The Embedded I/O Company

TEWS TECHNOLOGIES

TXMC637

Reconfigurable FPGA with 16 x 16 bit Analog Input 8 x 16 bit Analog Output and 32 digital I/O

Version 1.0

User Manual

Issue 1.0.3

June 2024

TEWS TECHNOLOGIES GmbH
Am Bahnhof 7 25469 Halstenbek, Germany
Phone: +49 (0) 4101 4058 0 Fax: +49 (0) 4101 4058 19
e-mail: info@tews.com www.tews.com

TXMC637-10R

16 x Analog In, 8 x Analog Out, 32 digital Front I/O, 64 direct FPGA Rear I/O Lines and 4 MGTs Rear I/O,

XC7A200T-2 FBG676 Artix-7 FPGA, 512MB DDR3

Style Conventions

Hexadecimal characters are specified with prefix 0x, i.e. 0x029E (that means hexadecimal value 029E).

For signals on hardware products, an ‚Active Low' is represented by the signal name with # following, i.e. IP_RESET#.

Access terms are described as:

W	Write Only
R	Read Only
R/W	Read/Write
R/C	Read/Clear
R/S	Read/Set

Issue	Description	Date
1.0.0	First Issue	January 2022
1.0.1	Added a note on the use of the ADC single ended correction values.	February 2022
1.0.2	The expression calibration values has been replaced by correction values.	May 2023
1.0.3	General Revision with minor corrections and additions.	June 2024

Table of Contents

1 PRODUCT DESCRIPTION 9
2 TECHNICAL SPECIFICATION 10
3 HANDLING AND OPERATION INSTRUCTION 12
3.1 ESD Protection 12
3.2 Thermal Considerations 12
4 PCI DEVICE TOPOLOGY 13
4.1 User FPGA (Artix-7) 14
4.2 BCC (Board Configuration Controller) FPGA 14
4.2.1 PCI Configuration Registers (PCR) 14
4.2.2 PCI BAR Overview 14
4.2.2.1 Local Configuration Register Space 15
4.2.2.2 In-System Programming Data Space 16
5 REGISTER DESCRIPTION 17
5.1 User FPGA (Artix-7) 17
5.2 BCC (Board Configuration Controller) FPGA 18
5.2.1 DAC Control / Status Register – 0x00 18
5.2.2 DAC Output Voltage Range Register – 0x04 19
5.2.3 Reference DAC Voltage Control Register – 0x10 to 0x4C 20
5.2.4 Digital I/O Interface Configuration /Status Register - 0x60 22
5.2.5 TXMC637 I/O Area Temperature Sensor Register - 0x64 24
5.2.6 User FPGA JTAG Control and Status Register – 0x80 25
5.2.7 User FPGA JTAG Signal Line Register – 0x84 26
5.2.8 User FPGA JTAG TMS Data Register - 0x88 27
5.2.9 User FPGA JTAG TDI Data Register - 0x8C 27
5.2.10 User FPGA JTAG TDO Data Register - 0x90 28
5.2.11 I2C Bridge Register - 0xA0 28
5.2.12 Interrupt Enable Register - 0xC0 28
5.2.13 Interrupt Status Register - 0xC4 29
5.2.14 User FPGA Configuration Control/Status Register - 0xD0 30
5.2.15 User FPGA Configuration Data Register - 0xD4 31
5.2.16 ISP Control Register - 0xE0 31
5.2.17 ISP Configuration Register - 0xE4 32
5.2.18 ISP Command Register - 0xE8 32
5.2.19 ISP Status Register - 0xEC 33
5.2.20 Temperature Sensor Register - 0xF4 34
5.2.21 Serial Number Register - 0xF8 34
5.2.22 BCC - FPGA Code Version - 0xFC 34
6 INTERRUPTS 35
6.1 Interrupt Sources 35
6.1.1 User FPGA (Artix-7) 35
6.1.2 BCC 35
6.2 Interrupt Handling 35
6.2.1 User FPGA (Artix-7) 35
6.2.2 BCC 35
7 FUNCTIONAL DESCRIPTION 36
7.1 User FPGA Block Diagram 36
7.2 User FPGA Highlights 37

7.3 User FPGA Gigabit Transceiver (MGT) 38
7.4 User FPGA Configuration 40
7.4.1 Master Serial SPI Flash Configuration 40
7.4.2 Manually User FPGA SPI Flash Reconfiguration 42
7.4.3 Slave Select Map Configuration 43
7.4.4 Configuration via JTAG 45
7.4.4.1 User JTAG Chain 45
7.4.4.2 TEWS Factory JTAG Chain 45
7.4.5 Programming User FPGA SPI Configuration Flash 46
7.4.6 Erasing User FPGA SPI Configuration Flash 47
7.4.7 Sector Erasing User FPGA SPI Configuration Flash 48
7.4.8 Reading User FPGA SPI Configuration Flash 49
7.5 BCC (Board Configuration Controller) FPGA 50
7.5.1 I2C Interface to BCC Register 50
7.6 Clocking 51
7.6.1 FPGA Clock Sources 51
7.7 Serial ADC Interface 53
7.7.1 Overview 53
7.7.2 Analog Input Stage 54
7.7.3 Analog Input Mode 55
7.7.4 Analog Input Range 55
7.7.5 A/D Data Coding 56
7.7.6 User FPGA Pinning 56
7.8 Parallel DAC Interface 58
7.8.1 Overview 58
7.8.2 Output Voltage Range 58
7.8.3 User FPGA Pinning 59
7.8.4 Programming Hints for AD5547 60
7.9 TTL I/O Interface 61
7.9.1 User FPGA Pinning 62
7.10 RS422 Interface 64
7.10.1 RS422 FPGA Interface 64
7.10.2 User FPGA Pinning 65
7.11 I/O Pull Configuration 67
7.12 Memory 68
7.12.1 DDR3 SDRAM 68
7.12.2 SPI-Flash 70
7.12.3 I2C - EEPROM 70
7.12.3.1 ADC and DAC Correctable Errors 72
7.12.3.2 ADC Correction Values 72
7.12.3.3 DAC Correction Values 78
7.12.3.4 ADC Data Correction Formula 80
7.12.3.5 DAC Data Correction Formula 80
7.12.3.6 DAC voltage ranges 81
7.12.3.7 Version of EEPROM data structure 82
7.13 Rear I/O Interface 83
7.14 Digital Interface to XMC P16 Connector 85
7.15 JTAG Controller to User FPGA JTAG Interface 86
7.15.1 Bit-IO 86
7.15.2 Vector-IO 86
7.16 I2C Bridge 87
7.17 On-Board Indicators 88
7.17.1 User FPGA Pinning 90
7.18 User FPGA Reset Inputs 90

8 DESIGN HELP 91
8.1 Board Reference Design 91
9 I/O INTERFACES 92
9.1.1 Front I/O - ADC Analog Input Level 92
9.1.2 Front I/O – Analog Output Level 94
9.1.3 Rear I/O Interface 94
10 I/O DESCRIPTION 95
10.1 Overview 95
10.2 Front I/O Connector (X1) 96
10.3 Rear I/O Connector (P14) 99
10.4 Rear I/O Connector (P16) 100
10.5 JTAG Header (X2) 101

