TE0807 Test Board

Table of contents

- 1 Overview
  - 1.1 Key Features
  - 1.2 Revision History
  - 1.3 Release Notes and Know Issues
  - 1.4 Requirements
    - 1.4.1 Software
    - 1.4.2 Hardware
  - 1.5 Content
    - 1.5.1 Design Sources
    - 1.5.2 Additional Sources
    - 1.5.3 Prebuilt
    - 1.5.4 Download
- 2 Design Flow
- 3 Custom
  - 3.1 FSBL (some additional outputs only)
    - 3.1.1 Get prebuilt boot binaries
    - 3.1.2 QSPI-Boot mode
    - 3.1.3 SD-Boot mode
    - 3.1.4 JTAG
  - 3.2 Usage
- 4 System Design - Vivado
  - 4.1 Block Design
    - 4.1.1 PS Interfaces
  - 4.2 Constrains
    - 4.2.1 Basic module constrains
    - 4.2.2 Design specific constrain
- 5 Software Design - Vitis
  - 5.1 Application
    - 5.1.1 zynqmp_fsbl
    - 5.1.2 hello_te0807
- 6 Additional Software
- 7 Appx. A: Change History and Legal Notices
  - 7.1 Document Change History
  - 7.2 Legal Notices
  - 7.3 Data Privacy
  - 7.4 Document Warranty
  - 7.5 Limitation of Liability
  - 7.6 Copyright Notice
  - 7.7 Technology Licenses
  - 7.8 Environmental Protection
  - 7.9 REACH, RoHS and WEEE
- 8 Table of contents

Overview

Design Example with minimum PS Setup (DDR, QSPI, UART0) only for custom boards or easier debug via Vitis.

Refer to http://trenz.org/te0807-info for the current online version of this manual and other available documentation.

Key Features

- Vitis/Vivado 2022.2
- QSPI
- Custom Carrier (minimum PS Design with available module components only)
- Modified FSBL (some additional outputs only)

Revision History

Date	Vivado	Project Built	Authors	Description
2023-08-17	2022.2	TE0807-test_board-vivado_2022.2-build_6_20230817092607.zip TE0807-test_board_noprebuilt-vivado_2022.2-build_6_20230817092607.zip	Manuela Strücker	• 2022.2 update • new assembly variants
2022-10-17	2021.2.1	TE0807-test_board_noprebuilt-vivado_2021.2-build_18_20221017093249.zip TE0807-test_board-vivado_2021.2-build_18_20221017093249.zip	Manuela Strücker	• script update
2022-09-12	2021.2.1	TE0807-test_board_noprebuilt-vivado_2021.2-build_15_20220912085423.zip TE0807-test_board-vivado_2021.2-build_15_20220912085423.zip	Manuela Strücker	• update board part file compatible to Vivado 2021.2.1
2022-05-18	2021.2	TE0807-test_board_noprebuilt-vivado_2021.2-build_14_20220518130935.zip TE0807-test_board-vivado_2021.2-build_14_20220518130935.zip	Manuela Strücker	• 2021.2 update • new assembly variants • update document style

2021-02-08	2020.2	TE0807-test_board_noprebuilt-vivado_2020.2-build_1_20210208093457.zip TE0807-test_board-vivado_2020.2-build_1_20210208093443.zip	John Hartfiel	• 2020.2 update
2020-10-06	2019.2	TE0807-test_board_noprebuilt-vivado_2019.2-build_15_20201006121447.zip TE0807-test_board-vivado_2019.2-build_15_20201006121342.zip	John Hartfiel	• new assembly variants
2020-03-25	2019.2	TE0807-test_board_noprebuilt-vivado_2019.2-build_8_20200325082749.zip TE0807-test_board-vivado_2019.2-build_8_20200325082730.zip	John Hartfiel	• script update
2020-01-27	2019.2	TE0807-test_board_noprebuilt-vivado_2019.2-build_4_20200127075704.zip TE0807-test_board-vivado_2019.2-build_4_20200127075454.zip	John Hartfiel	• 2019.2 update • Vitis support
2019-05-22	2018.3	TE0807-test_board_noprebuilt-vivado_2018.3-build_05_20190522132408.zip TE0807-test_board-vivado_2018.3-build_05_20190522132356.zip	John Hartfiel	• custom FSBL • Note: Prebuilt for ES2 version not included
2019-02-08	2018.2	TE0807-test_board_noprebuilt-vivado_2018.2-build_04_20190207111539.zip TE0807-test_board-vivado_2018.2-build_04_20190207111524.zip	John Hartfiel	• new assembly variant
2018-09-04	2018.2	TE0807-test_board_noprebuilt-vivado_2018.2-build_03_20180904121458.zip TE0807-test_board-vivado_2018.2-build_03_20180904121522.zip	John Hartfiel	• additional notes for FSBL generated with Win SDK • changed *.bif

