TE0722 Test Board

Revision v.9

Exported on 2024-03-25

Online version of this document:

https://wiki.trenz-electronic.de/display/PD/TE0722+Test+Board

1 Table of Contents

2 Table of Figures

3 Table of Tables

4 Overview

Zynq PS Design with DDR Less FSBL Example.

Refer to http://trenz.org/te0722-info for the current online version of this manual and other available documentation.

4.1 Key Features

- Vivado 2021.2
- UART
- I2C
- SD
- Modified FSBL for DDR Less Zynq
- Modified FSBL for DDR Less Zynq + small app with LED+Sensor and SD Card access
- Special FSBL for QSPI programming

4.2 Revision History

Date	Vivado	Project Built	Authors	Description
2023-02-13	2021.2	TE0722-test_board_noprebuilt-vivado_2021.2-build_20_20230214143311.zip TE0722-test_board-vivado_2021.2-build_20_20230214143311.zip	Waldemar Hanemann	• 2021.2 update
2020-04-16	2019.2	TE0722-test_board_noprebuilt-vivado_2019.2-build_10_20200416064916.zip TE0722-test_board-vivado_2019.2-build_10_20200416064756.zip	John Hartfiel	• 2019.2 update
2019-05-22	2018.3	TE0722-test_board-vivado_2018.3-build_05_20190522113216.zip	John Hartfiel	• split FSBL into 2 templates, one with and one without

Date	Vivado	Project Built	Authors	Description
		TE0722-test_board_noprebuilt-vivado_2018.3-build_05_20190522113228.zip		Sensor+LED access example app
2019-05-14	2018.3	TE0722-test_board-vivado_2018.3-build_05_20190510163659.zip TE0722-test_board_noprebuilt-vivado_2018.3-build_05_20190510163900.zip	John Hartfiel	• TE Script update • rework of the FSBLs • DDR LESS, Device ID, Sensor+LED access • VIO for RGB access
2018-08-14	2018.2	TE0722-test_board-vivado_2018.2-build_02_20180815123557.zip TE0722-test_board_noprebuilt-vivado_2018.2-build_02_20180815123610.zip	John Hartfiel	• initial release

Table 1: Design Revision History

4.3 Release Notes and Know Issues

Issues	Description	Workaround	To be fixed version

QSPI Flash Programming failed with 19.2	Depending on Flash content Flash programming failed with provided fsbl_flash (Xilinx AR# 70548[1]) 2019.2 version	• Option1:
		  • In case Flash is empty, use fsbl_flash on programming GUI
		  • In case Flash is programmed use normal	

1 https://www.xilinx.com/support/answers/70548.html

Issues	Description	Workaround	To be fixed version
		fsbl on programming GUI • Option2: use in both case fsbl_flash on programming GUI and Vivado LabTools 2018.3 • see also AR#00002[2] and TE0722-Recovery[3]	

Table 2: Known Issues

4.4 Requirements

4.4.1 Software

Software	Version	Note
Vitis	2021.2	needed, Vivado is included into Vitis installation

Table 3: Software

4.4.2 Hardware

Basic description of TE Board Part Files is available on TE Board Part Files.[4]

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

Design supports following modules:

Module Model	Board Part Short Name	PCB Revision Support	DDR	QSPI Flash	EMMC	Others	Notes
TE0722-01	10	REV01	0GB	16MB	NA	NA	NA

2 https://wiki.trenz-electronic.de/pages/viewpage.action?pageId=105689937
3 https://wiki.trenz-electronic.de/display/PD/TE0722-Recovery
4 https://wiki.trenz-electronic.de/display/PD/TE+Board+Part+Files

Module Model	Board Part Short Name	PCB Revision Support	DDR	QSPI Flash	EMMC	Others	Notes
TE0722-02	10	REV02	0GB	16MB	NA	NA	NA
TE0722-02I *	10_i	REV02	0GB	16MB	NA	NA	NA
TE0722-02IC7	10_i_c7	REV02	0GB	16MB	NA	"without SD"	NA
TE0722-02-07S-1C	7s	REV02	0GB	16MB	NA	NA	NA

Table 4: *used as reference

Additional HW Requirements:

Additional Hardware	Notes
TE0790 or other JTAG programmer	for JTAG, UART
external 3.3V power supply	

4.5 Content

For general structure and of the reference design, see Project Delivery - AMD devices[5]

4.5.1 Design Sources

Type	Location	Notes
Vivado	<project folder>\block_design <project folder>\constraints <project folder>\ip_lib <project folder>\board_files	Vivado Project will be generated by TE Scripts