List of Figures

FIGURE 1-1 : BLOCK DIAGRAM 9
FIGURE 4-1 : PCIE/PCI DEVICE TOPOLOGY 13
FIGURE 5-1 : DAC AND REF. DAC SCHEMATA 20
FIGURE 5-2 : DAC OUTPUT CHANNEL 20
FIGURE 7-1 : FPGA BLOCK DIAGRAM 36
FIGURE 7-2 : USER FPGA MGT BLOCK DIAGRAM 38
FIGURE 7-3 : USER JTAG-CHAIN 45
FIGURE 7-4 : TEWS FACTORY JTAG-CHAIN 45
FIGURE 7-5 : FPGA CLOCK SOURCES 51
FIGURE 7-6 : ADC INPUT STRUCTURE 53
FIGURE 7-7 : ANALOG INPUT STAGE 54
FIGURE 7-8 : ANALOG OUTPUT SECTION 58
FIGURE 7-9 : ANALOG OUTPUT SECTION 58
FIGURE 7-10 : ONE CHANNEL TTL I/O INTERFACE 61
FIGURE 7-11 : 1 CHANNEL DIGITAL I/O MULTIPLEXER BLOCK DIAGRAM 64
FIGURE 7-12 : ONE CHANNEL RS422 I/O INTERFACE 64
FIGURE 7-13 : USER FPGA I2C TO BCC I2C BRIDGE 87
FIGURE 9-1 : DAC OUTPUT INTERFACE 94
FIGURE 10-1 : FRONT PANEL I/O CONNECTOR NUMBERING 96
FIGURE 10-2 : PIN ASSIGNMENT P14 REAR I/O CONNECTOR 99
FIGURE 10-3 : PIN ASSIGNMENT P16 REAR I/O CONNECTOR 100

List of Tables

TABLE 2-1 : TECHNICAL SPECIFICATION 11
TABLE 4-1 : ON-BOARD PCIE / PCI DEVICES 13
TABLE 4-2 : PCI CONFIGURATION REGISTERS 14
TABLE 4-3 : PCI BAR OVERVIEW 14
TABLE 4-4 : LOCAL CONFIGURATION REGISTER SPACE 15
TABLE 5-1 : DAC CONTROL AND STATUS REGISTER 18
TABLE 5-2 : DAC OUTPUT VOLTAGE RANGE REGISTER 19
TABLE 5-3 : REFERENCE DAC VOLTAGE CONTROL REGISTER 20
TABLE 5-4 : VOLTAGE CODING FOR THE REFERENCE DAC 21
TABLE 5-5 : I/O PULL-RESISTOR CONFIGURATION REGISTER 23
TABLE 5-6 : TXMC637 I/O AREA TEMPERATURE SENSOR REGISTER 24
TABLE 5-7 : USER FPGA JTAG CONTROL AND STATUS REGISTER 25
TABLE 5-8 : USER FPGA JTAG SIGNAL LINE REGISTER 27
TABLE 5-9 : USER FPGA JTAG TMS DATA REGISTER 27
TABLE 5-10 : USER FPGA JTAG TDI DATA REGISTER 27
TABLE 5-11 : USER FPGA JTAG TDO DATA REGISTER 28
TABLE 5-12 : I2C BRIDGE REGISTER 28
TABLE 5-13 : INTERRUPT ENABLE REGISTER 28
TABLE 5-14 : INTERRUPT STATUS REGISTER 29
TABLE 5-15 : USER FPGA CONFIGURATION CONTROL/STATUS REGISTER 30
TABLE 5-16 : USER FPGA CONFIGURATION DATA REGISTER 31
TABLE 5-17 : ISP CONTROL REGISTER 31
TABLE 5-18 : ISP CONFIGURATION REGISTER 32
TABLE 5-19 : ISP COMMAND REGISTER 32
TABLE 5-20 : ISP STATUS REGISTER 33
TABLE 5-21 : TXMC637 TEMPERATURE SENSOR REGISTER 34
TABLE 5-22 : TXMC637 SERIAL NUMBER REGISTER 34
TABLE 5-23: BCC - FPGA CODE VERSION 34
TABLE 7-1 : TXMC637 FPGA FEATURE OVERVIEW 37
TABLE 7-2 : FPGA BANK USAGE 37
TABLE 7-3 : USER FPGA MGT CONNECTIONS 38
TABLE 7-4 : USER FPGA MGT REFERENCE CLOCKS 39
TABLE 7-5: USER FPGA I2C INTERFACE TO BCC 50
TABLE 7-6 : AVAILABLE FPGA CLOCKS 52
TABLE 7-7 : ANALOG INPUT RANGES 55
TABLE 7-8 : A/D DATA CODING 56
TABLE 7-9: ADC INTERFACE CONNECTIONS 57
TABLE 7-10: TXCM637 PARALLEL DAC INTERFACE 59
TABLE 7-11 : USER FPGA TTL IO AND OE PINS 63
TABLE 7-12 : USER FPGA RS422 INTERFACE PINS 66
TABLE 7-13: I/O PULL CONFIGURATION 67

TABLE 7-14 : USER FPGA PINS....67
TABLE 7-15 : DDR3 SDRAM INTERFACE....69
TABLE 7-16 : FPGA SPI-FLASH CONNECTIONS....70
TABLE 7-17: FPGA I2C EEPROM CONNECTIONS....70
TABLE 7-18: ADC CORRECTION VALUES....77
TABLE 7-19: DAC CORRECTION VALUES....79
TABLE 7-20: ADC AND DAC VOLTAGE RANGES....81
TABLE 7-21: VERSION OF EEPROM DATA STRUCTURE....82
TABLE 7-22 : DIGITAL REAR I/O INTERFACE....84
TABLE 7-23 : XMC P16 DIGITAL I/O INTERFACE....85
TABLE 7-24: BOARD-STATUS AND USER LEDS....89
TABLE 7-25: TXMC637 USER ON-BOARD INDICATORS....90
TABLE 7-26: USER FPGA RESET INPUTS....90
TABLE 9-1 : ADC INPUT WITHOUT GROUND REFERENCE....92
TABLE 9-2 : ANALOG INPUT RANGES....93
TABLE 9-3 : DAC ELECTRICAL INTERFACE....94
TABLE 10-1: PIN ASSIGNMENT FRONT PANEL I/O CONNECTOR DIGITAL PART....97
TABLE 10-2: PIN ASSIGNMENT FRONT PANEL I/O CONNECTOR ANALOG PART....98
TABLE 10-3: PIN ASSIGNMENT JTAG HEADER....101

1 Product Description

The TXMC637 is a standard single-width Switched Mezzanine Card (XMC) compatible module providing a user configurable FPGA (Artix-7).

32 ADC input channels, based on four ADAS3022, can be software configured in groups to operate in single-ended or differential mode. Each of the 32 channels has a resolution of 16bit and can work with up to 1 MSPS. The programmable gain amplifier is software configurable and allows a full-scale input voltage range of up to +/-10.24V.

The TXMC637 DAC output channels are based on the Dual 16bit AD5547 DAC. Each DAC output is designed as a configurable single-ended bipolar analog output. Output voltage is configurable as ±10.0V, ±5.0V or ±2.5V.

32 ESD-protected TTL lines provide a flexible digital interface. All I/O lines are individually programmable either as input or output. Input I/O lines are tri-stated and could be used with the on-board pull up or as tri-stated output. Each TTL I/O line has a pull resistor sourced by a common pull source. The pull voltage level is selectable to be either +3.3V, +5V and additionally GND.

16 of these ESD-protected TTL lines can be switched to be either a TTL interface or RS422 interface. Switching is done via the User FPGA. All 8 RS422 transceivers have individual internal switchable terminations.

The User FPGA is connected to a 512 Mbytes, 16 bit wide DDR3L SDRAM. (to be used with the Xilinx Memory Interface Generator)

For customer specific I/O extension or inter-board communication, the TXMC637 provides 64 FPGA I/Os on P14 (directly connected). All P14 I/O lines can be configured in accordance with 7-Series SelectI/O features e.g. as 64 single ended LVCMOS25 or as 32 differential LVDS25 interface

The User FPGA is configured by a serial quad SPI flash. For full PCIe specification compliance, the XILINX Tandem Configuration Feature can be required for FPGA configuration. XILINX Tandem Methodologies “Tandem PROM” should be the favored Methodology. The SPI flash device is in-system programmable. An in-circuit debugging option is available via a JTAG header for (real-time) debugging of the FPGA design.

User applications for the TXMC637 with Artix-7 FPGA can be developed using the design software Vivado Design Suite. A license for the Vivado Design Suite design tool is not required (Artix-Family is fully covered by Xilinx WebPack Tool).