2018-01-18	2017.4	TE0807-test_board_noprebuilt-vivado_2017.4-build_05_20180118152119.zip TE0807-test_board-vivado_2017.4-build_05_20180118152104.zip	John Hartfiel	• rework Board Part Files
2017-11-14	2017.2	TE0807-test_board_noprebuilt-vivado_2017.2-build_05_20171114115524.zip TE0807-test_board-vivado_2017.2-build_05_20171114115511.zip	John Hartfiel	• initial release

Design Revision History

Release Notes and Know Issues

Issues	Description	Workaround	To be fixed version
Xilinx Software	Incompatibility of board files for ZynqMP with eMMC activated between 2021.2 and 2021.2.1 patch, see Xilinx Forum Request	use corresponding board files for the Vivado versions	--
QSPI Flash	Flash programming is not supported with boot mode QSPI or SD.	If flash programming fails, configure device for JTAG boot mode and try again or use older Vivado Versions for programming. (Vivado 2020.2 or 2019.2)	--

Known Issues

Requirements

Software

Software	Version	Note
Vitis	2022.2	needed, Vivado is included into Vitis installation

Software

Hardware

Basic description of TE Board Part Files is available on TE Board Part Files.

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

~~TE0807-01-07EV-ES~~	~~es2_2gb~~	~~REV01~~	~~2GB~~	~~64GB~~	~~NA~~	~~NA~~	~~Not longer supported by vivado~~
TE0807-02-07EV-1E	7ev_1e_4gb	REV02	4GB	64GB	NA	NA	NA
TE0807-02-07EV-1EK	7ev_1e_4gb	REV02	4GB	64GB	NA	NA	with heat sink
TE0807-02-4BE21-A	4eg_1e_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-7DE21-A	7ev_1e_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-7DI21-C	7ev_1i_4gb	REV02	4GB	128MB	NA	NA	without encryption
TE0807-02-7DI21-A	7ev_1i_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-4AI21-A	4cg_1i_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-5AI21-A	5cg_1i_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-7AI21-A	7cg_1i_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-7DI24-A	7ev_1i_4gb	REV02	4GB	512MB	NA	NA	NA
TE0807-02-7DE21-AK	7ev_1e_4gb	REV02	4GB	128MB	NA	NA	with heat sink
TE0807-02-4AI21-X	4cg_1i_4gb	REV02	4GB	128MB	NA	NA	U41 replaced with diode
TE0807-02-4BE21-AK	4eg_1e_4gb	REV02	4GB	128MB	NA	NA	with heat sink
TE0807-02-7DI21-AK	7ev_1i_4gb	REV02	4GB	128MB	NA	NA	with heat sink
TE0807-02-5DI21-A	5ev_1i_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-02-7NE21-A	7ev_3e_4gb	REV02	4GB	128MB	NA	NA	NA
TE0807-03-5DI21-A	5ev_1i_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-7NE21-A	7ev_3e_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-4AI21-X	4cg_1i_4gb	REV03	4GB	128MB	NA	NA	U41 replaced with diode
TE0807-03-4AI21-A	4cg_1i_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-4AI21-C	4cg_1i_4gb	REV03	4GB	128MB	NA	NA	without encryption
TE0807-03-4BE21-A	4eg_1e_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-5AI21-A	5cg_1i_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-7AI21-A	7cg_1i_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-7DE21-A*	7ev_1e_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-7DE21-AK	7ev_1e_4gb	REV03	4GB	128MB	NA	NA	with heat sink
TE0807-03-7DI21-A	7ev_1i_4gb	REV03	4GB	128MB	NA	NA	NA