5 https://wiki.trenz-electronic.de/display/PD/Project+Delivery+-+AMD+devices

Type	Location	Notes
Vitis	<project folder>\sw_lib	Additional Software Template for Vitis and apps_list.csv with settings automatically for Vitis app generation

4.5.2 Additional Sources

Type	Location	Notes
--	--	--

4.5.3 Prebuilt

File	File-Extension	Description
BIF-File	*.bif	File with description to generate Bin-File
BIN-File	*.bin	Flash Configuration File with Boot-Image (Zynq-FPGAs)
BIT-File	*.bit	FPGA (PL Part) Configuration File
DebugProbes-File	*.ltx	Definition File for Vivado/Vivado Labtools Debugging Interface
Diverse Reports	---	Report files in different formats
Hardware-Platform-Specification-Files	*.xsa	Exported Vivado Hardware Specification for Vitis and PetaLinux
LabTools Project-File	*.lpr	Vivado Labtools Project File
Software-Application-File	*.elf	Software Application for Zynq or MicroBlaze Processor Systems

4.5.4 Download

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

- TE0722 "Test Board" Reference Design[6]

6 https://shop.trenz-electronic.de/Download/?path=Trenz_Electronic/Modules_and_Module_Carriers/special/TE0722/Reference_Design/2021.2/test_board

5 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:

- AMD Development Tools[7]
- Vivado Projects - TE Reference Design[8]
- Project Delivery.[9]

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/Vitis GUI. For currently Scripts limitations on Win and Linux OS see: Project Delivery Currently limitations of functionality[10]

> ⚠ **Caution!** Win OS has a 260 character limit for path lengths which can affect the Vivado tools. To avoid this issue, use Virtual Drive or the shortest possible names and directory locations for the reference design (for example "x:\<project folder>")

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

_create_win_setup.cmd/_create_linux_setup.sh

```
------------------------Set design paths---------------------------
-- Run Design with: _create_win_setup
-- Use Design Path: <absolute project path>
-------------------------------------------------------------------
------------------------TE Reference Design-------------------------
-------------------------------------------------------------------
-- (0)  Module selection guide, project creation...prebuilt export...
-- (1)  Create minimum setup of CMD-Files and exit Batch
-- (2)  Create maximum setup of CMD-Files and exit Batch
-- (3)  (internal only) Dev
-- (4)  (internal only) Prod
-- (c)  Go to CMD-File Generation (Manual setup)
-- (d)  Go to Documentation (Web Documentation)
-- (g)  Install Board Files from Xilinx Board Store (beta)
-- (a)  Start design with unsupported Vivado Version (beta)
-- (x)  Exit Batch (nothing is done!)
----
Select (ex.:'0' for module selection guide):
```

2. Press 0 and enter to start "Module Selection Guide"
3. (optional Win OS) Generate Virtual Drive or use short directory for the reference design (for example x:\<design name>)

7 https://wiki.trenz-electronic.de/display/PD/AMD+Development+Tools#AMDDevelopmentTools-XilinxSoftware-BasicUserGuides
8 https://wiki.trenz-electronic.de/display/PD/Vivado+Projects+-+TE+Reference+Design
9 https://wiki.trenz-electronic.de/display/PD/Project+Delivery+-+AMD+devices
10 https://wiki.trenz-electronic.de/display/PD/Project+Delivery+-+AMD+devices#ProjectDeliveryAMDdevices-Currentlylimitationsoffunctionality

4. Create project and follow instructions of the product selection guide, settings file will be configured automatically during this process.
   a. optional for manual changes: Select correct device and Xilinx install path on "design_basic_settings.cmd" and create Vivado project with "vivado_create_project_guimode.cmd"

      ⚠ Note: Select correct one, see also Vivado Board Part Flow[11]

5. Create hardware description file (.xsa file) for PetaLinux project and export to prebuilt folder

run on Vivado TCL (Script generates design and export files into "\prebuilt\hardware\")

```
\prebuilt\hardware\")">
TE::hw_build_design -export_prebuilt
```

ⓘ Using Vivado GUI is the same, except file export to prebuilt folder.