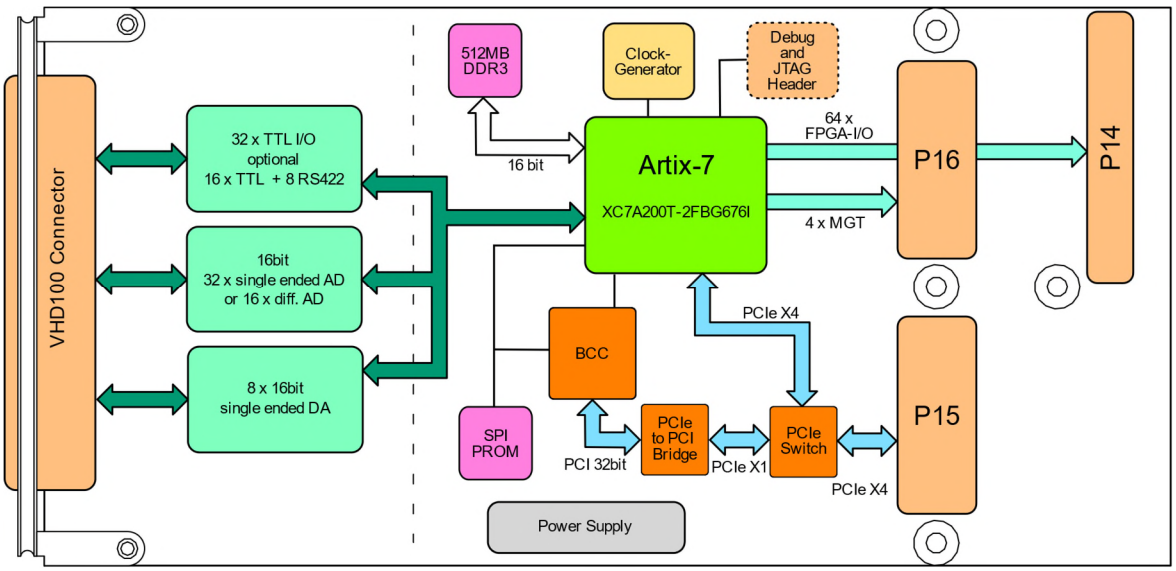


Figure 1-1 : Block Diagram

2 Technical Specification

PCIe Interface		
Mechanical Interface	Switched Mezzanine Card (XMC) Interface confirming to ANSI/VITA 42.0-2008 (Auxiliary Standard) Standard single-width (149mm x 74mm)	
Electrical Interface	PCI Express x4 Link (Base Specification 2.1) compliant interface conforming to ANSI/VITA 42.3-2006 (PCI Express Protocol Layer Standard)	
On-Board Devices		
PCI Express Switch	PI7C9X2G312GP (Pericom)	
PCI Express to PCI Bridge	XIO2001 (Texas Instruments)	
User configurable FPGA	TXMC637-10R: XC7A200T-2FBG676I (Xilinx)	
SPI-Flash	MT25QL128 (Micron) 128Mbit (contains TXMC637 FPGA BRD) or compatible; +3.3V supply voltage	
DDR3 RAM	1 x MT41K256M16TW-107 (Micron) 256Meg x 16Bit	
Board Configuration Controller	LCMXO2-7000HC (Lattice)	
ADC	ADAS3022BCPZ -16 (Analog Devices)	
DAC	AD5547BRUZ (Analog Devices)	
I/O Interface		
A/D Channels	32 Single-Ended or 16 Differential A/D Channels Input Configuration per ADC Device: 8x Single-Ended A/D Channels or 4x Differential A/D Channels A/D Channel Input Range Options: Single-Ended Input Voltage Ranges: ±0.64V, ±1.28V, ±2.56V, ±5.12V, ±10.24V, ±12.228V Differential Input Voltage Ranges: ±0.64V, ±1.28V, ±2.56V, ±5.12V, ±10.24V, ±20.48V, ±24.576V On-board analog input 1st order low-pass filter with -3dB cutoff frequency of approx. 105kHz on all A/D channels Max conversion Rate from 100ksps to 800ksps, depending on the number of active channels per ADC device. Pseudo-Simultaneous conversion for all A/D channels	
D/A Channels	8 Single-Ended 16 Bit D/A Channels Output range configurable per D/A channel. Simultaneous Conversion for all D/A Channels.	
	Maximum single-ended Output Voltage – Vout Maximum Output Drive Current for each Output Maximum Capacitive Load for each Output Typical Settling Time for a 10mA / 1000pF	±10V 10mA 1000pF < 1µs
Digital Front I/O Channels	32 digital I/O Lines • Default configured as 32 ESD-protected TTL lines • 16 I/O lines are configurable as 8 differential RS422 I/O lines with individual Termination enable.	

<table>
<tr><td>Digital Rear I/O Channels</td><td>64 direct FPGA I/O lines to P14 Rear I/O connector
Can be used as single-ended or differential I/O
FPGA I/O Standard: LVCMOS25, LVTTL25 and LVDS25
4 MGT line to P16 Rear I/O connector
Each line consists of one differential RX and TX pair.
Transmission speeds of up to 3.125 Gb/s are possible.</td></tr>
<tr><td colspan="2">I/O Connector</td></tr>
<tr><td>Front I/O</td><td>Front I/O Honda - HDRA-EC100LFDT-SL+</td></tr>
<tr><td>P14 Rear I/O</td><td>64 pin Mezzanine Connector (Molex 71436-2864 or compatible)</td></tr>
<tr><td>P16 Rear I/O</td><td>114 pin Mezzanine Connector (Samtec – ASP-105885-01)</td></tr>
<tr><td colspan="2">Physical Data</td></tr>
<tr><td>Power Requirements</td><td>Depends on FPGA design
With TXMC637 Board Reference Design / without external load<table><tr><td></td><td>typical @ +12V VPWR</td><td>typical @ +5V VPWR</td></tr><tr><td>TXMC637-10R</td><td>0.930A</td><td>2.0A</td></tr></table></td></tr>
<tr><td>Temperature Range</td><td>Operating
Storage</td><td>-40°C to +85°C
-40°C to +85°C</td></tr>
<tr><td>MTBF</td><td>TXMC637-10R: 170000 h
MTBF values shown are based on calculation according to MIL-HDBK-217F and MIL-HDBK-217F Notice 2; Environment: G_B 20°C.
The MTBF calculation is based on component FIT rates provided by the component suppliers. If FIT rates are not available, MIL-HDBK-217F and MIL-HDBK-217F Notice 2 formulas are used for FIT rate calculation.</td></tr>
<tr><td>Humidity</td><td>5 – 95 % non-condensing</td></tr>
<tr><td>Weight</td><td>TXMC637-10R: 130g</td></tr>
</table>

Table 2-1 : Technical Specification

3 Handling and Operation Instruction

3.1 ESD Protection

The TXMC637 is sensitive to static electricity. Packing, unpacking and all other handling of the TXMC637 has to be done in an ESD/EOS protected Area.

3.2 Thermal Considerations

Forced air cooling is required during operation. Without forced air cooling, damage to the device can occur.

Please also note chapter “Thermal Management”.

4 PCI Device Topology

The TXMC637 consists of two FPGAs. Both FPGAs are designed as PCIe / PCI endpoint devices. One FPGA is the User FPGA (Artix-7) which can be programmed with user defined FPGA code. The second FPGA takes control of on-board hardware functions of TXMC637 and also the configuration control of the User FPGA. This second FPGA is the BCC (Board Configuration Controller).

The BCC PCI endpoint is connected via a PCI-to-PCIe Bridge to the second x1 Downstream Port of the PCIe Switch (Pericom PI7C9X2G312GP). The User FPGA (Artix-7 PCIe endpoint) is directly connected to the first x4 Downstream Port.

The x4 Upstream Port of the PCIe Switch is connected to the XMC P15 Connector, communicating with the host system.

