

TEF1001 Test Board

Table of contents

- 1 [Table of contents](#)
- 2 [Overview](#)
 - 2.1 [Key Features](#)
 - 2.2 [Revision History](#)
 - 2.3 [Release Notes and Know Issues](#)
 - 2.4 [Requirements](#)
 - 2.4.1 [Software](#)
 - 2.4.2 [Hardware](#)
 - 2.5 [Content](#)
 - 2.5.1 [Design Sources](#)
 - 2.5.2 [Additional Sources](#)
 - 2.5.3 [Prebuilt](#)
 - 2.5.4 [Download](#)
- 3 [Design Flow](#)
- 4 [Launch](#)
 - 4.1 [Programming](#)
 - 4.1.1 [QSPI](#)
 - 4.1.2 [SD](#)
 - 4.1.3 [JTAG](#)
 - 4.2 [Usage](#)
 - 4.2.1 [JTAG/UART Console:](#)
 - 4.2.2 [Vivado HW Manager:](#)
 - 4.2.3 [PC:](#)
- 5 [System Design - Vivado](#)
 - 5.1 [Block Design](#)
 - 5.2 [Constrains](#)
 - 5.2.1 [Basic module constrains](#)
 - 5.2.2 [Design specific constrain](#)
- 6 [Software Design - SDK/HSI](#)
 - 6.1 [Application](#)
 - 6.1.1 [hello_tef1001](#)
 - 6.1.2 [SI5338_Init](#)
- 7 [Additional Software](#)
 - 7.1 [SI5338](#)
- 8 [Appx. A: Change History and Legal Notices](#)
 - 8.1 [Document Change History](#)
 - 8.2 [Legal Notices](#)
 - 8.3 [Data Privacy](#)
 - 8.4 [Document Warranty](#)
 - 8.5 [Limitation of Liability](#)
 - 8.6 [Copyright Notice](#)
 - 8.7 [Technology Licenses](#)
 - 8.8 [Environmental Protection](#)
 - 8.9 [REACH, RoHS and WEEE](#)

Overview

TEF1001 SI5338 Configuration, DDR Configuration and PCIe Core Example Design.

Refer to <http://trenz.org/tef1001-info> for the current online version of this manual and other available documentation.

Key Features

- MicroBlaze
- I2C
- Flash
- FMeter
- PCIe
- SI5338
- DDR3 ECC SODIMM (currently ECC disabled)

Revision History

Date	Vivado	Project Built	Authors	Description
2018-10-25	2018.2	TEF1001-test_board-vivado_2018.2-build_03_20181025165553.zip TEF1001-test_board_noprebuilt-vivado_2018.2-build_03_20181025165625.zip	John Hartfiel	• Add -410 assembly variant • Add some notes on Board part Files (summary window description)
2018-10-25	2018.2	TEF1001-test_board_noprebuilt-vivado_2018.2-build_03_20181024154054.zip TEF1001-test_board-vivado_2018.2-build_03_20181024154034.zip	John Hartfiel	• 2018.2 • add TEF1001-02 • MIG Configuration for AW12P7218BLK0M (4GB for REV01) • MIG Configuration for AW24P7228BLK0M (8GB for REV02) • BUGFIX QSPI IP configuration • add SREC to load application into DDR
2018-03-07	2017.4	TEF1001-test_board_noprebuilt-vivado_2017.4-build_06_20180307102924.zip TEF1001-test_board-vivado_2017.4-build_06_20180307102845.zip	John Hartfiel	• 2017.4 update • new assembly variant
2017-11-28	2017.2	TEF1001-test_board-vivado_2017.2-build_05_20171128114335.zip TEF1001-test_board_noprebuilt-vivado_2017.2-build_05_20171128114350.zip	John Hartfiel	• initial release

Design Revision History

Release Notes and Know Issues

Issues	Description	Workaround	To be fixed version
DDR3 ECC SODIMM	DDR3 does not work with ECC enabled	Disable ECC: • for Block Design MIG with AXI Interface, create 64Bit MIG • for RTL MIG with Native Interface, disable ECC on MIG configuration and use 72Bit for Data	---

