TEC0330 Test Board

Revision v.4

Exported on 2022-09-21

Online version of this document:

https://wiki.trenz-electronic.de/display/PD/TEC0330+Test+Board

1 Table of Contents

2 Table of Figures

3 Table of Tables

4 Overview

TEC0330 SI5338 Configuration, DDR Configuration and PCIe Core Example Design.

Refer to http://trenz.org/tec0330-info for the current online version of this manual and other available documentation.

4.1 Key Features

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

4.2 Revision History

Date	Vivado	Project Built	Authors	Description
2018-10-30	2018.2	TEC0330-test_board_noprebuilt-vivado_2018.2-build_03_20181030122205.zip TEC0330-test_board-vivado_2018.2-build_03_20181030122147.zip	John Hartfiel	• initial release

Table 1: Design Revision History

4.3 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	---

Table 2: Known Issues

4.4 Requirements

4.4.1 Software

Software	Version	Note
Vivado	2018.2	needed
SDK	2018.2	needed
SI5338 Clock Builder	---	optional

Table 3: Software

4.4.2 Hardware

Basic description of TE Board Part Files is available on TE Board Part Files.[1]

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
TEC0330-04-(330-2C)	330_2	REV04	DDR3 ECC SODIMM	32MB		• DDR configured for AW24P7228BLK0M (8GB)

Table 4: Hardware Modules

Design supports following carriers:

Carrier Model	Notes
PC with PCIe Card slot	Card need 3.3V from PCIe and 12V from ATX connector

Table 5: Hardware Carrier

Additional HW Requirements:

1 https://wiki.trenz-electronic.de/display/PD/TE+Board+Part+Files

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: • AW24P7228BLK0M (max. 8GB)

Table 6: Additional Hardware

4.5 Content

For general structure and of the reference design, see Project Delivery - Xilinx devices[2]

4.5.1 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

Table 7: Design sources

4.5.2 Additional Sources

Type	Location	Notes
SI5338	<design name>/misc/Si5338	SI5338 Project with current PLL Configuration

Table 8: Additional design sources

4.5.3 Prebuilt

2 https://wiki.trenz-electronic.de/display/PD/Project+Delivery+-+Xilinx+devices

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

Table 9: Prebuilt files (only on ZIP with prebult content)

4.5.4 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:

- TEC0330 "Test Board" Reference Design[3]

3 https://shop.trenz-electronic.de/Download/?path=Trenz_Electronic/PCIe_Cards/TEC0330/Reference_Design/2018.2/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.

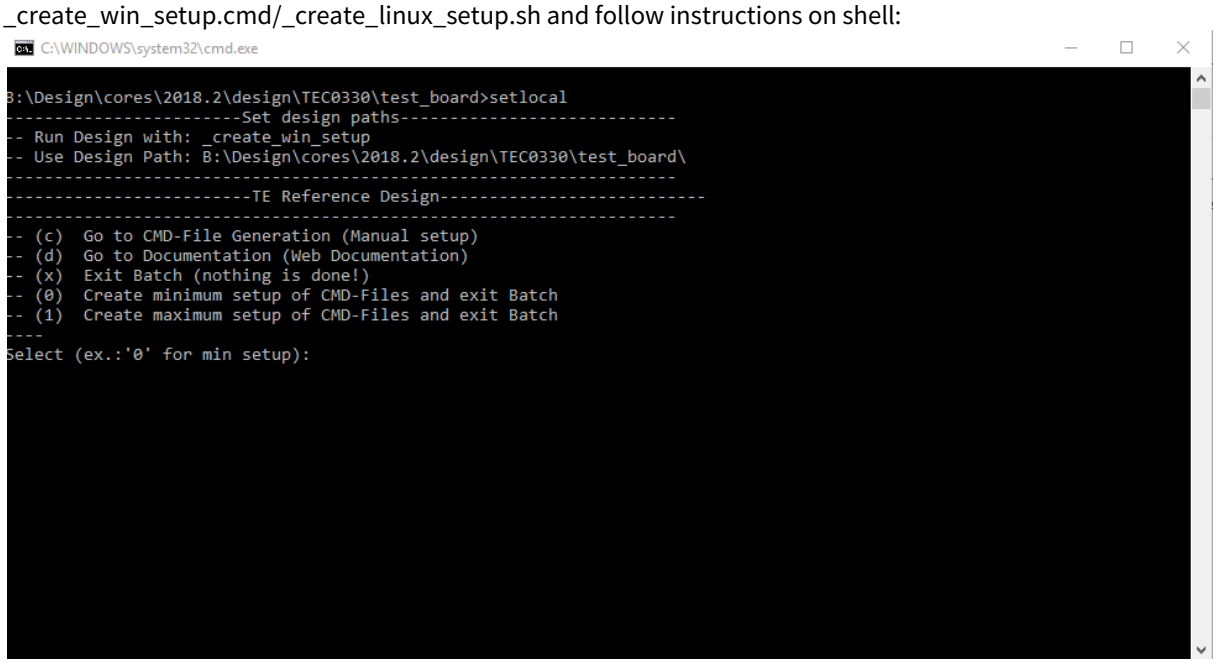
See also:

- Xilinx Development Tools[4]
- Vivado Projects - TE Reference Design[5]
- Project Delivery.[6]

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[7]

1. _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_guimode.cmd"
      Note: Select correct one, see TE Board Part Files[8]
5. Create HDF and export to prebuilt folder

4 https://wiki.trenz-electronic.de/display/PD/Xilinx+Development+Tools#XilinxDevelopmentTools-XilinxSoftware-BasicUserGuides
5 https://wiki.trenz-electronic.de/display/PD/Vivado+Projects+-+TE+Reference+Design
6 https://wiki.trenz-electronic.de/display/PD/Project+Delivery+-+Xilinx+devices
7 https://wiki.trenz-electronic.de/display/PD/Project+Delivery+-+Xilinx+devices#ProjectDeliveryXilinxdevices-Currentlylimitationsoffunctionality
8 https://wiki.trenz-electronic.de/display/PD/TE+Board+Part+Files

   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. 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[9]
   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.))
   c. (optional) for SI5338 reprogramming with MCS:
      i. Start SDK with Vivado GUI or start with TE Scripts on Vivado TCL: TE::sw_run_sdk to generate files manually
      ii. New Application with Project Name "SCU" and Processor "microblaze_mcs_0_microblaze_I", select TE Application "SCU-Firmware"
      iii. Create elf file
      iv. Copy "workspace\sdk\SCU\<release or debug>\SCU.elf" into "\firmware\microblaze_mcs_0" (replace shipped one) and regenerate design again (HW (Step5)+SW(Step6 only a.))

9 https://wiki.trenz-electronic.de/display/PD/SDK+Projects

6 Launch

6.1 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[10]

6.1.1 QSPI

1. Connect JTAG and Power ON PC
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_mcsfile -swapp hello_tec0330
4. Reboot PC

6.1.2 SD

Not supported.

6.1.3 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

6.2 Usage

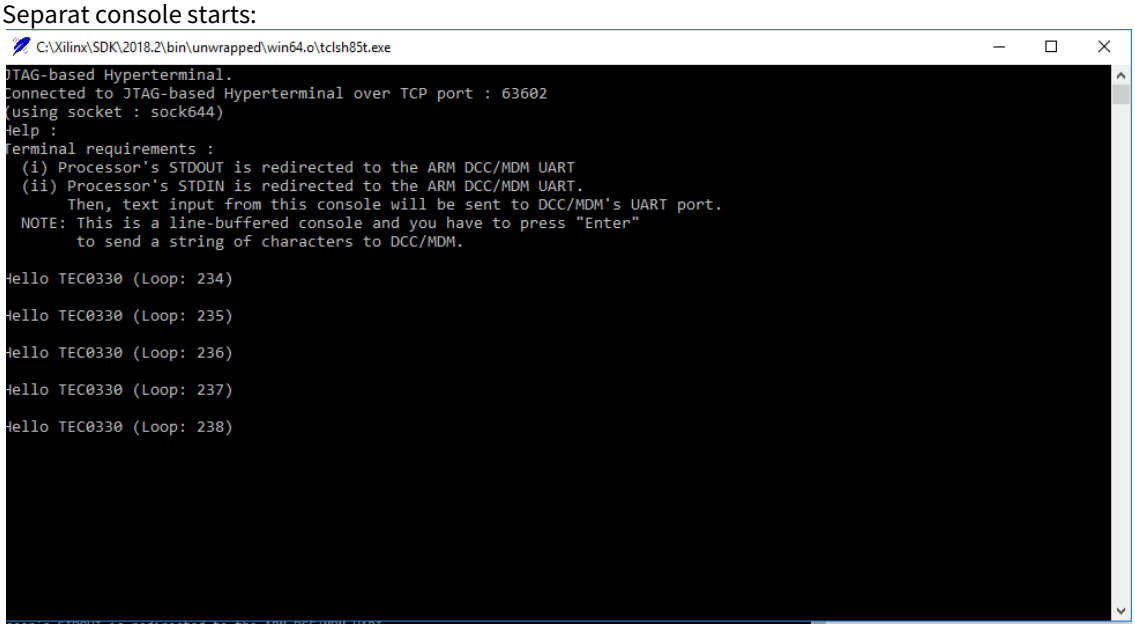
1. Prepare HW like described on section Programming(see page 12)
2. Power On PCB
   Note: 1. FPGA Load Bitfile into FPGA,MCS configure SI5338 and starts microblaze design, modified SREC Bootloader load application from QSPI into DDR (Depends on linker script)