TE0807-03-7DI21-C	7ev_1i_4gb	REV03	4GB	128MB	NA	NA	without encryption
TE0807-03-7DI24-A	7ev_1i_4gb	REV03	4GB	512MB	NA	NA	NA
TE0807-03-4BE21-AK	4eg_1e_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-S004	7ev_1e_me_8gb	REV03	8GB	128MB	NA	NA	CAO
TE0807-03-S005	7ev_1i_4gb	REV03	4GB	512MB	NA	NA	CAO
TE0807-03-S008	7ev_1i_me_8gb	REV03	8GB	128MB	NA	without PLL	CAO Micron DDR
TE0807-03-S014	4eg_1e_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-7DE21-AZ	7ev_1e_4gb	REV03	4GB	128MB	NA	NA	NA
TE0807-03-S011	7ev_1i_me_8gb	REV03	8GB	128MB	NA	without PLL	CAO Micron DDR

*used as reference

Hardware Modules

Note: Design contains also Board Part Files for TE0807+TEBF0808 configuration, this board part files are not used for this reference design.

Design supports following carriers:

Carrier Model	Notes
Custom PCB	use simple Board Part files, if MIO connected is different to TEBF0808
TEBF0808*	Used as reference carrier.
TEBT0808-01	Change UART0 to UART1 (MIO68...69) and regenerate design

*used as reference

Hardware Carrier

Additional HW Requirements:

Additional Hardware	Notes
---	---

*used as reference

Additional Hardware

Content

For general structure and usage of the reference design, see Project Delivery - AMD devices

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

Design sources

Additional Sources

Type	Location	Notes
---	---	---

Additional design sources

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-Description-File	*.xsa	Exported Vivado hardware description file for Vitis and PetaLinux
LabTools Project-File	*.lpr	Vivado Labtools Project File
Software-Application-File	*.elf	Software Application for Zynq or MicroBlaze Processor Systems

Prebuilt files (only on ZIP with prebult content)

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.

Reference Design is available on:

- TE0807 "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 launch.

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

See also:

- AMD Development Tools#XilinxSoftware-BasicUserGuides
- Vivado Projects - TE Reference Design
- 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/Vitis GUI. For currently Scripts limitations on Win and Linux OS see: Project Delivery Currently limitations of functionality

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. Create project and follow instructions of the product selection guide, settings file will be configured automatically during this process.
    - 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
>
> Important: Use Board Part Files, which **did not** end with *_tebf0808

4. Create hardware description file (.xsa file) and export to prebuilt folder

run on Vivado TCL (Script generates design and export files into "<project folder>\prebuilt\hardware\<short name>")

```
TE::hw_build_design -export_prebuilt
```

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

5. Generate Programming Files with Vitis

run on Vivado TCL (Script generates applications and bootable files, which are defined in "test_board\sw_lib\apps_list.csv")

```
TE::sw_run_vitis -all
TE::sw_run_vitis (optional; Start Vitis from Vivado GUI or start with TE Scripts on Vivado TCL)
```

> ⚠ TCL scripts generate also platform project, this must be done manually in case GUI is used. See Vitis

Launch

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

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

QSPI-Boot mode

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"

run on Vivado TCL (Script programs BOOT.bin on QSPI flash)
`TE::pr_program_flash -swapp hello_te0807`

SD-Boot mode

This does not work, because SD controller is not selected on PS.

JTAG

Load configuration and Application with Vitis Debugger into device

Usage

QSPI Boot:

1. Prepare HW like described on section Programming
2. Connect UART USB (most cases same as JTAG)
3. Select QSPI as Boot Mode

> Note: See TRM of the Carrier, which is used.