Known Issues

Requirements

Software

Software	Version	Note
Vivado	2018.2	needed
SDK	2018.2	needed

SI5338 Clock Builder	---	optional
----------------------	-----	----------

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	Others	Notes
TEF1001-01-160-2I	1_160_2	REV01	DDR3 ECC SODIMM*	32MB		DDR configured for AW12P7218BLK0M (4GB for REV01)
TEF1001-01-325-2C	1_325_2	REV01	DDR3 ECC SODIMM*	32MB		DDR configured for AW12P7218BLK0M (4GB for REV01)
TEF1001-02-160-2I	2_160_2	REV02	DDR3 ECC SODIMM	32MB		DDR configured for AW24P7228BLK0M (8GB for REV02)
TEF1001-02-325-2C	2_325_2	REV02	DDR3 ECC SODIMM	32MB		DDR configured for AW24P7228BLK0M (8GB for REV02)
TEF1001-02-410-2I	2_410_2	REV02	DDR3 ECC SODIMM	32MB		DDR configured for AW24P7228BLK0M (8GB for REV02)

* PCB REV01 DDR3 ECC SODIMM is limited to 4GB, for PCB REV02 up to 8GB is possible

Hardware Modules

Design supports following carriers:

Carrier Model	Notes
PC with PCIe Card slot	
Stand-alone	

Hardware Carrier

Additional HW Requirements:

Additional Hardware	Notes
JTAG Programmer	- TE0790 with TE0791 for CPLD or FPGA - Xilinx compatible JTAG programmer for FPGA

DDR3 (204 Pin with ECC)	- for example: - AW12P7218BLK0M (max. 4GB for REV01) - AW24P7228BLK0M (max. 8GB for REV02)
-------------------------	---

Additional Hardware

Content

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

Design Sources

Type	Location	Notes
Vivado	<design name>/block_design <design name>/constraints <design name>/ip_lib	Vivado Project will be generated by TE Scripts
SDK/HSI	<design name>/sw_lib	Additional Software Template for SDK/HSI and apps_list.csv with settings for HSI

Design sources

Additional Sources

Type	Location	Notes
SI5338	<design name>/misc/SI5338	SI5338 Project with current PLL Configuration

Additional design sources

Prebuilt

File	File-Extension	Description
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	*.hdf	Exported Vivado Hardware Specification for SDK/HSI and PetaLinux
LabTools Project-File	*.lpr	Vivado Labtools Project File
MCS-File	*.mcs	Flash Configuration File with Boot-Image (MicroBlaze or FPGA part only)
MMI-File	*.mmi	File with BRAM-Location to generate MCS or BIT-File with *.elf content (MicroBlaze only)
Software-Application-File	*.elf	Software Application for Zynq or MicroBlaze Processor Systems
SREC-File	*.srec	Converted Software Application for MicroBlaze Processor Systems

Prebuilt files (only on ZIP with prebuilt content)

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:

- [TEF1001 "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:

- [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/SDK GUI. For currently Scripts limitations on Win and Linux OS see: [Project Delivery Currently limitations of functionality](#)

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

```
C:\WINDOWS\system32\cmd.exe
B:\Design\cores\2017.2\design\TEF1001\test_board>setlocal
-----Set design paths-----
-- Run Design with: _create_win_setup
-- Use Design Path: B:\Design\cores\2017.2\design\TEF1001\test_board\
-----TE Reference Design-----
-- (c) Go to CMD-File Generation (Manual setup)
-- (d) Go to Documentation (Web Documentation)
-- (x) Exit Batch (nothing is done!)
-- (0) Create minimum setup of CMD-Files and exit Batch
-- (1) Create maximum setup of CMD-Files and exit Batch
-----
Select (ex.: '0' for min setup):
```

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_prebuiltNote: Script generate design and export files into \prebuilt\hardware\<short dir>. Use GUI is the same, except file export to prebuilt folder
6. Generate Programming Files with HSI/SDK

- a. Start with TE Scripts on Vivado TCL: `TE::sw_run_hsi`
(optional) Start SDK with Vivado GUI or start with TE Scripts on Vivado TCL: `TE::sw_run_sdk` to generate files manually
Note: See [SDK Projects](#)
- b. (optional)Copy "prebuilt\software\<short dir>\srec_spi_bootloader.elf" into "firmware\microblaze_0" (replace shipped one) and regenerate design again (HW (Step5)+SW(Step6 only a.))

