Application Note

Akademie věd České republiky
Ústav teorie informace a automatizace AV ČR, v.v.i.

Trenz TEBF0808 + TE0808-04-6EB21A SoM I²C Communication with NUCLEO STM32H753

Lukas Kohout
kohoutl@utia.cas.cz

Revision history

Rev.	Date	Author	Description
0	22. 03. 2021	L. Kohout	Initial draft
1			
2			

Content

Acknowledgement

This work has been supported from project Arrowhead Tools, project number ECSEL 826452 and corresponding Czech NFA (MSMT) institutional support project 8A19009.

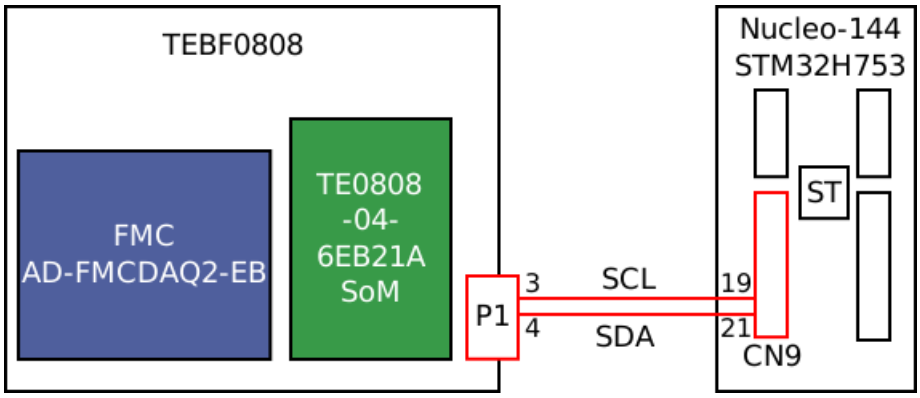

Figure 2: Simplified block diagram of the demonstrator.

1 Introduction

This document is a quick guide describing steps to establish a basic I^{2}C communication between Trenz Electronic TE0808 HW platform (TEBF0808 carrier board [1] with Trenz Electronic TE0808-04-6EB21A System on Module [2]) and ST NUCLEO-144 board with STM32H753 device. The procedure has been tested on PC running Ubuntu 16.04 LTS.

2 Description

A simplified block diagram of the demonstrator is shown in Figure 2. TE0808 HW platform acts as I^{2}C master. It uses built-in I^{2}C core in PS part of the Zynq UltraScale device. The I^{2}C lines are provided via P1 connector on the TE0808 platform. The data line SDA is available via pin 4 or pin 10 of the P1 connector. The clock line SCL is available via pin 3 or pin 9 of the P1 connector. These lines include external PULL-UP resistors tight to 3.3 V power supply on the TE0808 HW platform. The communication speed is set to 400 kHz. A software application of the TE0808 allows user to make read or write operations. These operations simulate reading or writing some register space on the slave device. It is assumed that the slave has up to 256 registers and each register stores one byte. For instance, write operation

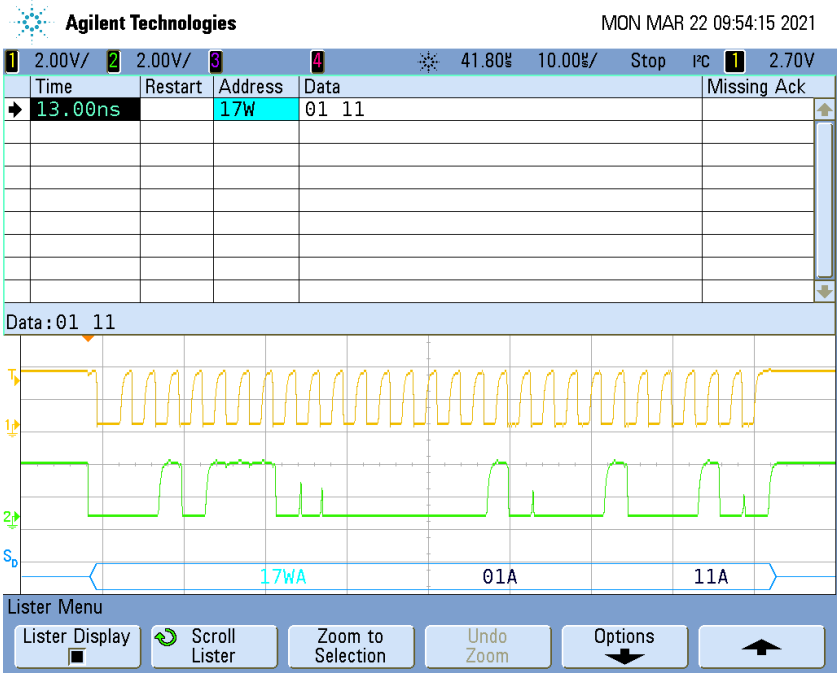

Figure 1: Oscilloscope snapshot - write operation.