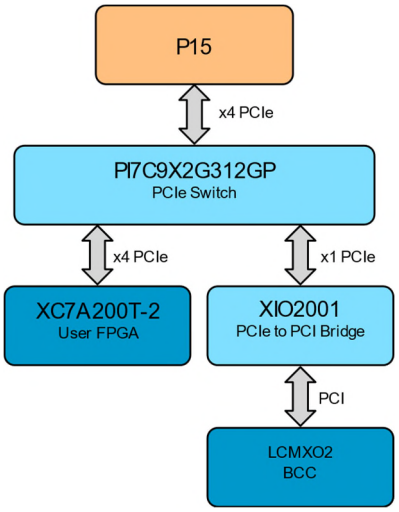


Figure 4-1 : PCIe/PCI Device Topology

Device	Vendor ID	Device ID	Class Code	Description (as shown by lspci)
PI7C9X2G312GP	0x12D8 (Pericom)	0x2312	0x060400	PCIe Switch: 0x04h to indicate device as PCI-to-PCI Bridge 0x06h to indicate device as Bridge device
XIO2001	0x104C (Texas Instruments)	0x8240	0x060400	PCI bridge: Texas Instruments 0x04h to indicate device as PCI-to-PCI Bridge 0x06h to indicate device as Bridge device
XC7A200T-2	user defined			Device identification for the User programmable FPGA is defined by user. The data will be created with the Xilinx Vivado “7 Series Integrated Block for PCI Express” IP generator.
BCC LCMXO2	0x1498 (TEWS)	0x927D	0x068000	Bridge Device: TEWS Technologies GmbH Device 927D (TXMC637).

Table 4-1 : On-Board PCIe / PCI Devices

4.1 User FPGA (Artix-7)

The User FPGA address map depends on the user application and is not part of this specification.

4.2 BCC (Board Configuration Controller) FPGA

4.2.1 PCI Configuration Registers (PCR)

PCI CFG Register Address	Write '0' to all unused (Reserved) bits				PCI writeable	Initial Values (Hex Values)
	31 24	23 16	15 8	7 0		
0x00	Device ID		Vendor ID		N	927D 1498
0x04	Status		Command		Y	0480 000B
0x08	Class Code			Revision ID	N	068000 01
0x0C	BIST	Header Type	PCI Latency Timer	Cache Line Size	Y[7:0]	00 00 00 08
0x10	PCI Base Address 0 for Local Address Space 0				Y	FFFFFF00
0x14	PCI Base Address 1 for Local Address Space 1				Y	FFFFFF00
0x18	PCI Base Address 2 for Local Address Space 2				N	00000000
0x1C	PCI Base Address 3 for Local Address Space 3				N	00000000
0x20	PCI Base Address 4 for Local Address Space 4				N	00000000
0x24	PCI Base Address 5 for Local Address Space 5				N	00000000
0x28	PCI CardBus Information Structure Pointer				N	00000000
0x2C	Subsystem ID		Subsystem Vendor ID		N	927D 1498
0x30	PCI Base Address for Local Expansion ROM				Y	00000000
0x34	Reserved			New Cap. Ptr.	N	000000 40
0x38	Reserved				N	00000000
0x3C	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	Y[7:0]	00 00 01 00

Table 4-2 : PCI Configuration Registers

4.2.2 PCI BAR Overview

BAR	Size (Byte)	Space	Prefetch	Port Width (Bit)	Endian Mode	Description
0	256	MEM	No	32	Little	Local Configuration Register Space
1	256	MEM	No	32	Little	In-System Programming Data Space

Table 4-3 : PCI BAR Overview

4.2.2.1 Local Configuration Register Space

Offset to PCI Base Address	Register Name	Size (Bit)
0x00	DAC Control and Status Register	32
0x04	DAC Output Voltage Range Register	32
0x08 – 0x0F	Reserved	-
0x10 – 0x2C	Register for VOFF and VREF of DAC Channel 1 to 8	8 x 32
0x30 – 0x5F	Reserved	-
0x60	Digital I/O Interface Configuration /Status Register	32
0x64	TXMC637 I/O Area Temperature	32
0x68 – 0x7F	Reserved	-
0x80	Artix-7 JTAG Control Register	32
0x84	Artix-7 JTAG Interface Register	32
0x88	Artix-7 JTAG Signal TMS Data Register	32
0x8C	Aritx-7 JTAG Signal TDI Data Register	32
0x90	Artix-7 JTAG Signal TDO Data Register	32
0x94 – 0x9F	Reserved	-
0xA0	I2C Bridge Register	32
0xA4 – 0xBF	Reserved	
0xC0	Interrupt Enable Register	32
0xC4	Interrupt Status Register	32
0xC8	Reserved	-
0xCC	Reserved	-
0xD0	User FPGA Configuration Control/Status Register	32
0xD4	User FPGA Configuration Data Register (Slave SelectMAP)	32
0xD8	Reserved	-
0xDC	Reserved	-
0xE0	ISP Control Register (SPI)	32
0xE4	ISP Configuration Register (SPI)	32
0xE8	ISP Command Register (SPI)	32
0xEC	ISP Status Register (SPI)	32
0xF0	Reserved	-
0xF4	Temperature Sensor Register	32
0xF8	Serial Number Register	32
0xFC	BCC – FPGA Code Version	32

Table 4-4 : Local Configuration Register Space

4.2.2.2 In-System Programming Data Space

The In-System Programming (ISP) Data Space is used for passing user FPGA configuration data for in-system programming of the User FPGA SPI Flash.

For ISP write/program instructions, the data must be written (zero-based) to the ISP Data Space before the instruction is started. The data must cover a complete SPI Flash memory page.

For ISP read instructions, the data can be read (zero-based) from the ISP Data Space after the instruction is done. The data is passed for a complete SPI Flash memory page.

The ISP Data Space size is 256byte, covering an SPI Flash Memory Page. All supported SPI Flash read and write instructions are page-based.

Control and status register for ISP are located in the Local Configuration Register Space. The data register for direct FPGA ISP is also located in the Local Configuration Register Space.

5 Register Description

5.1 User FPGA (Artix-7)

The FPGA register description depends on the user application and is not part of this specification.

5.2 BCC (Board Configuration Controller) FPGA

5.2.1 DAC Control / Status Register – 0x00

The output voltage ranges of the TXMC637 DAC outputs are set via *DAC Control / Status Register* and *DAC Output Voltage Range Register*.

For the three predefined ranges (±10V, ±5V and ±2.5V), the *DAC Output Voltage Range Register* is set first and then the values are transferred via the *DAC Control / Status Register*.

For the individual range selection the *Reference DAC Voltage Control Register* must also be set before updating the data via *DAC Control / Status Register*.

Bit	Symbol	Description	Access	Reset Value
31:16	-	Reserved	R	0
15:14	-	Reserved	R	0
7:6	-	Reserved	R	0
5	DAC1_OUTP_UPD	DAC #1 Output Update Initiate DAC Device Output Update with currently adjusted output values Self-Clearing	R/S	0
4	DAC1_CLR	DAC #1 Clear Causes DAC to be cleared. Includes setup with current adjusted values Self-Clearing	R/S	0
3	DAC1_OVRTMP	DAC #1 Over Temperature Status is provided by DAC device	R	0
2	-	Reserved		
1	DAC1_OUTPSTTLE	DAC #1 Output Settle Internal generated settling pulse (set to 20.5us)	R	0
0	DAC1_IOBSY	DAC #1 I/O Busy Indicates that • Controller is not in idle • Span-Configuration update is pending • Output Value update is pending • Internal synchronization is pending	R	0

Table 5-1 : DAC Control and Status Register

5.2.2 DAC Output Voltage Range Register – 0x04

Bit	Symbol	Description	Access	Reset Value
31:16	-	Reserved	-	00
15:14	REF_DAC8	DAC Output Voltage range selection for DAC channel 8 00 : ±10V range (default value) 01 : ±5V range 10 : ±2.5V range 11 : individual range selection	R/W	0b00
…	-	…	…	…
1:0	REF_DAC1	DAC Output Voltage range selection for DAC channel 1 00 : ±10V range (default value) 01 : ±5V range 10 : ±2.5V range 11 : individual range selection	R/W	0b00

Table 5-2 : DAC Output Voltage Range Register

For individual voltage range selection also see the chapter *5.2.3Reference DAC Voltage Control Register.*

5.2.3 Reference DAC Voltage Control Register – 0x10 to 0x4C

The 8 TXMC637 DAC outputs consist of $f(x) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cos\frac{n\pi x}{L} + b_n \sin\frac{n\pi x}{L}\right)$ dual DAC devices (AD5547). Each of these DACs could be used independently.

