_nets fclk_IBUF[0]]
```
Software Design - SDK/HSI

For SDK project creation, follow instructions from:

- SDK Projects

Application

zynq_fsbl

TE modified 2017.4 FSBL

Changes:

- Si5338 Configuration see fsbl_hooks.c  
  add define RECONFIGURE_SI5338 to enable PLL programming with given register_map.h setup
- Add register_map.h, si5338.c, si5338.h

zynq_fsbl_flash

TE modified 2017.4 FSBL

Changes:

- Set FSBL Boot Mode to JTAG
- Disable Memory initialisation

U-Boot

U-Boot.elf is generated with PetaLinux. SDK/HSI is used to generate Boot.bin.
Software Design - PetaLinux

For PetaLinux installation and project creation, follow instructions from:

- PetaLinux KICKstart

Config
No changes.

U-Boot
No changes.
Device Tree

```
/include/ "system-conf.dtsi"
/
};

/* default */

/* ETH PHY */
&gem0 {
    status = "okay";
    ethernet_phy0: ethernet-phy@0 {
        compatible = "marvell,88e1510";
        device_type = "ethernet-phy";
        reg = <0>;
    }
};

/* USB PHY */
&usb0 {
    usb_phy0: usb_phy@0 {
        compatible = "ulpi-phy";
        //compatible = "usb-nop-xceiv";
        #phy-cells = <0>;
        reg = <0xe0002000 0x1000>;
        view-port = <0x0170>;
        drv-vbus;
    }
};

&usb0 {
    dr_mode = "host";
    //dr_mode = "peripheral";
    usb-phy = <&usb_phy0>;
}

/* I2C */
// I2c PLL: 0x70, i2c eeprom: 0x50
&i2c1 {
    rtc@6F { // Real Time Clock
        compatible = "isl12022";
        reg = <0x6F>;
    }
};
```
Kernel

Activate:

- RTC_DRV_ISR12022

Rootfs

Activate:

- i2c-tools

Applications

startup

Script App to load init.sh from SD Card if available.

See: `os/petlinux/project-spec/meta-user/recipes-apps/startup/files`
Additional Software

SI5338

Download ClockBuilder Desktop for SI5338

1. Install and start ClockBuilder
2. Select SI5338
3. Options  Open register map file
   Note: File location <design name>/misc/Si5338/RegisterMap.txt
4. Modify settings
5. Options  save C code header files
6. Replace Header files from FSBL template with generated file
# Appx. A: Change History and Legal Notices

## Document Change History

To get content of older revision got to "Change History" of this page and select older document revision number.

<table>
<thead>
<tr>
<th>Date</th>
<th>Document Revision</th>
<th>Authors</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-10-01</td>
<td>v.31</td>
<td>John Hartfiel</td>
<td>- New assembly variant</td>
</tr>
<tr>
<td>27.03.2018</td>
<td>v.29</td>
<td>John Hartfiel</td>
<td>- Bugfix Board Part Files</td>
</tr>
<tr>
<td>2018-02-13</td>
<td>v.28</td>
<td>John Hartfiel</td>
<td>- Release 2017.4</td>
</tr>
<tr>
<td>2017-11-10</td>
<td>v.22</td>
<td>John Hartfiel</td>
<td>- Design Update with new options</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Add Si5338 section</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Update FSBL section</td>
</tr>
<tr>
<td>2017-10-19</td>
<td>v.21</td>
<td>John Hartfiel</td>
<td>- Download Update</td>
</tr>
<tr>
<td>2017-10-19</td>
<td>v.20</td>
<td>John Hartfiel</td>
<td>- Document style update</td>
</tr>
<tr>
<td>2017-10-06</td>
<td>v.18</td>
<td>John Hartfiel</td>
<td>- Text correction</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Update Launch section</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Supported PCBs</td>
</tr>
<tr>
<td>2017-10-02</td>
<td>v.14</td>
<td>John Hartfiel</td>
<td>- Document update on Prebuilt section</td>
</tr>
<tr>
<td>2017-09-28</td>
<td>v.13</td>
<td>John Hartfiel</td>
<td>- Release 2017.2</td>
</tr>
<tr>
<td>2017-09-11</td>
<td>v.1</td>
<td>John Hartfiel</td>
<td>- Initial release</td>
</tr>
<tr>
<td>All</td>
<td></td>
<td>John Hartfiel</td>
<td></td>
</tr>
</tbody>
</table>

## Legal Notices

### Data privacy

Document Warranty

The material contained in this document is provided “as is” and is subject to being changed at any time without notice. Trenz Electronic does not warrant the accuracy and completeness of the materials in this document. Further, to the maximum extent permitted by applicable law, Trenz Electronic disclaims all warranties, either express or implied, with regard to this document and any information contained herein, including but not limited to the implied warranties of merchantability, fitness for a particular purpose or non infringement of intellectual property. Trenz Electronic shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein.

Limitation of Liability

In no event will Trenz Electronic, its suppliers, or other third parties mentioned in this document be liable for any damages whatsoever (including, without limitation, those resulting from lost profits, lost data or business interruption) arising out of the use, inability to use, or the results of use of this document, any documents linked to this document, or the materials or information contained at any or all such documents. If your use of the materials or information from this document results in the need for servicing, repair or correction of equipment or data, you assume all costs thereof.

Copyright Notice

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Trenz Electronic.

Technology Licenses

The hardware / firmware / software described in this document are furnished under a license and may be used /modified / copied only in accordance with the terms of such license.

Environmental Protection

To confront directly with the responsibility toward the environment, the global community and eventually also oneself. Such a resolution should be integral part not only of everybody's life. Also enterprises shall be conscious of their social responsibility and contribute to the preservation of our common living space. That is why Trenz Electronic invests in the protection of our Environment.
REACH, RoHS and WEEE

REACH

Trenz Electronic is a manufacturer and a distributor of electronic products. It is therefore a so called downstream user in the sense of REACH. The products we supply to you are solely non-chemical products (goods). Moreover and under normal and reasonably foreseeable circumstances of application, the goods supplied to you shall not release any substance. For that, Trenz Electronic is obliged to neither register nor to provide safety data sheet. According to present knowledge and to best of our knowledge, no SVHC (Substances of Very High Concern) on the Candidate List are contained in our products. Furthermore, we will immediately and unsolicited inform our customers in compliance with REACH - Article 33 if any substance present in our goods (above a concentration of 0,1 % weight by weight) will be classified as SVHC by the European Chemicals Agency (ECHA).

RoHS

Trenz Electronic GmbH herewith declares that all its products are developed, manufactured and distributed RoHS compliant.

WEEE


Users of electrical and electronic equipment in private households are required not to dispose of waste electrical and electronic equipment as unsorted municipal waste and to collect such waste electrical and electronic equipment separately. By the 13 August 2005, Member States shall have ensured that systems are set up allowing final holders and distributors to return waste electrical and electronic equipment at least free of charge. Member States shall ensure the availability and accessibility of the necessary collection facilities. Separate collection is the precondition to ensure specific treatment and recycling of waste electrical and electronic equipment and is necessary to achieve the chosen level of protection of human health and the environment in the European Union. Consumers have to actively contribute to the success of such collection and the return of waste electrical and electronic equipment. Presence of hazardous substances in electrical and electronic equipment results in potential effects on the environment and human health. The symbol consisting of the crossed-out wheeled bin indicates separate collection for waste electrical and electronic equipment.

Trenz Electronic is registered under WEEE-Reg.-Nr. DE97922676.

2018-09-18