6.2.1 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

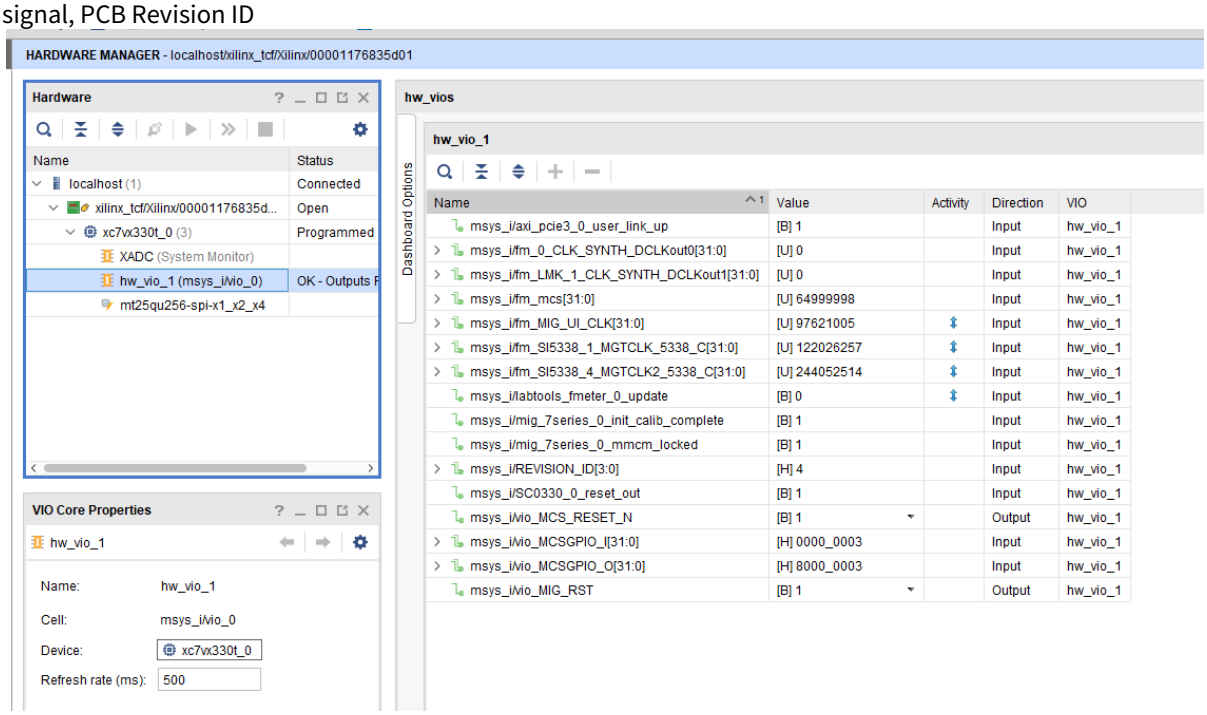
10 https://wiki.trenz-electronic.de/display/PD/Xilinx+Development+Tools#XilinxDevelopmentTools-XilinxSoftwareProgrammingandDebugging

