TEI0003 Test Board

Table of contents

- 1 Overview
  - 1.1 Key Features
  - 1.2 Block Diagram
  - 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 Prebuilt
    - 1.5.3 Download
- 2 Design Flow
- 3 Launch
  - 3.1 Programming
    - 3.1.1 Get prebuilt boot binaries
    - 3.1.2 Configuration Flash
    - 3.1.3 JTAG
  - 3.2 Usage
    - 3.2.1 UART
- 4 System Design - Quartus
  - 4.1 Block Design
- 5 Software Design - SDK
  - 5.1 Application
    - 5.1.1 test_tei0003
    - 5.1.2 hello_tei0003
- 6 Appx. A: Change History and Legal Notices
  - 6.1 Document Change History
  - 6.2 Legal Notices
  - 6.3 Data Privacy
  - 6.4 Document Warranty
  - 6.5 Limitation of Liability
  - 6.6 Copyright Notice
  - 6.7 Technology Licenses
  - 6.8 Environmental Protection
  - 6.9 REACH, RoHS and WEEE
- 7 Table of contents

Overview

Nios V/m design example with SDRAM controller (AXI4), 3-axis Accelerometer and different LED sequences.

Refer to http://trenz.org/tei0003-info for the current online version of this manual and other available documentation.

Key Features

- Quartus Prime Lite 22.1std
- Nios V/m
- SPI
- UART
- SDRAM controller (AXI4)
  - Source code: https://github.com/ultraembedded/core_sdram_axi4
- 3-axis Accelerometer
- User LEDs
- User buttons

Revision History

Date	Quartus	Project Built	Authors	Description
2023-09-13	22.1 Lite	TEI0003-test_board_noprebuilt-quartus_22.1std.2-20230913161158.zip TEI0003-test_board-quartus_22.1std.2-20230913161214.zip	Thomas Dück	• update to Quartus Prime Lite 22.1 • new board variants • change to Nios V/m • TE scripts update
2022-04-21	21.1 Lite	TEI0003-test_board_noprebuilt-quartus_21.1.0-20220421150041.zip TEI0003-test_board-quartus_21.1.0-20220421150114.zip	Thomas Dück	• update to Quartus Prime Lite 21.1
2021-07-09	20.1 Lite	TEI0003-test_board_noprebuilt-quartus_20.1.1-20210709110930.zip TEI0003-test_board-quartus_20.1.1-20210709110847.zip	Thomas Dück	• update to Quartus Prime Lite 20.1 • TE scripts update

2020-10-19	19.1 Lite	TEI0003-test_board_noprebuilt-quartus_19.1.0-20201019101802.zip TEI0003-test_board-quartus_19.1.0-20201019101738.zip	Thomas Dück	• script update • bugfixes
2020-05-13	19.1 Lite	TEI0003-test_board_noprebuilt-quartus_19.1.0-20200513080815.zip TEI0003-test_board-quartus_19.1.0-20200513081030.zip	Thomas Dück	• 19.1 update
2019-11-11	18.1	TEI0003-test_board_noprebuilt-quartus_18.1-20191111104152.zip TEI0003-test_board-quartus_18.1-20191111104339.zip	Thomas Dück	• add bash files for Linux OS
2019-10-29	18.1	TEI0003-test_board_noprebuilt-quartus_18.1-20191029121432.zip TEI0003-test_board-quartus_18.1-20191029121225.zip	Thomas Dück	• create project with TE scripts • new board variants
2019-04-02	18.1	TEI0003-02-test_board-quartus_18.1-20190402.zip	Thomas Dück	• initial release

Design Revision History

Release Notes and Know Issues

Issues	Description	Workaround	To be fixed version
No known issues	---	---	---

Known Issues

Requirements

Software

Software	Version	Note
Quartus Prime Lite	22.1std	Nios V license is needed. For more information see: Intel Nios V Processors
RiscFree IDE for Intel FPGAs	22.1std	needed

Software

Hardware

Complete List is available on *<project folder>/board_files/*_devices.csv*

Design supports following modules:

Module Model	PCB Revision Support	Board Part Short Name	SDRAM	Configuration flash	Others	Notes
TEI0003-02	REV01, REV02	QFCR1	8MByte	2MByte	--	--
TEI0003-02A	REV2	QFCR1	8MByte	2MByte	--	--
TEI0003-03A-S001	REV03	QFCR4_S1	8MByte	8MByte	--	--
TEI0003-03-QFCR1A	REV03	QFCR1	8MByte	2MByte	--	--
TEI0003-03-QFCR4A*	REV03	QFCR4	8MByte	8MByte	--	--

*used as reference

Hardware Modules

Design supports following carriers:

Carrier Model	Notes
---	

*used as reference

Hardware Carrier

Additional HW Requirements:

Additional Hardware	Notes
Micro USB cable for JTAG/UART	--

*used as reference

Additional Hardware

Content

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

Design Sources

Type	Location	Notes
Quartus	<project folder>/source_files/quartus/	Quartus project will be generated by TE Scripts
Software	<project folder>/source_files/software/	Additional software will be generated by TE Scripts

Design sources

Prebuilt

File	File-Extension	Description
SOPC Information File	*.sopcinfo	File with description of the *. qsys file to create software for the target hardware
SRAM Object File	*.sof	Ram configuration file
JTAG indirect configuration file	*.jic	Flash configuration file
Diverse Reports	---	Report files in different formats
Software Application File	*.elf	Software application for Nios V processor system

Prebuilt files (only in ZIP file with prebuilt content)

Download

Reference Design is only usable with the specified Quartus version. Do never use different versions of Quartus software for the same project.

Reference Design is available on:

- TEI0003 "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 Quartus Design Flow.

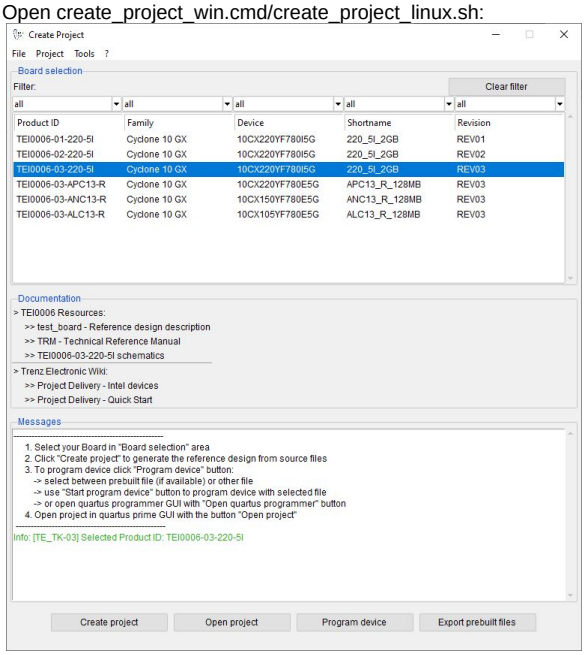
See also:

- Project Delivery - Intel devices

The Trenz Electronic FPGA Reference Designs are TCL-script based projects. To create a project, open a project or program a device execute "create_project_win.cmd" on Windows OS and "create_project_linux.sh" on Linux OS.

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

1. Open create_project_win.cmd/create_project_linux.sh:



2. Select Board in "Board selection"
3. Click on "Create project" button to create project
    a. (optional for manual changes) Select correct quartus installation path in "*<project folder>/settings/design_basic_settings.tcl*"

Launch

Programming

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

Get prebuilt boot binaries

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

1. Run create_project_win.cmd/create_project_linux.sh
2. Select Module in 'Board selection'
3. Click on 'Export prebuilt files' button
    a. Folder <*project folder*>/*_binaries_*<*Article Name*> with subfolder *programming_files* will be generated and opened

Configuration Flash

1. Connect the Module to USB-Port
2. Open create_project_win.cmd/create_project_linux.sh
3. Select correct board in "Board selection"
4. Click on "Program device" button
    a. if prebuilt files are available: select "Program prebuilt file"

b. using own generated programming file: select "Program other file" and click on "Browse ..." to open own generated programming file
c. (optional) click on "Open programmer GUI" to program device with Quartus programmer GUI

5. Click on "Start program device" button

JTAG

Not used on this example.

