Online version of this manual and other related documents can be found at https://wiki.trenz-electronic.de/display/PD/Trenz+Electronic+Documentation

1 Overview

Example show, how to reconfigure SI5338 with MCS and monitor CLK. Additional MicroBlaze is add for Hello TE0741 example.

1.1 Key Features

- MicroBlaze
- I2C
- UART
- Flash
- FMeter
- SI5338 initialisation with MCS

1.2 Revision History

Date	Vivado	Project Built	Authors	Description
2018-04-16	2017.4	TE0741-test_board-vivado_2017.4-build_07_20180416142156.zip TE0741-test_board_noprebuilt-vivado_2017.4-build_07_20180416142217.zip	John Hartfiel	• initial release

1.3 Release Notes and Know Issues

Issues	Description	Workaround	To be fixed version
---	---	---	---

1.4 Requirements

1.4.1 Software

Software	Version	Note
Vivado	2017.4	needed
SDK	2017.4	needed

1.4.2 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
TE0741-03-070-2IF	070_2if	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-160-2IF	160_2if	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-325-2IF	325_2if	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-410-2IF	410_2if	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-070-2CF	070_2cf	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-160-2CF	160_2cf	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-325-2CF	325_2cf	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	
TE0741-03-410-2CF	410_2cf	REV02, REV03	---	32MB	MGT LR: 6,6 Gb/s	

Module Model	Board Part Short Name	PCB Revision Support	DDR	QSPI Flash	Others	Notes
TE0741-03-160-2C1	160_2c1	REV02, REV03	---	32MB	MGT LR: 10,3125 Gb/s	

Design supports following carriers:

Carrier Model	Notes
TE0701	
TE0703	
TE0705	
TE0706	
TEBA0841	used as reference carrier

Additional HW Requirements:

Additional Hardware	Notes
USB Cable for JTAG/UART	Check Carrier Board and Programmer for correct type
XMOD Programmer	Carrier Board dependent, only if carrier has no own FTDI

1.5 Content

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

1.5.1 Design Sources

Type	Location	Notes
Vivado	<design name>/block_design <design name>/constraints <design name>/ip_lib <design name>/firmware	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

1.5.2 Additional Sources

Type	Location	Notes
SI5338 Project	\misc\SI5338	

1.5.3 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

1.5.4 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:

- TE0741 "Test Board" Reference Design

2 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.

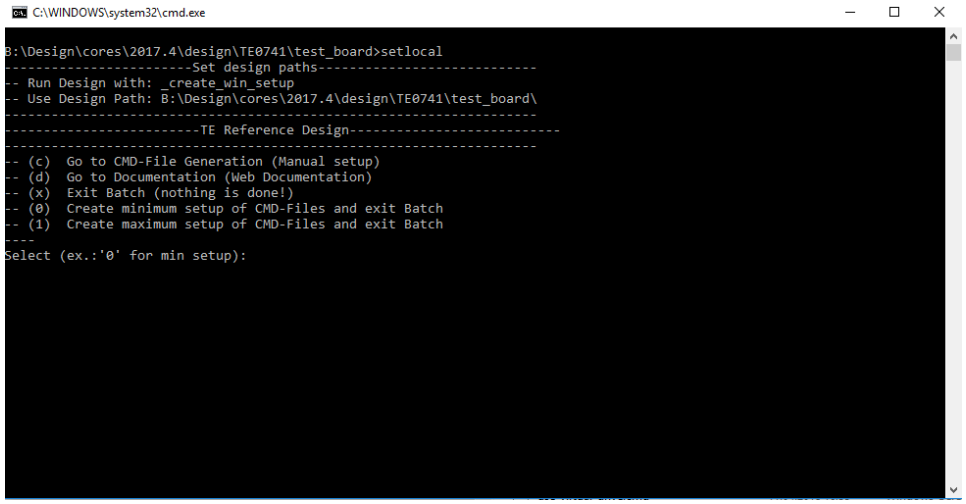
See also:Xilinx Development Tools

- Xilinx Development Tools
- 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:



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
5. Create HDF and export to prebuilt folder
    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 MCS Firmware (optional):
    a. Create SDK Project with TE Scripts on Vivado TCL: TE::sw_run_sdk
    b. Create "SCU" application
    Note: Select MCS Microblaze and SCU Application
    c. Select Release Built
    d. Regenerate App
7. Generate Programming Files with HSI/SDK

   a. Run on Vivado TCL: TE::sw_run_hsi
      Note: Scripts generate applications and bootable files, which are defined in "sw_lib\apps_list.csv"
   b. (alternative) Start SDK with Vivado GUI or start with TE Scripts on Vivado TCL: TE::sw_run_sdk
      Note: See SDK Projects
8. Copy "\prebuilt\software\<short name>\hello_te0741.elf" into "\firmware\microblaze_0\"
9. (optional) Copy "\\workspace\sdk\scu\Release\scu.elf" into "\firmware\microblaze_mcs_0\"
10. Regenerate Vivado Project or Update Bitfile only with "hello_te0741.elf" and "scu.elf"

3 Launch

3.1 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

3.1.1 QSPI

1. Connect JTAG and power on PCB
2. (if not done) Select correct device and Xilinx install path on "design_basic_settings.cmd" and create Vivado project with "vivado_create_project_guimode.cmd" or open with "vivado_open_project_guimode.cmd", if generated.
3. Type on Vivado Console: TE::pr_program_flash_mcsfile -swapp u-boot
   Note: Alternative use SDK or setup Flash on Vivado manually
4. Reboot (if not done automatically)

3.1.2 SD

Not used on this Example.

3.1.3 JTAG

1. Connect JTAG and power on PCB
2. Open Vivado HW Manager
3. Program FPGA with Bitfile from "prebuilt\hardware\<short dir>"

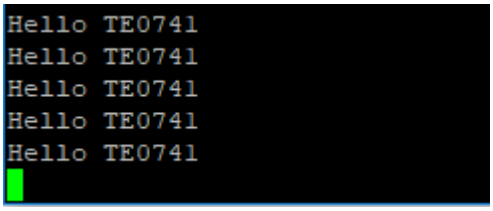
3.2 Usage

1. Prepare HW like described on section Programming
2. Connect UART USB (most cases same as JTAG)
3. Power on PCB
   Note: FPGA Loads Bitfile from Flash,MCS Firmware configure SI5338 and starts Microblaze, Hello TE0741 from Bitfile Example will be run on UART console.
   Do not reboot, if Bitfile programming over JTAG is used as programming method.

3.2.1 UART

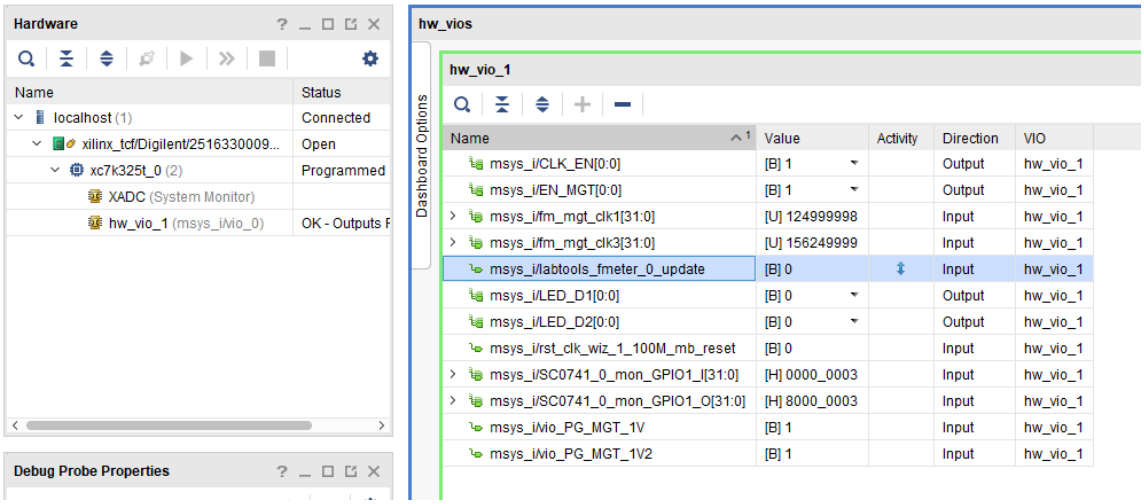
Open Serial Console (e.g. putty)