Launch

Programming



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

Xilinx documentation for programming and debugging: [Vivado/SDK/SDSoC-Xilinx Software Programming and Debugging](#)

QSPI

1. Connect JTAG and Power ON PC
2. Open Vivado Project with "vivado_open_existing_project_gui mode.cmd" or if not created, create with "vivado_create_project_gui mode.cmd"
3. Type on Vivado TCL Console: `TE::pr_program_flash_mcsfile -swapp hello_tef1001`
4. Reboot PC

SD

Not supported.

JTAG

- Connect Vivado HW Manager and program FPGA
Note: PCIe enumeration will be not done in this case. SREC Bootloader need Hello TEF1001 application on QSPI Flash for output

Usage

1. Prepare HW like described on section [Programming](#)
2. Power On PCB
Note: 1. FPGA Load Bitfile into FPGA, modified SREC Bootloader configure SI5338 and load application from QSPI into DDR (Depends on linker script)

JTAG/UART Console:

- Launch the XSDB console on SDK (Xilinx XSCT Console):
 - type: connect
 - type: targets -set -filter {name =~ "MicroBlaze Debug*"} -index 0
 - type: jtagterminal -start

- Separat console starts:

```

C:\Xilinx\SDK\2018.2\bin\unwrapped\win64-o\tclsh85t.exe

Terminal requirements :
(i) Processor's STDOUT is redirected to the ARM DCC/MDM UART
(ii) Processor's STDIN is redirected to the ARM DCC/MDM UART.
Then, text input from this console will be sent to DCC/MDM's UART port.
NOTE: This is a line-buffered console and you have to press "Enter"
to send a string of characters to DCC/MDM.

Hello TEF1001 (Loop: 329)
Hello TEF1001 (Loop: 330)
Hello TEF1001 (Loop: 331)
Hello TEF1001 (Loop: 332)
Hello TEF1001 (Loop: 333)
Hello TEF1001 (Loop: 334)
Hello TEF1001 (Loop: 335)
Hello TEF1001 (Loop: 336)
Hello TEF1001 (Loop: 337)
Hello TEF1001 (Loop: 338)
Hello TEF1001 (Loop: 339)

```

Vivado HW Manager:

1. Open Vivado HW Manager
2. Add VIO to Dashboard:
3. Set Radix to unsigned integer for FMeterCLKs (labt_SI_*)
4. Control:
 - a. USER LEDs are selectable
Note USR_CPLD_LED on PCB REV1 and REV02, USR_LED Matrix only on REV02
 - b. Optional PCIe Core Reset (on FPGA only)
 - c. Optional System Reset (on FPGA only)
5. Read: All SI5338 CLKs (Unit Hz), PCIe Cor MMCM Lock signal, MIG MMCM Lock signal, MIG Init Calibration Done

HARDWARE MANAGER - localhost\Xilinx_tcl\Xilinx\00001176835d01

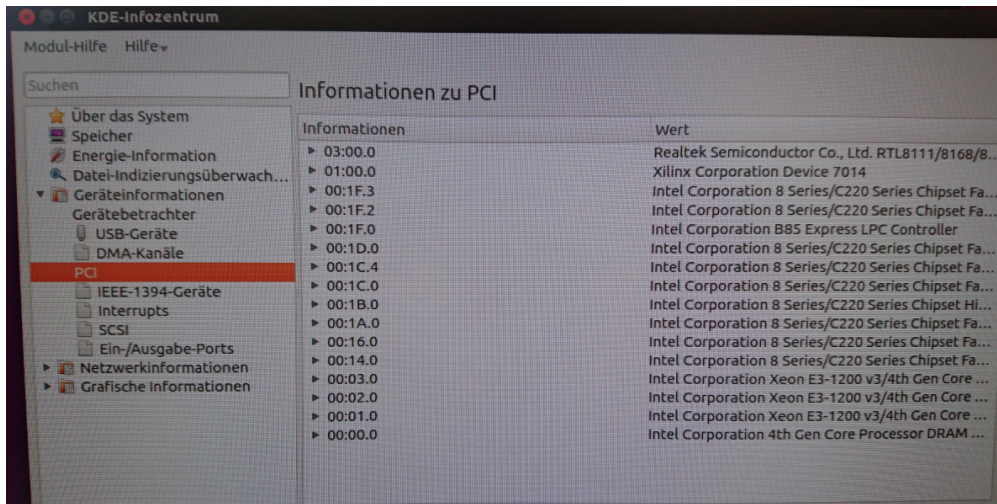
Name	Status
localhost (1)	Connected
Xilinx_tcl\Xilinx\00001176835d01	Open
xc7k160t_0 (3)	Programmed
XADC (System Monitor)	
hw_vio_1 (msys_vio_0)	OK - Outputs F
mt25qu256-spi-x1_x2_x4	