Usage

1. Prepare Hardware like described on section Programming
2. Connect UART USB (most cases same as JTAG)

UART

1. Open Serial Console (e.g. PuTTY)
   a. select COM Port

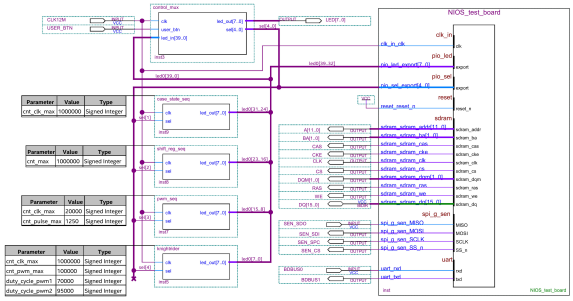
> Win OS: see device manager
>
> Linux OS: see ls -l dev/serial/by-id (UART is *USB1)

   b. Speed: 115200
2. Press reset button
3. Console output depends on used Software project, see Software Design - SDK#Application

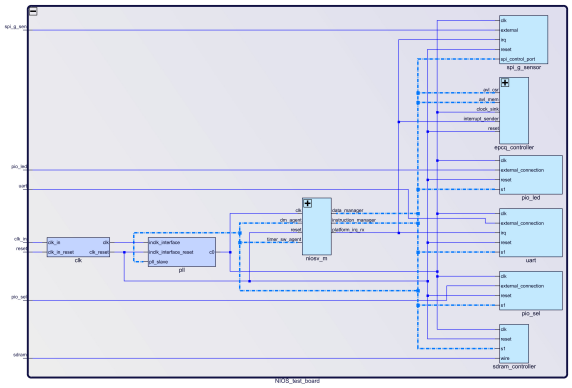
System Design - Quartus

Block Design

The block designs may differ depending on the assembly variant.



Block Design - Project



Block Design - NIOS_test_board.qsys

Block Design - Platform Designer

Software Design - SDK

Application

Used software project depends on board assembly variant. Template location: *<project folder> /source_files/software/*

test_tei0003

Software example to test TEI0003 module.

- You can select between following modes by pressing the user button:
    1. Spirit level
    2. Case statement sequence
    3. Shift register sequence
    4. Knightrider sequence
    5. Pulse-width modulation sequence

hello_tei0003

Software example with 'Hello TEI0003' console output in endless loop.

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.

Date	Document Revision	Authors	Description
			• update to Quartus Prime Lite 22.1 • document style update

Error rendering macro 'page-info' Ambiguous method overloading for method jdk.proxy24.$Proxy3572#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page]	**Error rendering macro 'page-info'** Ambiguous method overloading for method jdk.proxy24.$Proxy3572#hasContentLevelPermission. Cannot resolve which method to invoke for [null, class java.lang.String, class com.atlassian.confluence.pages.Page]	**Error rendering macro 'page-info'** Ambiguous method overloading for method jdk.proxy24.$Proxy3572#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.

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.

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.core. ContentEntityObject]	atlassian . confluence.core. ContentEntityObject]	
2022-04-22	v.9	Thomas Dück	• update to Quartus Prime Lite 21.1
2021-07-09	v.8	Thomas Dück	• update to Quartus Prime Lite 20.1 • document style update • script update
2020-10-09	v.7	Thomas Dück	• script update • bugfixes
2020-05-13	v.6	Thomas Dück	• 19.1 release
2019-11-11	v.5	Thomas Dück	• add bash files for Linux OS
2019-10-29	v.3	Thomas Dück	• change design to TE scripts • new board variants
2019-04-03	v.2	Thomas Dück	• Initial release 18.1
--	all	**Error rendering macro 'page-info'** Ambiguous method	--

overloading for method jdk.proxy24.$Proxy3572#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]	

Document change history.

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.proxy244.$Proxy3572#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]