For the generation of the reference voltage of these 8 TXMC637 DAC channels, an additional serial DAC device is placed on the TXMC637. This Reference DAC has 16 analog outputs. Two reference voltages for each TXMC637 DAC output.

Via these reference voltages (VREF and VOFF), an individual voltage range can be set for each TXMC637 DAC channel. Both reference voltages are each set as 16 bit value via a 32bit register of the BCC. This results in 8 32bit registers via which the reference voltage and thus the output voltage range of the TXMC637 can be set.

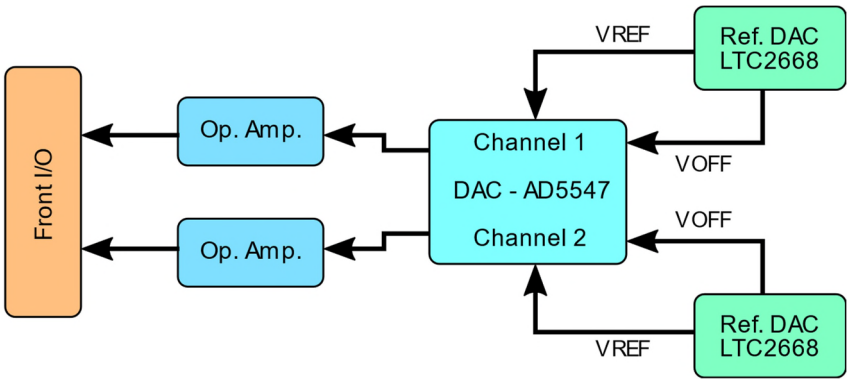


Figure 5-1 : DAC and Ref. DAC Schemata

One 32bit Register for each of the 8 TXMC637 DAC outputs.

Bit	Symbol	Description	Access	Reset Value
31:16	VREF	16-bit Data Value for VREF which specifies the half voltage swing of the reference voltage for the AD5547 DAC channel.	R/W	0x8000
15:0	VOFF	16-bit Data Value for VOFF which represents the negative reference voltage of the AD5547 DAC channel.	R/W	0x8000

Table 5-3 : Reference DAC Voltage Control Register

Each TXMC637 analog output consists of a half AD5547 output DAC, an operational amplifier driving the output load, and two references DAC outputs each for VREF and VOFF.

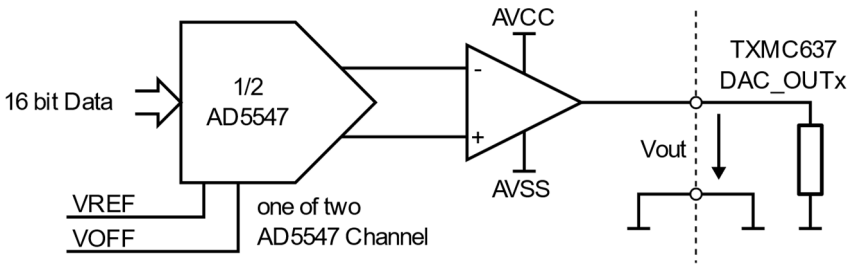


Figure 5-2 : DAC Output Channel

The following table shows the data coding for VREF and VOFF output voltage range.

Description	Digital Code
+9.9997V	0xFFFF
+5.0V	0xC000
+2.5V	0xA000
0V	0x8000
-2.5V	0x6000
-5.0V	0x4000
-10.0V	0x0000

Table 5-4 : Voltage coding for the Reference DAC

The output voltage range is then calculated with these two reference values (VREF and VOFF) as follows:

Vrange_high = -1 * VOFF

Vrange_low = -1 * VOFF - 2 * VREF

See the following Examples for:

VOFF	VREF	Vrange_low	Vrange_high	Voltage Range	Note
+10V (0xFFFF)	-10V (0x0000)	-10V	+10V	-10V … +10V	typical range
+5V (0xC000)	-5V (0x4000)	-5V	+5V	-5V … +5V	typical range
+1.25V (0x2000)	-5V (0x4000)	-1.25V	+8.75V	-1.25V … +8.75V	asymmetric range
-5V (0x4000)	+5V (0xC000)	+5V	-5V	+5V … -5V	changes sign

Thus, any desired output voltages can be generated. But in any case it is important to ensure that the voltages Vout1 and Vout2 are never set above +10V or below -10V. Voltages above these limits cannot be generated and lead to incorrect outputs.

5.2.4 Digital I/O Interface Configuration /Status Register - 0x60

Bit	Symbol	Description	Access	Reset Value
31:18	-	Reserved	-	0
17	IO_MUX_FLT	Multiplexer Fault Indicaton 0: device is operational 1: fault condition signaled *Fault assertion is captured (registered) to ensure detection of short-term events*	R/C	0
16	IO_MUX_OE	Multiplexer Output Enable Controls all multiplexed digital IO Lines in common. It is strongly recommended to enable the Multiplexer Outputs only after correct MUX configuration. 0: disable all multiplexed digital IO Lines 1: enable all multiplexed digital IO Lines	R/W	0
15	IO_MUX_SEL7	Digital Front I/O Line Multiplexer Selection. 1: corresponding I/O lines are TTL 0: corresponding I/O lines are RS422 TbD: *The digital I/O channels that will be multiplexed is determined during the layout or routing process.*	R/W	0xFF
14	IO_MUX_SEL6			
13	IO_MUX_SEL5			
12	IO_MUX_SEL4			
11	IO_MUX_SEL3			
10	IO_MUX_SEL2			
9	IO_MUX_SEL1			
8	IO_MUX_SEL0			
7:5	-	Reserved	-	0
4	PULL_SRC_SEL	I/O Group Pull Source Selection Defines whether the BCC or the User FPGA controls the I/O Group Pull Selection. 0: BCC 1: User FPGA	R/W	0
3:2	PULL_G1	I/O Group Pull Selection Value could be changed only if Control/Status Register PULL_CNT is set to BCC controlling. 11 : pull-down 10 : pull-up to 3.3V 01 : pull-up to 5V 00 : No pull I/O lines are summarized in the following groups. G0 = I/O_0 … I/O_15 G1 = I/O_16 … I/O_31	R/W	0x0
1:0	PULL_G0			

Table 5-5 : I/O Pull-Resistor Configuration Register

Each TTL I/O Line has a 4k7 Pull-Resistor. The 32 I/O Lines are divided into two groups (G0, G1) which can be configured as 3.3V pull-up, 5V pull-up or pull-down. In addition, the Pull-Resistors can float.

If the Pull-Resistors float, the user should keep in mind that the 16 I/O Lines of a group are connected via their Pull-Resistors.

The default adjustment is that the USER FPGA code must control the I/O Pull Configuration depending on USER FPGA I/O Function (see also chapter "I/O Pull Configuration").