hw_vio_1

Name	Value	Activity	Direction	VIO
msys_v/axi_pcie_0_mmcm_lock	[B] 1		Input	hw_vio_1
msys_v/labt_SI_CLK0[31:0]	[U] 62501923		Input	hw_vio_1
msys_v/labt_SI_CLK1[31:0]	[U] 125003845		Input	hw_vio_1
msys_v/labt_SI_CLK2[31:0]	[U] 250007694		Input	hw_vio_1
msys_v/labt_SI_MGT[31:0]	[U] 156254808		Input	hw_vio_1
msys_v/mig_7series_0_init_calib_complete	[B] 1		Input	hw_vio_1
msys_v/mig_7series_0_mmcm_locked	[B] 1		Input	hw_vio_1
msys_v/pcie_rst_n	[B] 1		Output	hw_vio_1
msys_v/sys_rst_n	[B] 1		Input	hw_vio_1
msys_v/sys_rst_n_1	[B] 1		Output	hw_vio_1
msys_v/USR_CPLD_LED[0:0]	[B] 0		Output	hw_vio_1
msys_v/USR_LED[9:0]	[H] 000		Output	hw_vio_1

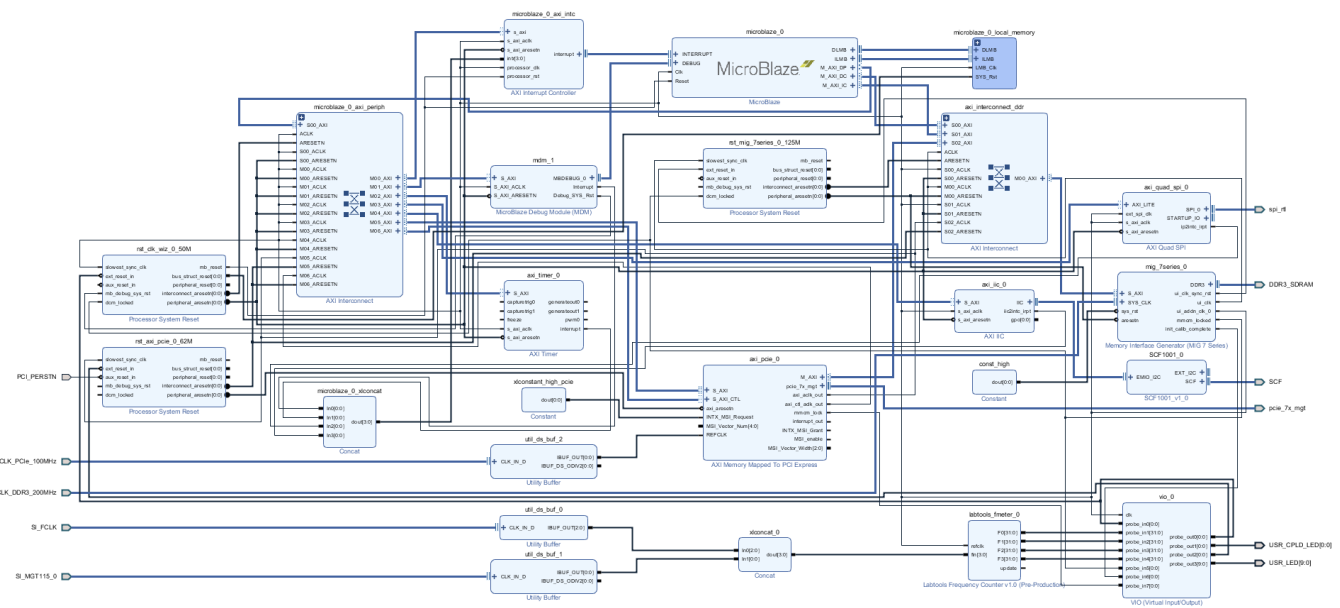
PC:

- Use for example PCI-Z (Win) or KInfoCenter (Linux) or lspci command (Linux console) to detect PCIe Card



