1 Overview

MicroBlaze Design with Linux example.

Refer to http://trenz.org/te0xyz-info for the current online version of this manual and other available documentation.

1.1 Key Features

- Vitis/Vivado 2019.2
- PetaLinux
- MIG
- FLASH

1.2 Revision History

Date	Vivado	Project Built	Authors	Description
2020-07-09	2019.2	TE0713-test_board_noprebuilt-vivado_2019.2-build_13_20200709071700.zip TE0713-test_board-vivado_2019.2-build_13_20200709071613.zip	John Hartfiel	• initial release

Table 1: Design Revision History

1.3 Release Notes and Know Issues

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

Table 2: Known Issues

1.4 Requirements

1.4.1 Software

Software	Version	Note
Vitis	2019.2	needed, Vivado is included into Vitis installation
PetaLinux	2019.2	needed

Table 3: Software

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	EMMC	Others	Notes
TE0713-02-100-2c	100_2c	REV02|REV01	1GB	32MB	NA	NA	NA
TE0713-02-200-2c	200_2c	REV02|REV01	1GB	32MB	NA	NA	NA

Table 4: Hardware Modules
Design supports following carriers:

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

Table 5: Hardware Carrier
Additional HW Requirements:

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

Table 6: Additional Hardware

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	Vivado Project will be generated by TE Scripts
Vitis	<design name>/sw_lib	Additional Software Template for Vitis and apps_list.csv with settings automatically for Vitis app generation
PetaLinux	<design name>/os/petalinux	PetaLinux template with current configuration

Table 7: Design sources

1.5.2 Additional Sources

Type	Location	Notes
--	--	--

Table 8: Additional design sources

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

File	File-Extension	Description
Hardware-Platform-Specification-Files	*.xsa	Exported Vivado Hardware Specification for Vitis 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)
OS-Image	*.ub	Image with Linux Kernel (On Petalinux optional with Devicetree and RAM-Disk)
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)

1.5.4 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:

- TE0713 "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.

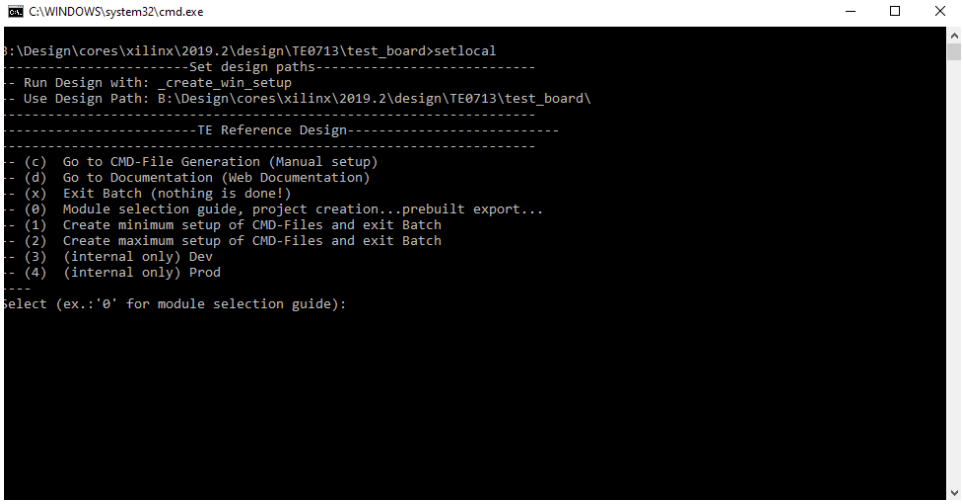
See also:

- Xilinx Development Tools
- 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 to start "Module Selection Guide"
3. (optional Win OS) Generate Virtual Drive or use short directory for the reference design (for example x: \<design name>)
4. Create Project (follow instruction of the product selection guide), settings file will be configured automatically during this process)
    a. (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 TE Board Part Files
5. Create XSA 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. Create Linux (uboot.elf and image.ub) with exported XSA
    a. XSA is exported to "prebuilt\hardware\<short name>"
    Note: HW Export from Vivado GUI create another path as default workspace.
    Create Linux images on VM, see PetaLinux KICKstart
        i. Use TE Template from /os/petalinux
7. Add Linux files (uboot.elf and image.ub) to prebuilt folder
    a. "prebuilt\os\petalinux\<ddr size>" or "prebuilt\os\petalinux\<short name>"
8. Generate Programming Files with Vitis
    a. Run on Vivado TCL: TE::sw_run_vitis -all
    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_vitis
    Note: TCL scripts generate also platform project, this must be done manuelly in case GUI is used. See Vitis

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 Get prebuilt boot binaries

1. _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 foler>/_binaries_<Artikel Name>) with subfolder (boot_<app name>) for different applications will be generated

3.1.2 QSPI

Optional for Boot.bin on QSPI Flash and image.ub on SD.