5.2.5 TXMC637 I/O Area Temperature Sensor Register - 0x64

Bit	Symbol	Description	Access	Reset Value
31:21	-	Reserved	r	-
20	TMP441_AUTO _TRD_EN	TMP441 Automatic Temperature Read Enable Controls the periodic board temperature read feature. Refresh time = 1s '0' = disabled '1' = enabled *Automatic mode must be disabled before enabling the I2C bridge mode*	R/W	1
19:16	-	Reserved		0
15:8	TMP441_TEMP	TMP441 Temperature Data Measured data of the I/O area temperature sensor The read value of the temperature sensor is stored sign-extended as a 8bit two's complement. To actually calculate the temperature from the two's complement data value, use the following formula: Temperature (°C) = TEMP	R	-
7:0	-	Reserved	-	-

Table 5-6 : TXMC637 I/O Area Temperature Sensor Register

5.2.6 User FPGA JTAG Control and Status Register – 0x80

Bit	Symbol	Description		Access	Reset Value
31:10	-	Reserved		-	00
9:8	JTAG_VIO_CTRL_STAT	JTAG Vector-I/O Controller Status Signals the status of the Vector I/O controller. 0b00 = Controller disabled 0b01 = Controller idle (read-for-request) 0b10 = Ongoing Vector I/O controller operation 0b11 = illegal *Initiate JTAG operations only if the Vector I/O controller is in idle state.*		r	01
7:5	-	Reserved		-	0
4	JTAG_EXT#_INT	JTAG External/Internal Indicates whether a JTAG device is detected/present at the Debug Connector or not. 0b0 = Device detected/present 0b1 = No device detected *Detection is based on the Debug Connector Present signal*		r	-
3	-	Reserved		-	0
2:0	JTAG_MUX	BCC controlled JTAG Multiplexer to USER FPGA		r/w	0b000
		0b000	Fixed priority. Which JTAG connection is active depends on the inserted cable. *The priority is as follows:* *1) Debug Connector* *2) USB JTAG* *3) BCC JTAG (V-I/O or B-I/O)*		
		0b001	Hardwired JTAG interface from Debug Connector to Artix-7 is active.		
		0b010	reserved		
		0b100	BCC internal JTAG Controller is connected to Artix-7.		

Table 5-7 : User FPGA JTAG Control and Status Register

5.2.7 User FPGA JTAG Signal Line Register – 0x84

Refer to the chapter JTAG Controller to A7 JTAG Interface for a functional description.

Bit	Symbol	Description	Access	Reset Value
31:24	JTAG_VIO_TCK_CLK_DIV	JTAG Vector-I/O TCK Clock Divider Divider applied onto the internal processing clock to obtain/generate the JTAG TCK I/O clock JTAG_TCK = 53.2MHz / (divider +1) *Default value results in an 1.6625 MHz TCK frequency*	r/w	0x1F
23:16	JTAG_VIO_XFER_LEN	JTAG Vector-I/O Transfer Length Sets the number of transfer cycles (TCK clocks) performed during a Vector-I/O operation *Default value causes 256 full clock cycles (rising and falling edges) to be emitted on TCK*	r/w	0xFF
15:14	-	Reserved	-	0
13	JTAG_VIO_SHIFT_REQ	JTAG Vector-I/O Shift Request Initiates a TMS/TDI shift-out operation (incl. TDO read-in) operation *TMS/TDI are coupled. Both vectors are shifted out at the same time.* *The TMS/TDI data interface registers limit the transfer length*	r/s	0
12	JTAG_VIO_GET_REQ	JTAG Vector-I/O Get Request Initiates a TDO read-in only operation *The TDO data interface register limits the transfer length if all read-in data is required*	r/s	0
11:9	-	Reserved	-	0
8	JTAG_VIO_EN-	JTAG Vector-I/O Enable Controls the state of the BCC JTAG Vector-I/O Interface. 0b0 = Vector-I/O disabled 0b1 = Vector-I/O enabled *Functionality is held in reset until enabled is set*	r/w	0

7	JTAG_BIO_TDO	JTAG Bit-I/O - TDO TDO line output state of the User FPGA	r	-
6	JTAG_BIO_TDI	JTAG Bit-I/O - TDI Output state for the User FPGA TDI input line	r/w	1
5	JTAG_BIO_TMS	JTAG Bit-I/O - TMS Output state for the User FPGA TMS input line	r/w	1
4	JTAG_BIO_TCK	JTAG Bit-I/O - TCK Output state for the User FPGA TCK input line	r/w	0
3:1	-	Reserved	-	0
0	JTAG_BIO_EN	JTAG Bit-I/O Enable Controls the state of the BCC JTAG Bit-I/O Interface. 0b0 = Bit-I/O disabled 0b1 = Bit-I/O enabled	r/w	0

Table 5-8 : User FPGA JTAG Signal Line Register

5.2.8 User FPGA JTAG TMS Data Register - 0x88

Bit	Symbol	Description	Access	Reset Value
31:0	JTAG_VIO_TMS_DATA	JTAG Vector-I/O TMS Data Sets the JTAG TMS bit data that is shifted-out during TMS/TDI shift operations. *Note: Bit 0 is shifted-out first (right-alignment)*	r/w	0

Table 5-9 : User FPGA JTAG TMS Data Register

5.2.9 User FPGA JTAG TDI Data Register - 0x8C

Bit	Symbol	Description	Access	Reset Value
31:0	JTAG_VIO_TDI_DATA	JTAG Vector-I/O TDI Data Sets the JTAG TDI bit data that is shifted-out during TMS/TDI shift operations. *Note: Bit 0 is shifted-out first (right-alignment)*	r/w	0

Table 5-10 : User FPGA JTAG TDI Data Register

5.2.10 User FPGA JTAG TDO Data Register - 0x90

Bit	Symbol	Description	Access	Reset Value
31:0	JTAG_VIO_TDO _DATA	JTAG Vector-I/O TDO Data Accumulates TDO bit data read-in during TMS/TDI shift and TDO get request operations *Note: Bit 0 is shifted-in last (right-alignment)*	r/w	0

Table 5-11 : User FPGA JTAG TDO Data Register

5.2.11 I2C Bridge Register - 0xA0

Refer to the chapter I2C Bridge for a functional description.

Bit	Symbol	Description	Access	Reset Value
31:1	-	-	-	0
0	I2C_BRDG_MODE_EN	I2C Bridge Mode Enable Controls the USER I2C Bus BCC reach-through onto the management I2C Bus '0' = disabled '1' = enabled	r/w	0

Table 5-12 : I2C Bridge Register

5.2.12 Interrupt Enable Register - 0xC0

Bit	Symbol	Description	Access	Reset Value
31:2		Reserved		0
1	ISP_INS_IE	ISP SPI Instruction Done Event Interrupt Enable 0: Interrupt Disabled 1: Interrupt Enabled While disabled, the corresponding bit in the Interrupt Status Register is '0'. Disabling interrupts does not affect the interrupt source.	R/W	0
0	ISP_DAT_IE	ISP SPI Page Data Request Event Interrupt Enable 0: Interrupt Disabled 1: Interrupt Enabled While disabled, the corresponding bit in the Interrupt Status Register is '0'. Disabling interrupts does not affect the interrupt source.	R/W	0

Table 5-13 : Interrupt Enable Register

5.2.13 Interrupt Status Register - 0xC4

Bit	Symbol	Description	Access	Reset Value
31:2		Reserved		0
1	ISP_INS_IS	ISP SPI Instruction Done Event Interrupt Status When set, the PCI INTA# interrupt is asserted. The Interrupt is cleared by writing a '1'. 0: Interrupt not active or disabled 1: Interrupt active and enabled	R/C	0
0	ISP_DAT_IS	ISP SPI Page Data Done Event Interrupt Status When set, the PCI INTA# interrupt is asserted. The Interrupt is cleared by writing a '1'. 0: Interrupt not active or disabled 1: Interrupt active and enabled	R/C	0