6. Generate Programming Files with Vitis
   a. Run on Vivado TCL:

run on Vivado TCL (Script generates design and export files into "\prebuilt\hardware\")

```
\prebuilt\hardware\")">
TE::sw_run_vitis -all
```

Note: Scripts generate applications and bootable files, which are defined in "sw_lib\apps_list.csv"

   b. (alternative) Start Vitis with Vivado GUI or start with TE Scripts on Vivado TCL: TE::sw_run_vitis
   Note: TCL scripts generate also platform project, this must be done manuelly in case GUI is used. See Vitis[12]
   Projects contains 3 FSBL template: zynq_fsbl (FSBL modified for DDR Less application → use for Boot.bin), zynq_fsbl_app (FSBL modified for DDR Less application and with demo app included → create Boot with this FSBL and Bitstream only), zynq_fsbl_flash(FSBL modified for Flash programming →FSBL which must be selected separately to program Flash)

      ⓘ TE0722 is without DDR, so special FSBL (sources on reference designs) is needed, see also: DDR less ZYNQ Design[13]

11 https://wiki.trenz-electronic.de/display/PD/Vivado+Board+Part+Flow
12 https://wiki.trenz-electronic.de/display/PD/Vitis
13 https://wiki.trenz-electronic.de/display/PD/DDR+less+ZYNQ+Design

6 Launch

Basic Information, see TE0722 Getting Started[14]

6.1 Programming

> ⚠ 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 launch.

Xilinx documentation for programming and debugging: Vivado/Vitis/SDSoC-Xilinx Software Programming and Debugging[15]

6.1.1 Get prebuilt boot binaries

1. Run _create_win_setup.cmd/_create_linux_setup.sh and follow instructions on shell
2. Press 0 and enter to start "Module Selection Guide"
   a. Select assembly version
   b. Validate selection
   c. Select create and open delivery binary folder

   > ⓘ Note: Folder "<project folder>_binaries_<Article Name>" with subfolder "boot_<app name>" for different applications will be generated

6.1.2 QSPI-Boot mode

Option for **Boot.bin** on QSPI Flash

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:

run on Vivado TCL (Script programs BOOT.bin on QSPI flash)

```
TE::pr_program_flash -swapp zynq_fsbl_app
TE::pr_program_flash -swapp hello_te0820 (optional)
```

14 https://wiki.trenz-electronic.de/display/PD/TE0722+Getting+Started
15 https://wiki.trenz-electronic.de/display/PD/AMD+Development+Tools#AMDDevelopmentTools-XilinxSoftwareProgrammingandDebugging

6.1.3 SD-Boot mode

Xilinx Zynq devices in CLG225 package do not support SD Card boot directly from ROM bootloader. Use QSPI for primary boot and SD for secondary boot only. See also Xilinx AR#66846[16]

6.1.4 JTAG

Not used on this Example.

6.2 Usage

1. Prepare HW like described on section Programming (see page 14)
2. Connect UART USB (most cases same as JTAG)
3. Power On PCB
   boot process

   1. Zynq Boot ROM loads FSBL from SD/QSPI into OCM,

   2. FSBL init PS, programs PL using the bitstream

   3. FSBL starts application (included into the FSBL Code)

6.2.1 Standalone Application

Note: UART over J2 is used, this is only available, if PL part is configured with correct UART connection.

1. Open Serial Console (e.g. putty)
   a. Speed: 115200
   b. select COM Port

   > Win OS, see device manager, Linux OS see dmesg |grep tty (UART is *USB1)

2. Output:
   a. Default output appears only 10 times. Reboot device: force ResN Pin to GND for short time, location see: TE0722 Getting Started[17]

16 https://www.xilinx.com/support/answers/66846.html
17 https://wiki.trenz-electronic.de/display/PD/TE0722+Getting+Started

```
Xilinx First Stage Boot Loader for DDR less
Device IDCODE: 13722093
Device Name: 7z010 (2)
Device Revision: 1
--------------------------------------------------------------------------------
TE0722 TE_FsblHookBeforeHandoff_Custom



  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0xA)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x9)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x8)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x7)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x6)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x5)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x4)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x3)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x2)

  Read LIGHT SENSOR ID: Unknown PARTID 0x93 Revision: 93 Sequencer Revision:  Unknown ID 0x93
 LED D4 off (Remaining Loops 0x1)
 Loop finished...
--------------------------------------------------------------------------------
SD read txt file success
SD write binary file success
```

6.2.2 Vivado HW Manager

Open Vivado HW-Manager and add VIO signal to dashboard (*.ltx located on prebuilt folder)

- Control:
  - Enable/Disable RGB LED Counter (default on)
  - Enable/Disable different colors (default all off)