- Separat console starts:



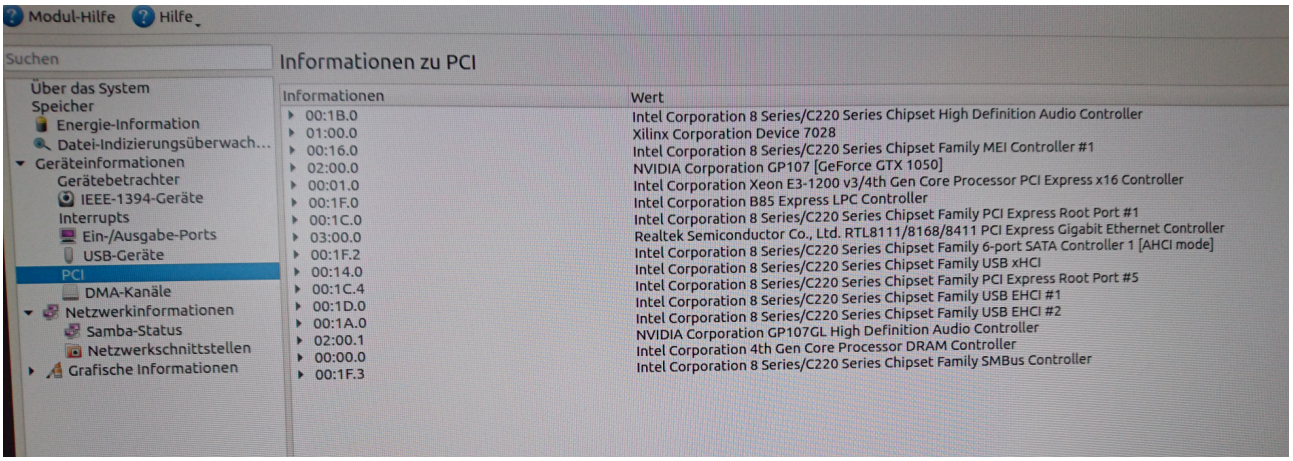
6.2.2 Vivado HW Manager:

1. Open Vivado HW Manager
2. Add VIO to Dashboard
3. Set Radix to unsigned integer for FMeterCLKs (fm_*). Note measurement is not accurate
4. Control:
   a. MCS Reset
   b. MIG Reset
5. Read: SI5338 CLKs (Unit Hz), PCIe Core User Link Up signal, MIG MMCM Lock signal, MIG Init Calibration Done signal, PCB Revision ID