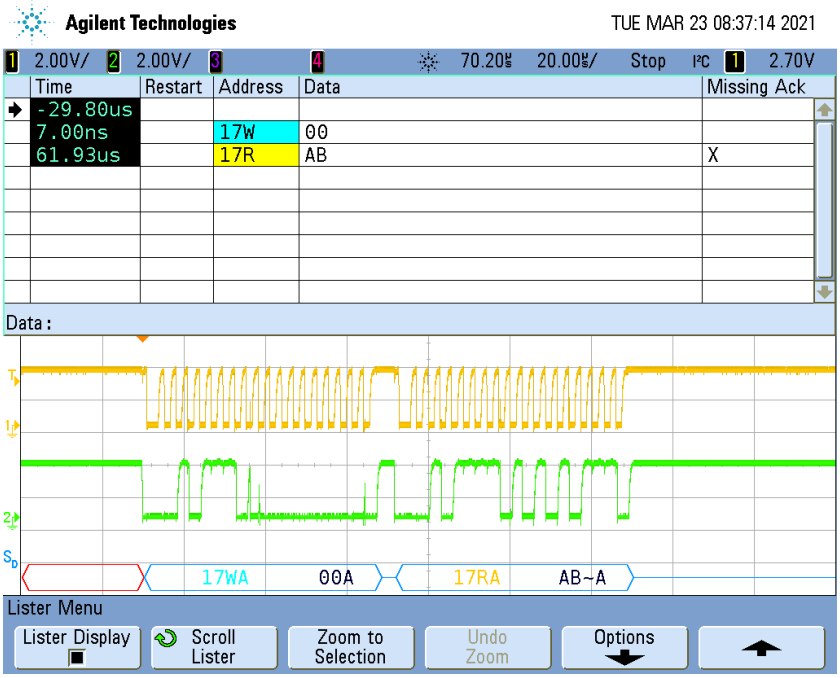

Figure 3: Oscilloscope snapshot - read operation

means sending of two bytes, where first byte is register address and second byte is the value to be written. Figure 1 shows an oscilloscope snapshot of the I^{2}C write operation of the register number 0x01 with value 0x11 on the slave device with address 0x17 (7-bit). Read operation includes sending of the register address to the slave and receiving the value from the slave device. Figure 3 shows an oscilloscope snapshot of the I^{2}C read operation of the register number 0x00 on the slave device which has address 0x17 (7-bit). The slave returns value 0xAB.

The NUCLEO board is in role of the slave device on the I^{2}C bus. Its address is 0x17 in 7-bit representation. The data line SDA is available via pin 21 the CN9 connector (port PF0). The clock line SCL is available via pin 19 of the CN9 connector (port PF1). The SW application implements a testing register file of 7 eight-bit registers. The structure of the register file is described in Table 1.

Table 1: Register file description.

Number	R/W	Default value	Description
0x00	R	0xAB	ID
0x01	W	0x00	Configuration
0x02	R	0x00	Status
0x03	W	0x00	Test 0
0x04	W	0x02	Test 1
0x05	W	0x04	Test 2
0x06	W	0x08	Test 3

3 Used tools and Resources

- TE0808 HW platform
  - Trenz Electronic TEBF0808 carrier board [1].
  - Trenz Electronic TE0808-04-6EB21A SoM. It contains Xilinx Zynq UltraScale+ device with 4GB DDR4 memory [2].
  - Mini USB cable.
  - Xilinx Software Development Kit 2018.2, it a part of Xilinx Vivado Design Suite 2018.2 (Web Pack edition is sufficient). To download the tool go to Xilinx web page: https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vivado-design-tools/archive.html.
- ST NUCLEO-144 board
  - ST NUCLEO-144 board with STM32H753 device [3].
  - Micro USB cable
  - System Workbench for STM32, free IDE on Windows, Linux and OS X. To download the tool, follow steps described on this web page: https://www.openstm32.org/HomePage.
  - STM32Cube MCU Package for STM32H7 series. This package contains HAL, Low-Layer APIs and CMSIS, USB, TCP/IP, File system, RTOS, Graphic and examples running on ST boards with STM32H7 device. Download link: https://www.st.com/content/st_com/en/products/embedded-software/mcu-mpu-embedded-software/stm32-embedded-software/stm32cube-mcu-mpu-packages/stm32cubeh7.html.