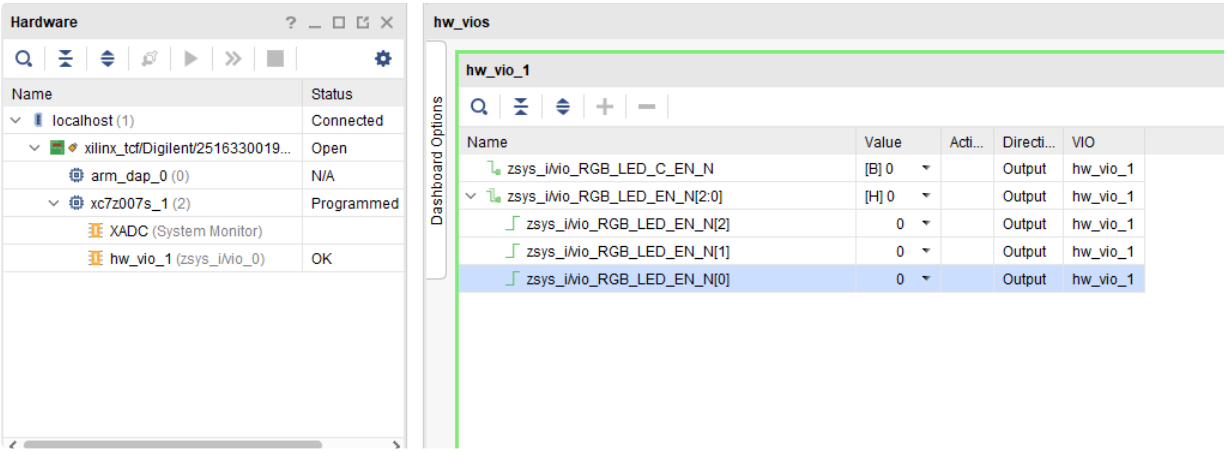
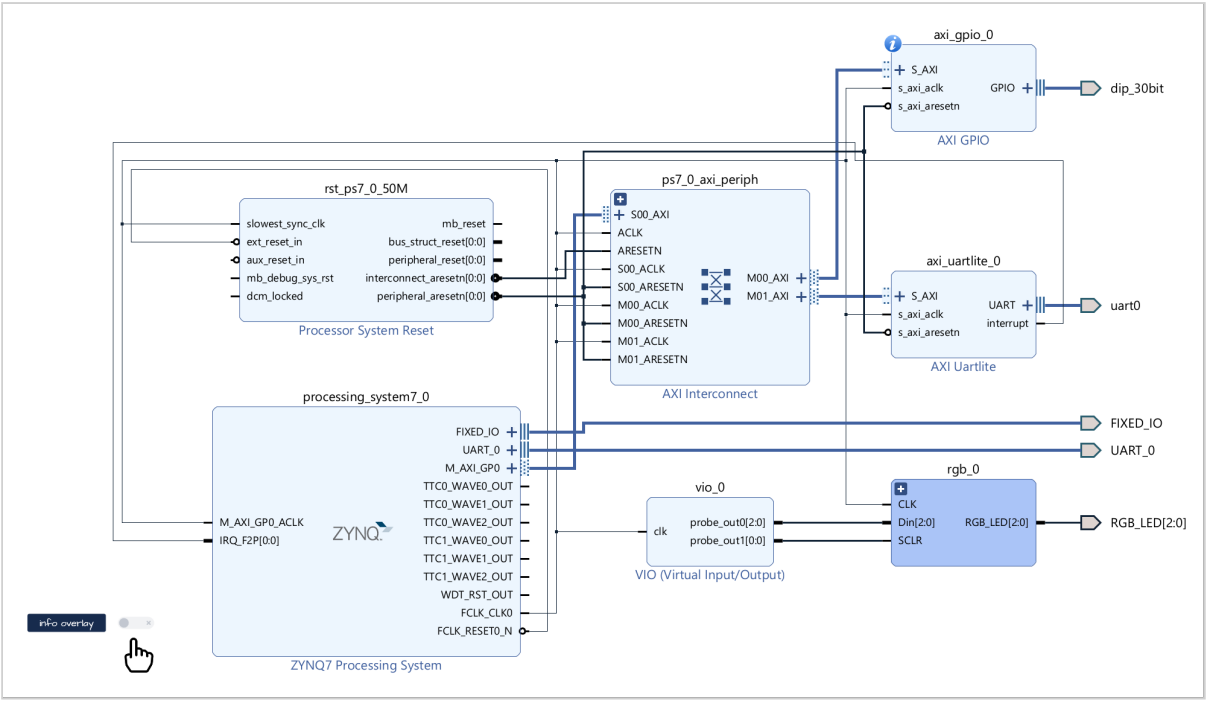