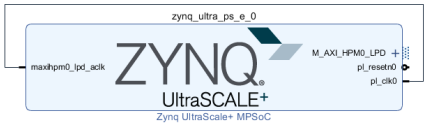
4. Power On PCB

   1. ZynqMP Boot ROM FSBL from QSPI into OCM,

   2. FSBL init the PS, programs the PL using the bitstream and loads PMU, ATF and U-boot from QSPI into DDR

System Design - Vivado

Block Design



Block Design

PS Interfaces

Activated interfaces:

Type	Note
DDR	
QSPI	MIO
UART0	MIO, please select other one, if you have connected UART to second controller or other MIO
SWDT0..1	
TTC0..3	

PS Interfaces

Constrains

Basic module constrains

_i_bitgen.xdc

```
set_property BITSTREAM.GENERAL.COMPRESS TRUE [current_design]
set_property BITSTREAM.CONFIG.UNUSEDPIN PULLNONE [current_design]
```

Design specific constrain

Not needed.

Software Design - Vitis

For Vitis project creation, follow instructions from:

Vitis

Application

Template location: "<project folder>\sw_lib\sw_apps\"

zynqmp_fsbl

TE modified 2022.2 FSBL

General:

- Modified Files: xfsbl_main.c, xfsbl_hooks.h/.c, xfsbl_board.h/.c (search for 'TE Mod' on source code)
- Add Files: te_xfsbl_hooks.h/.c (for hooks and board)
- General Changes:
  - Display FSBL Banner and Device Name

hello_te0807

Hello TE0807 is a Xilinx Hello World example as endless loop instead of one console output.

Additional Software

No additional software is needed.

Appx. A: Change History and Legal Notices

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
Error renderi ng macro 'page-info' Ambiguo us method	**Error renderi ng macro 'page-info'** Ambiguo us method	**Error renderi ng macro 'page-info'** Ambiguo us method	• 2022.2 update • new assembly variants

overloading for method jdk.proxy279.$Proxy4022#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page] due to overlapping prototypes between: [interface com.

overloading for method jdk.proxy279.$Proxy4022#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page] due to overlapping prototypes between: [interface com.

overloading for method jdk.proxy279.$Proxy4022#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page] due to overlapping prototypes between: [interface com.

atlassian . confluence.user.ConfluenceUser, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject] [interface com.atlassian.user.User, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject]	atlassian . confluence.user.ConfluenceUser, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject] [interface com.atlassian.user.User, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject]	atlassian . confluence.user.ConfluenceUser, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject] [interface com.atlassian.user.User, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject]	

2022-10-17	v.21	Manuela Strücker	• script update
2022-09-12	v.20	Manuela Strücker	• update board part file compatible to Vivado 2021.2.1
2022-09-06	v.19	Manuela Strücker	• 2021.2 update • new assembly variants • update document style
2021-02-08	v.15	John Hartfiel	• new assembly variants • document style update
2020-10-06	v.14	John Hartfiel	• new assembly variants
2020-03-25	v.13	John Hartfiel	• script update
2020-01-27	v.12	John Hartfiel	• Release 2019.2 • new assembly variants
2019-05-22	v.10	John Hartfiel	• Release 2018.3
2019-02-07	v.9	John Hartfiel	• new assembly variant
2018-09-04	v.7	John Hartfiel	• Release 2018.2
2018-02-08	v.5	John Hartfiel	• Release 2017.4
2017-11-14	v.3	John Hartfiel	• Release 2017.2
--	all	**Error renderi ng macro 'page-info'**	--

Ambiguous method overloading for method jdk.proxy279.$Proxy4022#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page] due to overlapping prototypes between

:
[interfac
e com.
atlassian
.
confluen
ce.user.
Conflue
nceUser
, class
java.
lang.
String,
class
com.
atlassian
.
confluen
ce.core.
Content
EntityOb
ject]
[interfac
e com.
atlassian
.user.
User,
class
java.
lang.
String,
class
com.
atlassian
.
confluen
ce.core.
Content
EntityOb

		ject]	

Legal Notices

Data Privacy

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

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.

Error rendering macro 'page-info'

Ambiguous method overloading for method jdk.proxy279.$Proxy4022#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page] due to overlapping prototypes between: [interface com.atlassian.confluence.user.ConfluenceUser, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject] [interface com.atlassian.user.User, class java.lang.String, class com.atlassian.confluence.core.ContentEntityObject]