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"
3. Type on Vivado TCL Console: TE::pr_program_flash -swapp u-boot
   Note: To program with SDK/Vivado GUI, use special FSBL (zynqmp_fsbl_flash) on setup
   optional "TE::pr_program_flash -swapp hello_te0820" possible
4. Copy image.ub on SD-Card
    - use files from (<project foler>/_binaries_<Articel Name>)/boot_linux from generated binary folder,see: Get prebuilt boot binaries
    - or use prebuilt file location, see <design_name>/prebuilt/readme_file_location.txt
5. Insert SD-Card

3.1.3 SD

Not used on this Example.

3.1.4 JTAG

Not used on this Example.

3.2 Usage

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

3. Select SD Card as Boot Mode (or QSPI - depending on step 1)
   Note: See TRM of the Carrier, which is used.
4. Power On PCB
   Note: FPGA Loads Bitfile from Flash, SREC Bootloader from Bitfile Firmware loads U-Boot into DDR (This takes a while), U-boot loads Linux from QSPI Flash into DDR

Boot process takes a while, please wait.

```
SREC SPI Bootloader (TE modified): Start initialization

SREC SPI Bootloader (TE modified): SPI driver Init passed

SREC SPI Bootloader (TE modified): Serial Flash Library Init passed

SREC SPI Bootloader (TE modified): Load Image
Loading SREC image from flash @ address: 005e0000
Please wait...
```

3.2.1 Linux

1. Open Serial Console (e.g. putty)
   a. Speed: 9600
   b. COM Port: Win OS, see device manager, Linux OS see dmesg |grep tty (UART is *USB1)
2. Linux Console:
   Note: Wait until Linux boot finished For Linux Login use:
   a. User Name: root
   b. Password: root

3.2.2 Vivado HW Manager

Open Vivado HW-Manager and add VIO signal to dashboard (*.ltx located on prebuilt folder)