1. Speed: 9600
2. COM Port: Win OS, see device manager, Linux OS see dmesg |grep tty (UART is *USB1)



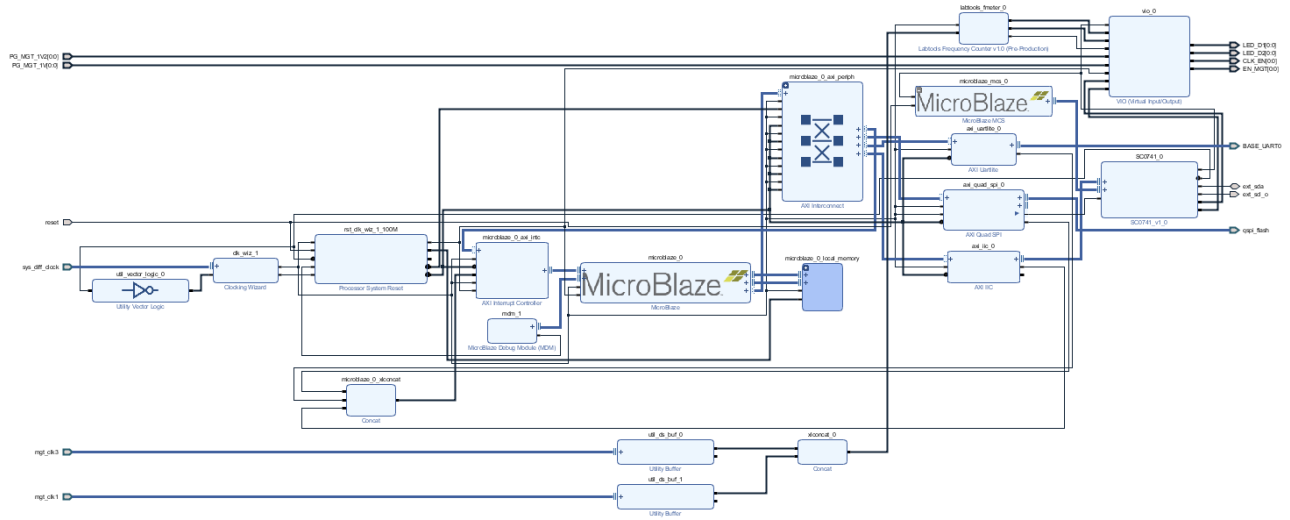
3.2.2 Vivado HW Manager:

1. Open Vivado HW-Manager and add VIO signal to dashboard (*.ltx located on prebuilt folder).
   a. Set radix from VIO signals (MGT...) to unsigned integer.
      Note: Frequency Counter is inaccurate and displayed unit is Hz
   b. MGT REFCL1~125MHz, GT_REFCLK3~156,25MHz (default off, configured with MCS Firmware)
   c. LED_D1/D2 control
   d. SI5338 25MHz REF CLK Enable
   e. MGT Power Monitoring+MGT Enable



4 System Design - Vivado

4.1 Block Design



4.2 Constrains

4.2.1 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 3.3 [current_design]
set_property CFGBVS VCCO [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]
```

4.2.2 Design specific constrain

_i_io.xdc