6.2.3 PC:

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



7 System Design - Vivado

7.1 Block Design

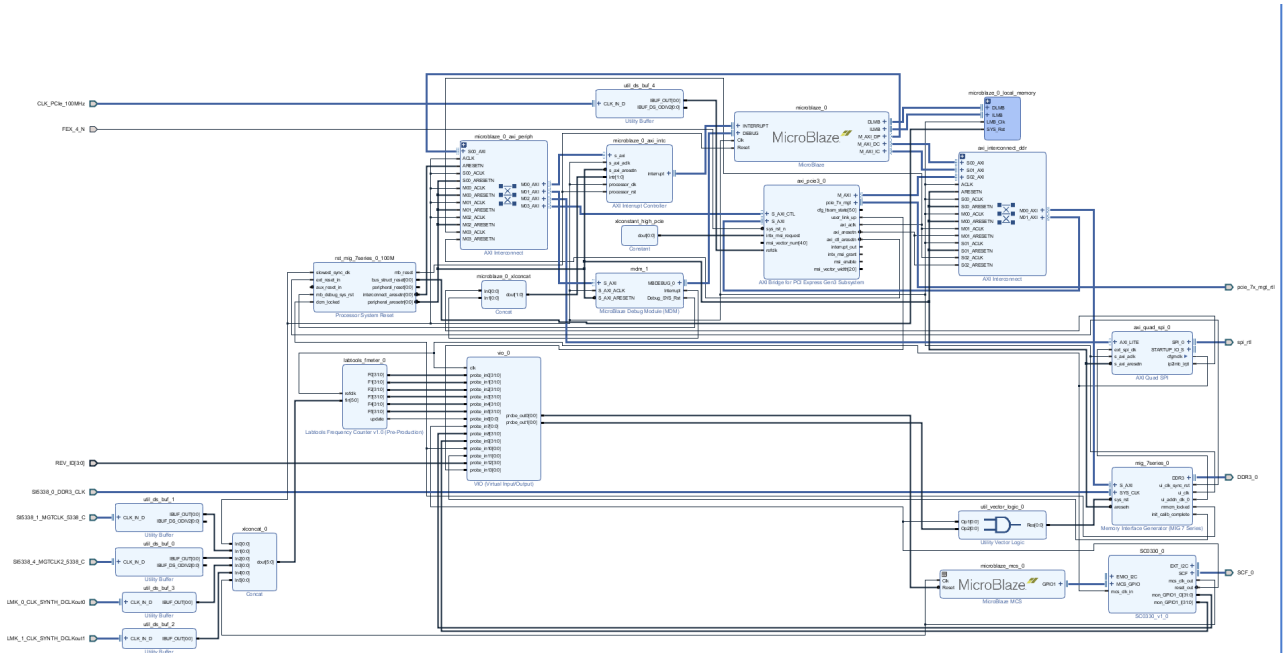


Figure 1: Block Design

7.2 Constrains

7.2.1 Basic module constrains

_i_bitgen_common.xdc

```
#
# Default common settings that do not depend assembly variant
#
set_property BITSTREAM.GENERAL.COMPRESS TRUE [current_design]
set_property BITSTREAM.CONFIG.CONFIGRATE 66 [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 PULLDOWN [current_design]
```