Table 5-14 : Interrupt Status Register

5.2.14 User FPGA Configuration Control/Status Register - 0xD0

Bit	Symbol	Description	Access	Reset Value
31:5		Reserved		0
4	A7_LINK_ENA	1: Artix-7 to PCIe-Switch LINK is enabled 0: Artix-7 to PCIe-Switch LINK is disabled	R/W	1
3	FP_INIT_STAT	User FPGA INIT_B Pin Status 0: FPGA INIT_B Pin Level is Low (active) 1: FPGA INIT_B Pin Level is High (not active)	R	x
2	FP_DONE_STAT	User FPGA DONE Pin Status The FPGA Done pin is high in case of successful FPGA configuration. 0: FPGA DONE Pin Level is Low (not active) 1: FPGA DONE Pin Level is High (active)	R	x
1	FP_RE_CFG	After power-up the FPGA automatically configures from the on-board SPI Flash in 'Master Serial / SPI' mode. User FPGA Re-Configuration 1: Set all FPGA I/O pins to High-Z and prepare a User FPGA Re-Configuration 1 → 0: Start User FPGA Re-Configuration	R/W	0
0	FP_CFG_MD	Set User FPGA Configuration Mode 0: Master Serial / SPI 1: Slave SelectMap (Parallel) After power-up the User FPGA automatically configures from the on-board SPI Flash in 'Master Serial / SPI' mode.	R/W	0

Table 5-15 : User FPGA Configuration Control/Status Register

5.2.15 User FPGA Configuration Data Register - 0xD4

Bit	Symbol	Description	Access	Reset Value
31:0	ISP_FP_DAT	ISP Select Map Write Data Write Data Register for direct Slave Select Map FPGA programming mode Must be written with 32-bit FPGA programming data until the FPGA Done pin goes high (after the actual programming data, writing some dummy data may be required).	W	-

Table 5-16 : User FPGA Configuration Data Register

The User FPGA Configuration Data Register is used to write data within the User FPGA Slave Select Map Configuration directly to the User FPGA.

5.2.16 ISP Control Register - 0xE0

Bit	Symbol	Description	Access	Reset Value
31:1		Reserved		0
0	ISP_EN	ISP Mode Enable 0: Disable ISP Mode 1: Enable ISP Mode This bit controls the BCC interface between BCC, SPI-Flash and the User FPGA (Artix-7). When set, the BCC is both SPI Flash Master and FPGA Configuration Interface Master. Must be set to 1 for direct Slave Select Map mode or SPI Flash programming. Must be set to 0 when the User FPGA should configure from the SPI Flash (e.g. after SPI Flash programming) in 'Master Serial / SPI' mode. Note, that for ISP Direct FPGA Programming, the FPGA must first be set to Slave Select Map configuration mode.	R/W	0

Table 5-17 : ISP Control Register

5.2.17 ISP Configuration Register - 0xE4

Bit	Symbol	Description	Access	Reset Value
31:24	ISP_SPI_ADD	SPI Flash Address A7-A0	W	0x00
23:16		SPI Flash Address A15-A8	W	0x00
15:8		SPI Flash Address A23-A16	W	0x00
7:0	ISP_SPI_INS	SPI Flash Instruction Code Supported Instructions: 0x02 – Page Program 0x20 – Sector Erase 0xC7 – Chip Erase 0x03 – Read Data	W	0x00

Table 5-18 : ISP Configuration Register

5.2.18 ISP Command Register - 0xE8

Bit	Symbol	Description	Access	Reset Value
31:2		Reserved	-	0
1	ISP_SPI_RST_CMD	ISP SPI Reset Command Bit Writing a '1' sets the Instruction Busy Bit in the ISP Status Register (if not already set). Breaks any ISP SPI instruction in progress and resets the ISP SPI logic. Check the Instruction Busy Bit in the ISP Status Register for reset done status. Always read as '0'.	R/W	0
0	ISP_SPI_INS_CMD	ISP SPI Start Instruction Command Bit Writing a '1' sets the SPI Instruction Busy Bit in the ISP Status Register and starts the configured SPI instruction. Ignored (lost) while the Instruction Busy Bit is set in the ISP Status Register. Always read as '0'.	R/W	0

Table 5-19 : ISP Command Register

5.2.19 ISP Status Register - 0xEC

Bit	Symbol	Description	Access	Reset Value
31:2		Reserved	-	0x00_0000
1	ISP_SPI_ INS_BSY	ISP SPI Instruction Busy Status Set & Cleared automatically by HW. Includes SPI Flash internal program/erase times. When clear again after being set, a new ISP SPI instruction may be started. Capable of generating an event based interrupt. 0: No ISP SPI Instruction in Progress 1: ISP SPI Instruction in Progress	R	0
0	ISP_SPI_ DAT_BSY	ISP SPI Data Transfer Busy Status Set & Cleared automatically by HW. Does not include SPI Flash internal program/erase times. When clear again after being set, new SPI Flash page data may be written to the ISP Data Space (in program mode) or SPI Flash page data is available in the ISP data space (in read mode). Capable of generating an event based interrupt. 0: No ISP SPI Data Transfer in Progress 1: ISP SPI Data Transfer in Progress	R	0

Table 5-20 : ISP Status Register

5.2.20 Temperature Sensor Register - 0xF4

Bit	Symbol	Description	Access	Reset Value
31:21	-	Reserved	r	-
20	TMP441_AUTO _TRD_EN	TMP441 Automatic Temperature Read Enable Controls the periodic board temperature read feature. Refresh time = 1s '0' = disabled '1' = enabled *Automatic mode must be disabled before enabling the I2C bridge mode*	r/w	1
19:16	-	Reserved		0
15:8	TMP441_TEMP	TMP441 Temperature Data Measured data of the on-board temperature sensor The read value of the temperature sensor is stored sign-extended as a 8bit two's complement. To actually calculate the temperature from the two's complement data value, use the following formula: Temperature (°C) = TEMP	r	-
7:0	-	Reserved	-	-

Table 5-21 : TXMC637 Temperature Sensor Register

5.2.21 Serial Number Register - 0xF8

Bit	Symbol	Description	Access	Reset Value
31:0	S_NUMBER	The value is the unique serial number of each TXMC637 module	r	-

Table 5-22 : TXMC637 Serial Number Register

Example: 0x0091_5CC9 => SNo.: 9526473

5.2.22 BCC - FPGA Code Version - 0xFC

Bit	Symbol	Description	Access	Reset Value
31:0	CODE_VER	The value shows the BCC Firmware code version of the TXMC637 module.	R	-

Table 5-23: BCC - FPGA Code Version

6 Interrupts

6.1 Interrupt Sources

6.1.1 User FPGA (Artix-7)

The User FPGA interrupt sources depend on the user application and are not part of this target specification.

6.1.2 BCC

The BCC provides two interrupt sources. Both interrupts are used in the context of SPI programming instructions. The Slave Select Map Mode does not provide interrupt support.

- ISP SPI Instruction Done Event Interrupt

  Event-based internal interrupt that becomes active (if enabled), when the ISP SPI Instruction Busy status bit changes from busy to not-busy.

- ISP SPI Page Data Done Event Interrupt

  Event-based internal interrupt that becomes active (if enabled), when the ISP SPI Data Busy status bit changes from busy to not-busy.

These interrupts indicates that an ISP operation is done.

6.2 Interrupt Handling

6.2.1 User FPGA (Artix-7)

The interrupt handling depends on the user application and is not part of this target specification.

6.2.2 BCC

The Local Configuration Register Space located in the BCC provides an interrupt enable register and an interrupt status register.

Interrupt sources are enabled in the interrupt enable register. If enabled, upon an interrupt event, the corresponding bit in the interrupt status register is set. If any interrupt status bit is set in the interrupt status register, an interrupt request is asserted.

Both Interrupts of the BCC must be cleared via write access to the corresponding Interrupt Status Flag in the Interrupt Status Register (active-high write clear).