```
#LED
set_property PACKAGE_PIN D26 [get_ports {LED_D1[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {LED_D1[0]}]
set_property PACKAGE_PIN E26 [get_ports {LED_D2[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {LED_D2[0]}]
#MGT Power
set_property PACKAGE_PIN G25 [get_ports {PG_MGT_1V2[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {PG_MGT_1V2[0]}]
set_property PACKAGE_PIN K23 [get_ports {PG_MGT_1V[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {PG_MGT_1V[0]}]
set_property PACKAGE_PIN H22 [get_ports {EN_MGT[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {EN_MGT[0]}]
#SI5338 CLK
set_property PACKAGE_PIN C26 [get_ports {CLK_EN[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {CLK_EN[0]}]
#I2C PLL SI5338
set_property PACKAGE_PIN A20 [get_ports ext_scl_o]
set_property IOSTANDARD LVCMOS33 [get_ports ext_scl_o]
set_property PACKAGE_PIN B21 [get_ports ext_sda]
set_property IOSTANDARD LVCMOS33 [get_ports ext_sda]
```

_i_timing.xdc

```
#Fmeter can be ignored, it's only simple measurement

set_false_path -from [get_pins {msys_i/labtools_fmeter_0/U0/FMETER_gen[*].COUNTER_F_inst/bl.DSP48E_2/CLK}] -to [get_pins {msys_i/labtools_fmeter_0/U0/F_reg[*]/D}]

set_false_path -from [get_pins msys_i/labtools_fmeter_0/U0/toggle_reg/C] -to [get_pins {msys_i/labtools_fmeter_0/U0/FMETER_gen[*].COUNTER_F_inst/bl.DSP48E_2/RSTC}]
set_false_path -from [get_pins msys_i/labtools_fmeter_0/U0/toggle_reg/C] -to [get_pins {msys_i/labtools_fmeter_0/U0/FMETER_gen[*].COUNTER_F_inst/bl.DSP48E_2/RSTA}]
set_false_path -from [get_pins msys_i/labtools_fmeter_0/U0/toggle_reg/C] -to [get_pins {msys_i/labtools_fmeter_0/U0/FMETER_gen[*].COUNTER_F_inst/bl.DSP48E_2/RSTB}]
set_false_path -from [get_pins msys_i/labtools_fmeter_0/U0/toggle_reg/C] -to [get_pins {msys_i/labtools_fmeter_0/U0/FMETER_gen[*].COUNTER_F_inst/bl.DSP48E_2/CEALUMODE}]
set_false_path -from [get_pins msys_i/labtools_fmeter_0/U0/toggle_reg/C] -to [get_pins {msys_i/labtools_fmeter_0/U0/FMETER_gen[*].COUNTER_F_inst/bl.DSP48E_2/RSTCTRL}]


set_false_path -from [get_clocks -of_objects [get_pins msys_i/clk_wiz_1/inst/mmcm_adv_inst/CLKOUT0]] -to [get_clocks {msys_i/util_ds_buf_0/U0/IBUF_OUT[0]}]
```

```
13  set_false_path -from [get_clocks -of_objects [get_pins msys_i/clk_wiz_1/
    inst/mmcm_adv_inst/CLKOUT0]] -to [get_clocks {msys_i/util_ds_buf_1/U0/
    IBUF_OUT[0]}]
```

5 Software Design - SDK/HSI

For SDK project creation, follow instructions from:

SDK Projects

5.1 Application

5.1.1 SCU

MCS Firmware to configure SI5338 and Reset System.

Template location: \sw_lib\sw_apps\scu

5.1.2 Hello TE0741

Xilinx Hello World example as andless loop

Template location: \sw_lib\sw_apps\hello_te0741

6 Additional Software

6.1 SI5338

Download ClockBuilder Desktop for SI5338

1. Install and start ClockBuilder
2. Select SI5338
3. Options → Open register map file
   Note: File location <design name>/misc/Si5338/RegisterMap.txt
4. Modify settings
5. Options → save C code header files
6. Replace Header files from SCU template with generated file

7 Appx. A: Change History and Legal Notices

7.1 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
2018-05-15	v.7	@ John Hartfiel	• Release 2017.4 • small description update
2018-04-16	v.1	@ John Hartfiel	• Initial release
	All	@ John Hartfiel , Waldemar Hanemann	

7.2 Legal Notices

7.3 Data Privacy

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

7.4 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.

7.5 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.

7.6 Copyright Notice

7.7 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.

7.8 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.

7.9 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. 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.

2019-06-07