- Monitoring:
  - MIG Calibration Done
  - Main Reset
  - MicroBlaze Reset

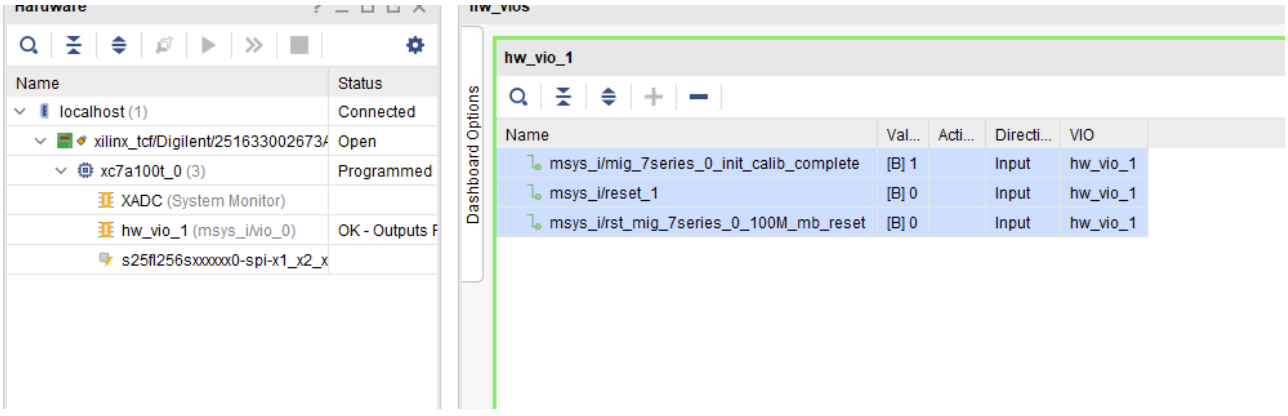


Figure 1: Vivado Hardware Manager

4 System Design - Vivado

4.1 Block Design

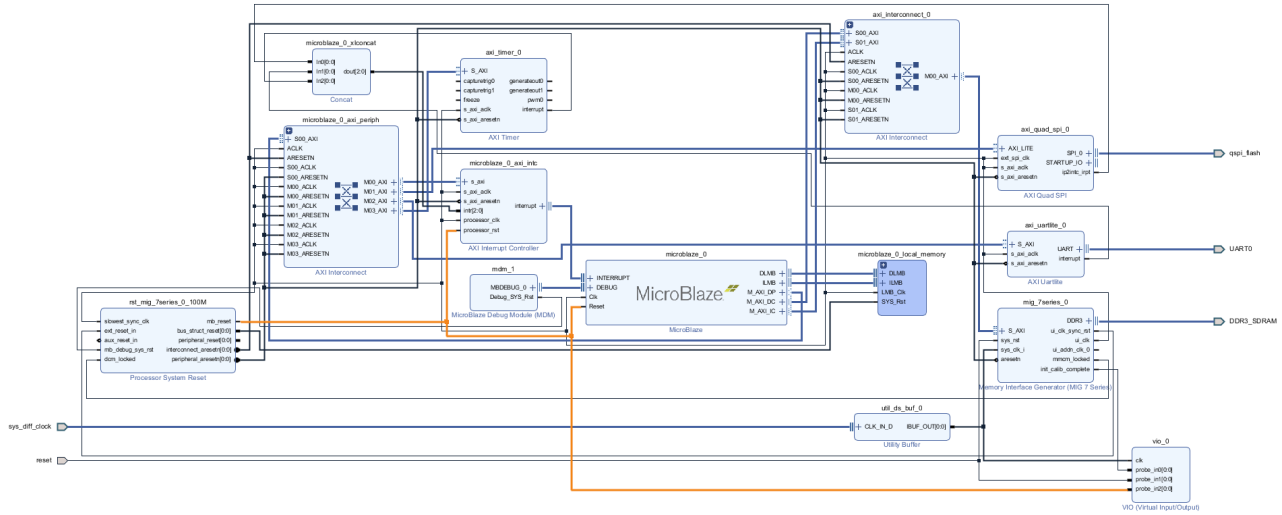


Figure 2: Block Design

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]
```

_i_bitgen.xdc

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

```
#
```

4.2.2 Design specific constrain

5 Software Design - Vitis

For SDK project creation, follow instructions from:

Vitis

5.1 Application

Template location: ./sw_lib/sw_apps/

5.1.1 srec_spi_bootloader

TE modified 2019.2 SREC

Bootloader to load app or second bootloader from flash into DDR

Descriptions:

- Modified Files: blconfig.h, bootloader.c
- Changes:
  - Add some console outputs and changed bootloader read address.
  - Add bugfix for 2018.2 qspi flash (some reinitialisation)

5.1.2 xilisf_v5_14

TE modified 2019.2 xilisf_v5_14

- Changed default Flash type to 5.

5.1.3 hello_te0712

Hello TE0712 is a Xilinx Hello World example as endless loop instead of one console output.

5.1.4 u-boot

U-Boot.elf is generated with PetaLinux. SDK/HSI is used to generate u-boot.srec. Vivado to generate *.mcs

6 Software Design - PetaLinux

For PetaLinux installation and project creation, follow instructions from:

- PetaLinux KICKstart

6.1 Config

Start with **petalinux-config** or **petalinux-config --get-hw-description**

Changes:

- SUBSYSTEM_FLASH_AXI_QUAD_SPI_0_BANKLESS_PART0_SIZE = 0x5E0000
- SUBSYSTEM_FLASH_AXI_QUAD_SPI_0_BANKLESS_PART1_SIZE = 0x300000
- SUBSYSTEM_FLASH_AXI_QUAD_SPI_0_BANKLESS_PART2_SIZE = 0x20000
- SUBSYSTEM_FLASH_AXI_QUAD_SPI_0_BANKLESS_PART3_SIZE = 0xA00000
    - (Set kernel flash Address to 0x900000 and Kernel size to 0xA00000)

6.2 U-Boot

Start with **petalinux-config -c u-boot**

Changes:

- CONFIG_ENV_IS_NOWHERE=y
- # CONFIG_ENV_IS_IN_SPI_FLASH is not set

Change platform-top.h:

```
```

6.3 Device Tree

```
/include/ "system-conf.dtsi"
/ {
};
```

6.4 Kernel

Start with **petalinux-config -c kernel**

Changes:

- No changes.

6.5 Rootfs

Start with **petalinux-config -c rootfs**

Changes:

- No changes.

6.6 Applications

No additional application.

7 Additional Software

No additional software is needed.

8 Appx. A: Change History and Legal Notices

8.1 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
2020-07-09	v.1	John Hartfiel	• 2019.2 initial release
--	all	John Hartfiel	--

Table 10: Document change history.

8.2 Legal Notices

8.3 Data Privacy

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

8.4 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.

8.5 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.

8.6 Copyright Notice

8.7 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.

8.8 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.

8.9 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