Figure 1: Vivado Hardware Manager

7 System Design - Vivado

7.1 Block Design



7.1.1 PS Interfaces

Type	Note
DDR	Disabled!
QSPI	MIO
SD	MIO
UART0	EMIO
I2C1	MIO
GPIO	MIO
SWDT0	EMIO

Type	Note
TTC0..1	EMIO

Table 5: PS Interfaces

7.2 Constraints

7.2.1 Basic module constraints

_i_bitgen_common.xdc

```
#
# Common BITGEN related settings for TE0722
#
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.UNUSEDPIN PULLUP [current_design]
```

7.2.2 Design specific constraints

_i_uart_j2xmod.xdc

```
set_property PACKAGE_PIN K15 [get_ports UART_0_txd]
set_property PACKAGE_PIN L13  [get_ports UART_0_rxd]

set_property IOSTANDARD LVCMOS33 [get_ports UART_0_*]
```

_i_io.xdc

```
#RGB LED
#R
set_property PACKAGE_PIN J15 [get_ports {RGB_LED[0]}]
#G
set_property PACKAGE_PIN L14 [get_ports {RGB_LED[1]}]
#B
set_property PACKAGE_PIN K12 [get_ports {RGB_LED[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {RGB_LED[*]}]
```

8 Software Design - Vitis

For Vitis project creation, follow instructions from:

Vitis[18]

8.1 Application

Source location: \sw_lib\sw_apps

8.1.1 zynq_fsbl

TE modified 2019.2 FSBL

General:

- Modified Files:main.c, fsbl_hooks.h/.c (search for 'TE Mod' on source code)
- Add Files: te_fsbl_hooks.h/.c(for hooks and board)\n\
- General Changes:
    - Display FSBL Banner and Device ID
    - Disable Memory initialisation on main.c

8.1.2 zynq_fsbl_app

TE modified 2019.2 FSBL

General:

- Modified Files:main.c, fsbl_hooks.h/.c (search for 'TE Mod' on source code)
- Add Files: te_fsbl_hooks.h/.c(for hooks and board)\n\
- General Changes:
    - Display FSBL Banner and Device ID
    - Disable Memory initialisation on main.c

Module Specific:

- Add Files: all TE Files start with te_*
    - Example app for LED access over MIO and sensor access over I2C
    - SD Card access read/write file

8.1.3 zynq_fsbl_flash

TE modified 2019.2 FSBL

General:

- Modified Files: main.c
- General Changes:
    - Display FSBL Banner
    - Set FSBL Boot Mode to JTAG
    - Disable Memory initialisation on main.c

18 https://wiki.trenz-electronic.de/display/PD/Vitis

9 Additional Software

No additional software is needed.

10 Appx. A: Change History and Legal Notices

10.1 Document Change History

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

Date	Document Revision	Authors	Description
2023-02-14	v.9 (see page 6)	Waldemar Hanemann[19]	• 2021.2 release
2020-04-16	v.8	John Hartfiel	• 2019.2 release
2020-04-16	v.7	John Hartfiel	• separate template for FSBL with App included
2019-05-14	v.6	John Hartfiel	• 2018.3 release
2018-08-15	v.5	John Hartfiel	• 2018.2 release
--	all	John Hartfiel[20] , Waldemar Hanemann[21]	--

Table 6: Document change history.

10.2 Legal Notices

10.3 Data Privacy

Please also note our data protection declaration at https://www.trenz-electronic.de/en/Data-protection-Privacy

19 https://wiki.trenz-electronic.de/display/~w.hanemann
20 https://wiki.trenz-electronic.de/display/~j.hartfiel
21 https://wiki.trenz-electronic.de/display/~w.hanemann

10.4 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.

10.5 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.

10.6 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.

10.7 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.

10.8 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.

10.9 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[22]. The products we supply to you are solely non-chemical products (goods). Moreover and under normal and reasonably foreseeable circumstances of application, the goods

22 http://guidance.echa.europa.eu/

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[23] 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)[24].

RoHS

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

WEEE

Information for users within the European Union in accordance with Directive 2002/96/EC of the European Parliament and of the Council of 27 January 2003 on waste electrical and electronic equipment (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.

2019-06-07

23 https://echa.europa.eu/candidate-list-table
24 http://www.echa.europa.eu/