7.2.2 Design specific constrain

_i_io.xdc

```
#----------
#IIC to CPLD
set_property PACKAGE_PIN W29 [get_ports SCF_0_cpld_25_scl]
set_property PACKAGE_PIN W26 [get_ports SCF_0_cpld_19_oe]
set_property PACKAGE_PIN V29 [get_ports SCF_0_cpld_24_sda]
set_property IOSTANDARD LVCMOS18 [get_ports SCF_0_cpld_25_scl]
set_property IOSTANDARD LVCMOS18 [get_ports SCF_0_cpld_19_oe]
set_property IOSTANDARD LVCMOS18 [get_ports SCF_0_cpld_24_sda]
#----------
#PCIe
set_property PACKAGE_PIN E33 [get_ports FEX_4_N]
set_property IOSTANDARD LVCMOS18 [get_ports FEX_4_N]
set_property PACKAGE_PIN AD6 [get_ports {CLK_PCIe_100MHz_clk_p[0]}]
#todo check auto placement:
set_property CLOCK_DEDICATED_ROUTE FALSE [get_nets msys_i/axi_pcie3_0/inst/
pcie3_ip_i/inst/gt_top_i/pipe_wrapper_i/pipe_lane[0].gt_wrapper_i/pipe_txoutclk_out]
#----------
#Revision ID
set_property PACKAGE_PIN AP27 [get_ports {REV_ID[0]}]
set_property PACKAGE_PIN AN27 [get_ports {REV_ID[1]}]
set_property PACKAGE_PIN AP26 [get_ports {REV_ID[2]}]
set_property PACKAGE_PIN AP25 [get_ports {REV_ID[3]}]
set_property IOSTANDARD LVCMOS18 [get_ports {REV_ID[*]}]
#----------
#QSPI
set_property PACKAGE_PIN AL33 [get_ports {spi_rtl_ss_io[0]}]
set_property IOSTANDARD LVCMOS18 [get_ports {spi_rtl_ss_io[0]}]
set_property PACKAGE_PIN AN33 [get_ports spi_rtl_io0_io]
set_property PACKAGE_PIN AN34 [get_ports spi_rtl_io1_io]
set_property PACKAGE_PIN AK34 [get_ports spi_rtl_io2_io]
set_property PACKAGE_PIN AL34 [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]
#----------
#CLKS
##SI5338_0_DDR3_CLK #diff 1.5V AG17/AH17
set_property PACKAGE_PIN AG17 [get_ports {SI5338_0_DDR3_CLK_clk_p}]
set_property IOSTANDARD DIFF_SSTL15 [get_ports {SI5338_0_DDR3_CLK_clk_p}]
```

```
##SI5338_1_MGTCLK_5338_C #diff MGT 1.8V AB6/AB5
set_property PACKAGE_PIN AB6 [get_ports {SI5338_1_MGTCLK_5338_C_clk_p[0]}]
###SI5338_3_LMK_CLK #diff MGT 1.8V to LMK CLKin1
##SI5338_4_MGTCLK2_5338_C #diff MGT 1.8V H6/H5
set_property PACKAGE_PIN H6 [get_ports {SI5338_4_MGTCLK2_5338_C_clk_p[0]}]
##LMK_0_CLK_SYNTH_DCLKout0 #diff  1.8V AD29/AE29
set_property PACKAGE_PIN AD29 [get_ports {LMK_0_CLK_SYNTH_DCLKout0_clk_p[0]}]
set_property IOSTANDARD LVDS [get_ports {LMK_0_CLK_SYNTH_DCLKout0_clk_p[0]}]
set_property DIFF_TERM TRUE [get_ports {LMK_0_CLK_SYNTH_DCLKout0_clk_p[0]}]
##LMK_1_CLK_SYNTH_DCLKout1 #diff  1.8V AE31/AF31
set_property PACKAGE_PIN AE31 [get_ports {LMK_1_CLK_SYNTH_DCLKout1_clk_p[0]}]
set_property IOSTANDARD LVDS [get_ports {LMK_1_CLK_SYNTH_DCLKout1_clk_p[0]}]
set_property DIFF_TERM TRUE [get_ports {LMK_1_CLK_SYNTH_DCLKout1_clk_p[0]}]
###LMK_2_CLKIN_5338_P #diff  1.8Vto Si5338 IN1/IN2
###LMK_3_CLK_SYNTH_SDCLKout3 #diff  1.8Vto N.C.
###LMK_4_CLK_SYNTH_SDCLKout4 #diff MGT 1.8V T6/T5
###LMK_5_CLK_SYNTH_SDCLKout5 #diff  1.8Vto N.C.
###LMK_6_CLK_SYNTH_SDCLKout6 #diff  1.8Vto N.C.
###LMK_7_CLK_SYNTH_SDCLKout7 #diff MGT 1.8V F6/F5
###LMK_8_CLK_SYNTH_SDCLKout8 #diff  1.8Vto N.C.
###LMK_9_CLK_SYNTH_SDCLKout9 #diff  1.8Vto N.C.
###LMK_10_CLK_SYNTH_SDCLKout10 #diff  1.8Vto N.C.
###LMK_11_CLK_SYNTH_SDCLKout11 #diff  1.8Vto N.C.
###LMK_12_CLK_SYNTH_SDCLKout12 #diff  1.8Vto N.C.
###LMK_13_CLK_SYNTH_SDCLKout13 #diff  1.8Vto N.C.



#----------
```

8 Software Design - SDK/HSI

For SDK project creation, follow instructions from:

SDK Projects[11]

8.1 Application

Template location: ./sw_lib/sw_apps/

8.1.1 hello_tec0330

- Xiline Hello World as endless loop

8.1.2 scu

- Si5338 I2C Configuration via MCS.

srec_spi_bootloader

- modified Xilinx SREC Bootloade
  - modified Files: blconfig.h, bootloader.c
  - modified xilisf_v5_11: xilisf.mld (default Flash Typ:5)

11 https://wiki.trenz-electronic.de/display/PD/SDK+Projects

9 Additional Software

9.1 SI5338

File location <design name>/misc/Si5338/RegisterMap.txt

General documentation how you work with these project will be available on Si5338[12]

12 https://wiki.trenz-electronic.de/display/PD/Si5338

10 Appx. A: Change History and Legal Notices

10.1 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
2018-10-31	v.4(see page 6) Unbekanntes Makro: 'metadata'	John Hartfiel[13]	• 2018.2 release
--	all	John Hartfiel[14]	--

Table 10: 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

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

13 https://wiki.trenz-electronic.de/display/~j.hartfiel
14 https://wiki.trenz-electronic.de/display/~j.hartfiel

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[15]. 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[16] 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)[17].

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.

15 http://guidance.echa.europa.eu/
16 https://echa.europa.eu/candidate-list-table
17 http://www.echa.europa.eu/

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