System Design - Vivado

Block Design



Block Design

Constrains

Basic module constrains

i_bitgen_common.xdc

```
set_property BITSTREAM.GENERAL.COMPRESS TRUE [current_design]
set_property BITSTREAM.CONFIG.CONFIGRATE 66 [current_design]
set_property CONFIG_VOLTAGE 1.8 [current_design]
set_property CFGBVS GND [current_design]
set_property CONFIG_MODE SPIx4 [current_design]
set_property BITSTREAM.CONFIG.SPI_32BIT_ADDR YES [current_design]
set_property BITSTREAM.CONFIG.SPI_BUSWIDTH 4 [current_design]
set_property BITSTREAM.CONFIG.M1PIN PULLNONE [current_design]
set_property BITSTREAM.CONFIG.M2PIN PULLNONE [current_design]
set_property BITSTREAM.CONFIG.M0PIN PULLNONE [current_design]

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

i_common.xdc

```
#
#
#
set_property BITSTREAM.CONFIG.UNUSEDPIN PULLUP [current_design]
```

Design specific constrain

i_io.xdc

```
#-----
#USER LED Matrix
#

#USER LEDS CONNECTED TO A FMC_ADJ VCC0 BANK (default config 1.8V)
set_property PACKAGE_PIN K25 [get_ports {USR_LED[0]}]
set_property PACKAGE_PIN K26 [get_ports {USR_LED[1]}]
set_property PACKAGE_PIN P26 [get_ports {USR_LED[2]}]
set_property PACKAGE_PIN R26 [get_ports {USR_LED[3]}]
set_property PACKAGE_PIN N16 [get_ports {USR_LED[4]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[0]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[1]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[2]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[3]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[4]}]

#USER LEDS CONNECTED TO A 1.8V VCC0 BANK
set_property PACKAGE_PIN J26 [get_ports {USR_LED[5]}]
set_property PACKAGE_PIN H26 [get_ports {USR_LED[6]}]
set_property PACKAGE_PIN E26 [get_ports {USR_LED[7]}]
set_property PACKAGE_PIN A24 [get_ports {USR_LED[8]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[5]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[6]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[7]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[8]}]

#USER LED CONNECTED TO A FMC_ADJ VCC0 BANK (default config 1.8V)
set_property PACKAGE_PIN F19 [get_ports {USR_LED[9]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_LED[9]}]
```

```

#-----
#USER LED over CPLD
# FEX11
set_property PACKAGE_PIN B21 [get_ports {USR_CPLD_LED[0]}]
set_property IOSTANDARD LVCMOS18 [get_ports {USR_CPLD_LED[0]}]
#-----
#CLK DDR3
#AC9 /AD9 for REV01
#AB11 / AC11 for REV02
##set_property PACKAGE_PIN AB11 [get_ports CLK_DDR3_200MHz_clk_p]
##set_property PACKAGE_PIN AC11 [get_ports CLK_DDR3_200MHz_clk_n]
##set_property IOSTANDARD DIFF_SSTL15 [get_ports CLK_DDR3_200MHz_clk_p]
##set_property IOSTANDARD DIFF_SSTL15 [get_ports CLK_DDR3_200MHz_clk_n]
#-----
#QSPI
set_property PACKAGE_PIN C23 [get_ports {spi_rtl_ss_io[0]}]
set_property IOSTANDARD LVCMOS18 [get_ports {spi_rtl_ss_io[0]}]
set_property PACKAGE_PIN B24 [get_ports spi_rtl_io0_io]
set_property PACKAGE_PIN A25 [get_ports spi_rtl_io1_io]
set_property PACKAGE_PIN B22 [get_ports spi_rtl_io2_io]
set_property PACKAGE_PIN A22 [get_ports spi_rtl_io3_io]
set_property IOSTANDARD LVCMOS18 [get_ports spi_rtl_io0_io]
set_property IOSTANDARD LVCMOS18 [get_ports spi_rtl_io1_io]
set_property IOSTANDARD LVCMOS18 [get_ports spi_rtl_io2_io]
set_property IOSTANDARD LVCMOS18 [get_ports spi_rtl_io3_io]
#-----
#IIC to CPLD
set_property PACKAGE_PIN G26 [get_ports SCF_cpld_1_scl]
set_property PACKAGE_PIN F25 [get_ports SCF_cpld_14_oe]
set_property PACKAGE_PIN G25 [get_ports SCF_cpld_16_sda]
set_property IOSTANDARD LVCMOS18 [get_ports SCF_cpld_1_scl]
set_property IOSTANDARD LVCMOS18 [get_ports SCF_cpld_14_oe]
set_property IOSTANDARD LVCMOS18 [get_ports SCF_cpld_16_sda]
#-----
#SI5338 CLKs
set_property PACKAGE_PIN H6 [get_ports {SI_MGT115_0_clk_p[0]}]