4 Demonstrator Startup

This section is separated onto two sub-sections. The first sub-section describes startup procedure for Trenz TE0808 HW platform, the second one covers steps for NUCLEO board.

4.1 TE0808 HW Platform

The TE0808 HW platform is configured to be I2C master. To run this part of the demonstrator follow the steps bellow.

1. Unpack the package attached with this application note. The package content is listed in Section 5.
2. Start Xilinx SDK 2018.2, set workspace path to *te0808-6EG-1EE/sdk*.
3. Create new HW project, menu
   File->New->Project...
   Xilinx->Hardware Platform Specification
   Project name: *hw*
   Browse to file *te0808-6EG-1EE/prebuilt/system.hdf*
4. Create Board Support Package, menu
   File->New->Board Support Package
   Project name: *bsp*
   Hardware Platform: *hw*
   CPU: *psu_cortex53_0*
   Compiler: *64-bit*
   Board Support Package OS: *standalone*
5. Import SW project, menu
   File->Import

General->Existing Projects into Workspace

Click *Browse* button

Click *OK* button

Select project: i2c

Click *Finish*

6. Compile the project, menu *Project->Build All*.
7. As this demonstrator has been developed from the demonstrator for Xilinx XCU102 board, initial configuration of the Zynq UltraScale+ has to be modified. Patch *psu_init.tcl* file:

   Replace file *te0808-6EG-1EE/sdk/i2c/hw/psu_init.tcl*

   with *te0808-6EG-1EE/prebuilt/psu_init.tcl* file.

   It is possible that this step will be required after each Xilinx SDK startup, because the file is auto-generated.
8. Configure TEBF0808 carrier board
   - Set *S4* to "off off off on"
   - Set *S5* to "on on on on"
   - All other switches and jumpers let in their defaults.
9. Connect mini USB cable to JTAG, XMOD1 module on the TEBF0808 carrier board.
10. Connect I^2C lines, TE0808 HW platform provides I^2C on connector P1
    - SDA – P1 pin 4 or pin 10
    - SCL – P1 pin 3 or pin 9

    NUCLEO board provides I^2C on connector CN9
    - SDA – CN9 pin 21 (port PF0)
    - SCL – CN9 pin 19 (port PF1)
11. Power the board on. Push button S1 and then push button S2.
12. Start serial terminal, putty for instance. The settings are:
    - Baud rate – 115200
    - Data bits – 8
    - Stop bits – 1
    - Parity – none
    - Flow control – none

    Download bitstream to the board, in Xilinx SDK *Xilinx->Program FPGA*
13. Download SW to the board, in *Project Explorer* select *i2c*, right click on it:

    Run As->Launch on Hardware (System Debugger)
14. Observe the serial terminal:

```
Xilinx Zynq MP First Stage Boot Loader
Release 2018.2   Mar  5 2021  -  10:35:23
PMU-FW is not running, certain applications may not be supported.


******************************************************************************
*  Signal Processing Dept., UTIA AV CR, v.v.i.                               *
*  I2C_PS - TE0808 PMOD1 Example                                             *
******************************************************************************

Initialize I2C_PS0 ...OK
TCA9548 Select Port 0 - PMOD ... OK
```

```
Menu:
-----------------------------------------------
w - I2C_PS Write register
W - I2C_PS Write register and readback
r - I2C_PS Read register
m - Print this menu
x - Quit
-----------------------------------------------
```

4.2 NUCLEO Board

The NUCLEO board with STM32H753 device is configured to be a slave on I2C bus. To run this part of the demonstrator follow the steps bellow.

1. Unpack the package attached with this application note. The package content is listed in Section 5.
2. Download and unpack STM32Cube MCU Package for STM32H7 series. The name of the package folder should be *STM32Cube_FW_H7_V1.9.0*. Current version is 1.9.0 (03. 22. 2021). Download link:
   https://www.st.com/content/st_com/en/products/embedded-software/mcu-mpu-embedded-software/stm32-embedded-software/stm32cube-mcu-mpu-packages/stm32cubeh7.html.
3. Update the STM32Cube package with the demonstrator project. Copy content of the *STM32Cube_FW_H7_V1.9.0* folder of the demonstrator package to the *STM32Cube_FW_H7_V1.9.0* folder of the STM32Cube package.
4. Start System Workbench for STM32, set workspace path to demonstrator project in the STM32Cube package, folder *STM32Cube_FW_H7_V1.9.0/Projects/NUCLEO-H753ZI/Examples/I2C/I2C_TwoBoards_ComIT/SW4STM32*.
5. Import project into workspace, menu
   File->Import
   General->Existing Projects into Workspace
   Click *Browse* button
   Click *OK* button
   Select project: *STM32H753ZI-Nucleo*
   Click *Finish*
6. Compile the project, menu *Project->Build All*.
7. Connect I^2C lines, NUCLEO board provides I^2C on connector CN9
   - SDA – CN9 pin 21 (port PF0)
   - SCL – CN9 pin 19 (port PF1)

   TE0808 HW platform provides I^2C on connector P1
   - SDA – P1 pin 4 or pin 10
   - SCL – P1 pin 3 or pin 9
8. Connect micro USB cable to CN1 connector on the NUCLEO board. This cable provides programming interface and power supply of the board.
9. Download SW to the board, in *Project Explorer* select *STM32H753ZI-Nucleo*, right click on it: *Run As->Ac6 STM32 C/C++ Application*.
10. To test the I^2C communication, use user interface on the TE0808 HW platform (see Section 4.1 step 14).

5 Package Content

```
.
├── doc
├── nucleo-STM32H753
│   ├── prebuilt
│   │   └── STM32H753ZI-Nucleo.elf
│   └── STM32Cube_FW_H7_V1.9.0
│       └── Projects
│           └── NUCLEO-H753ZI
│               └── Examples
│                   └── I2C
│                       └── I2C_TwoBoards_ComIT
└── te0808-6EG-1EE
    ├── prebuilt
    │   ├── BOOT.bin
    │   ├── i2c.elf
    │   ├── psu_init.tcl
    │   └── system.hdf
    └── sdk
        └── i2c
```

6 References

[1] Trenz Electronic, „UltraITX+ Baseboard for Trenz Electronic TE080X UltraSOM+,“ [Online]. Available: https://shop.trenz-electronic.de/en/TEBF0808-04A-UltraITX-Baseboard-for-Trenz-Electronic-TE080X-UltraSOM.

[2] Trenz Electronic, „UltraSOM+ MPSoC Module with Zynq UltraScale+ XCZU6EG-1FFVC900E, 4 GB DDR4,“ [Online]. Available: https://shop.trenz-electronic.de/en/TE0808-04-6BE21-A-UltraSOM-MPSoC-Modul-with-Zynq-UltraScale-XCZU6EG-1FFVC900E-4-GB-DDR4.

[3] ST, „STM32 Nucleo-144 development board with STM32H753ZI MCU, supports Arduino, ST Zio and morpho connectivity,“ [Online]. Available: https://www.st.com/content/st_com/en/products/evaluation-tools/product-evaluation-tools/mcu-mpu-eval-tools/stm32-mcu-mpu-eval-tools/stm32-nucleo-boards/nucleo-h753zi.html.

Disclaimer

This disclaimer is not a license and does not grant any rights to the materials distributed herewith. Except as otherwise provided in a valid license issued to you by UTIA AV CR v.v.i., and to the maximum extent permitted by applicable law:

(1) THIS APPLICATION NOTE AND RELATED MATERIALS LISTED IN THIS PACKAGE CONTENT ARE MADE AVAILABLE "AS IS" AND WITH ALL FAULTS, AND UTIA AV CR V.V.I. HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and

(2) UTIA AV CR v.v.i. shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under or in connection with these materials, including for any direct, or any indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or UTIA AV CR v.v.i. had been advised of the possibility of the same.

Critical Applications:
UTIA AV CR v.v.i. products are not designed or intended to be fail-safe, or for use in any application requiring fail-safe performance, such as life-support or safety devices or systems, Class III medical devices, nuclear facilities, applications related to the deployment of airbags, or any other applications that could lead to death, personal injury, or severe property or environmental damage (individually and collectively, "Critical Applications"). Customer assumes the sole risk and liability of any use of UTIA AV CR v.v.i. products in Critical Applications, subject only to applicable laws and regulations governing limitations on product liability.