set_property PACKAGE_PIN G22 [get_ports {SI_FCLK_clk_p[1]}]
set_property PACKAGE_PIN D23 [get_ports {SI_FCLK_clk_p[2]}]
set_property PACKAGE_PIN G24 [get_ports {SI_FCLK_clk_p[0]}]
set_property IOSTANDARD LVDS_25 [get_ports {SI_FCLK_*}]

```

_i_pcie.xdc

```

#-----
# FEX0
set_property PACKAGE_PIN B20 [get_ports {PCI_PERSTN}]
set_property IOSTANDARD LVCMOS18 [get_ports {PCI_PERSTN}]
#-----
set_property PACKAGE_PIN K6 [get_ports {CLK_PCIE_100MHz_clk_p[0]}]
set_property PACKAGE_PIN N4 [get_ports {pcie_7x_mgt_rxp[2]}]
set_property PACKAGE_PIN R4 [get_ports {pcie_7x_mgt_rxp[3]}]
set_property PACKAGE_PIN L4 [get_ports {pcie_7x_mgt_rxp[1]}]
set_property PACKAGE_PIN J4 [get_ports {pcie_7x_mgt_rxp[0]}]

```

PCB REV01:

_i_io_ddr_clk.xdc

```
#-----  
#CLK DDR3  
#AC9 /AD9 for REV01  
#AB11 / AC11 for REV02  
set_property PACKAGE_PIN AC9 [get_ports CLK_DDR3_200MHz_clk_p]  
set_property PACKAGE_PIN AD9 [get_ports CLK_DDR3_200MHz_clk_n]  
set_property IOSTANDARD DIFF_SSTL15 [get_ports CLK_DDR3_200MHz_clk_p]  
set_property IOSTANDARD DIFF_SSTL15 [get_ports CLK_DDR3_200MHz_clk_n]
```

PCB REV02:

_i_io_ddr_clk.xdc

```
#-----  
#CLK DDR3  
#AC9 /AD9 for REV01  
#AB11 / AC11 for REV02  
set_property PACKAGE_PIN AB11 [get_ports CLK_DDR3_200MHz_clk_p]  
set_property PACKAGE_PIN AC11 [get_ports CLK_DDR3_200MHz_clk_n]  
set_property IOSTANDARD DIFF_SSTL15 [get_ports CLK_DDR3_200MHz_clk_p]  
set_property IOSTANDARD DIFF_SSTL15 [get_ports CLK_DDR3_200MHz_clk_n]
```

Software Design - SDK/HSI

For SDK project creation, follow instructions from:

[SDK Projects](#)

Application

Template location: ./sw_lib/sw_apps/

hello_tef1001

- Xiline Hello World as endless loop

SI5338_Init

- SI5338 I2C Configuration example only.

srec_spi_bootloader

- modified Xilinx SREC Bootloader, including SI5338 configuration
 - modified Files: blconfig.h, bootloader.c
 - add Files: si5338.h, si5338.c, register_map.h
 - modified xilisf_v5_11: xilisf.mld (default Flash Typ:5)

Additional Software

SI5338

File location <design name>/misc/SI5338/RegisterMap.txt

General documentation how you work with these project will be available on [SI5338](#)

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
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.	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]  Unknown macro: 'metadata'	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.	• typo correction part name • typo correction on programming chapter • note pcie

confluence.core. ContentEntityObject]		String, class com.atlassian. confluence.core. ContentEntityObject]	
25 Oct 2018	v.9	John Hartfiel	• add -410 assembly variant
25 Oct 2018	v.8	John Hartfiel	• 2018.2 release
07 Mar 2018	v.6	John Hartfiel	• 2017.4 release
2018-02-08	v.5	John Hartfiel	• 2017.2 release
2017-11-28	v.1	John Hartfiel	• initial release
--	all	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]
```

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.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]`