7 Functional Description

7.1 User FPGA Block Diagram

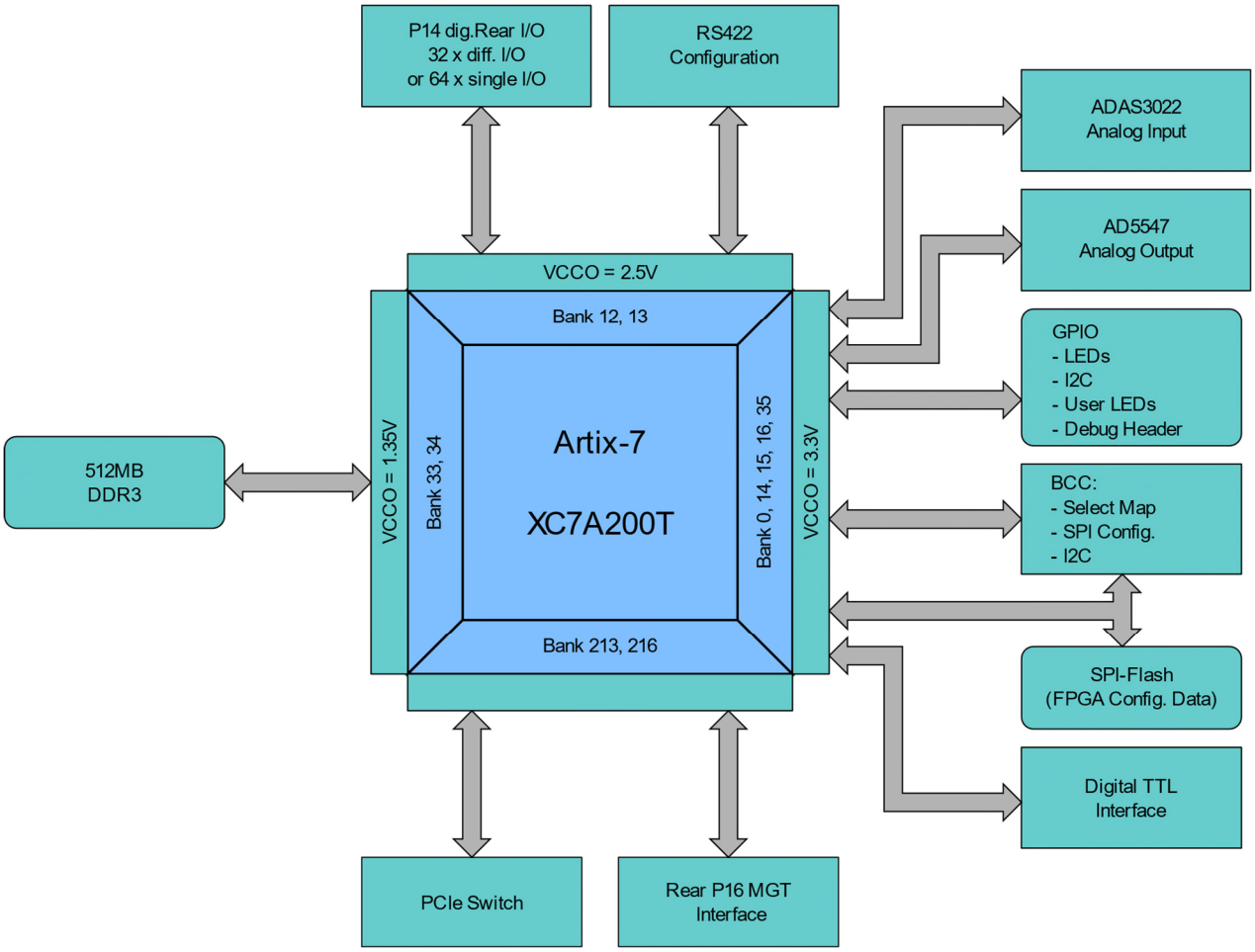


Figure 7-1 : FPGA Block Diagram

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.2 User FPGA Highlights

The FPGA is a Artix-7 XCAK200T FPGA. The Artix-7 FPGA in a FBG676 package provides eight MGT, four for high speed rear I/O communication and four for the PCI Express interface (x4 Linkage).

Artix-7	Logic Cells	Slices	DSP Slices	Block RAM (Kb)			CMTs	GTP	XADC Block
				18 Kb	36 Kb	max(Kb)			
XC7A200T	215,360	33.650	740	730	365	13,140	10	8	1

Table 7-1 : TXMC637 FPGA Feature Overview

PCI Express Highlights:

- Compliant to the PCI Express Base Specification 2.1 with Endpoint and Root Port capability.
- Supports Gen1 (2.5 Gb/s) and Gen2 (5 Gb/s)

XADC Highlights:

XADC (Analog-to-Digital Converter)

On-chip temperature (±4°C max error) and power supply (±1% max error) sensors

- Continuous JTAG access to ADC measurements
- Internal access to all internal sensors of the Artix-7

The board supports JTAG, master serial mode configuration from SPI-Flash or Slave Select MAP configuration for the User FPGA (Artix-7) via the Board Configuration Controller (BCC).

The User FPGA is equipped with 8 I/O banks and 8 GTP (Gigabit Transceiver).

Bank	VCCO	VREF	Signals	Note
Bank 0	3.3V	none	SPI Configuration Slave Select Map Configuration parallel DAC Interface ADC Interface User LEDs Digital TTL Interface Digital RS422	
Bank 14	3.3V	none		
Bank 15	3.3V	none		
Bank 16	3.3V	none		
Bank 35	3.3V	none		
Bank 12	2.5V	none	64bit Rear I/O Interface to P14 RS422 Enable Interface Additional digital interface to P16	
Bank 13	2.5V	none		
Bank 33	1.35V	0.625V	16 bit DDR3 Memory Interface 512MB	
Bank 34	1.35V	0.625V		
Bank 213	Connected to XMC P16 Rear I/O connector			
Bank 216	PCIe X4 Interface to PCIe Switch Device			

Table 7-2 : FPGA Bank Usage

For a detailed overview of all user FPGA I/O pins the TEWS Board Reference Design and the appendix can be used.

7.3 User FPGA Gigabit Transceiver (MGT)

The TXMC637 provides four MGT as Artix-7 PCI Express Endpoint Block and four MGT for a high speed XMC P16 Rear I/O interface.

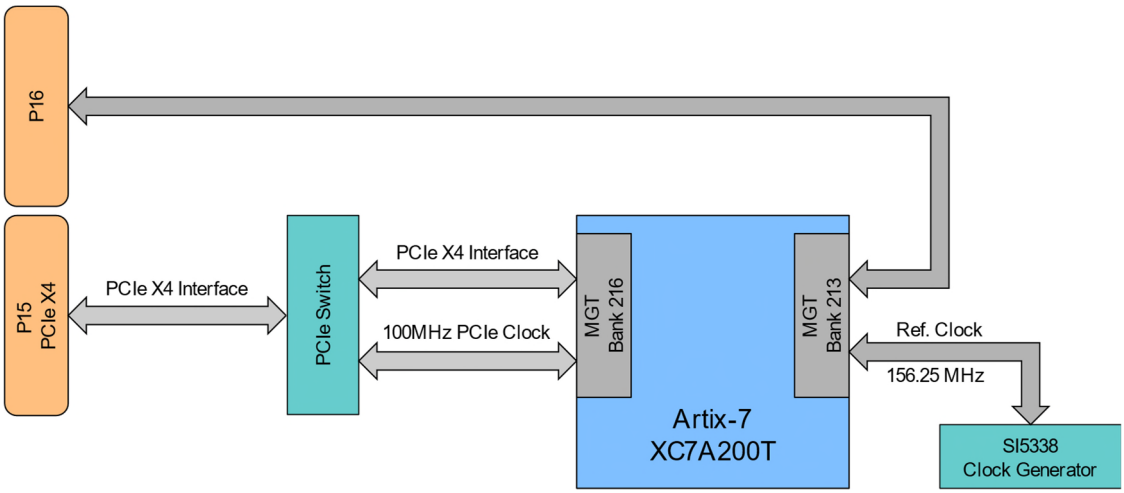


Figure 7-2 : User FPGA MGT Block Diagram

MGT	TXMC637 Signal	FPGA Pins	Connected to
MGTXTXP0_213	MGTTX0	AC10 / AD10	connected to Rear I/O XMC P16 Interface
	MGTRX0	AC12 / AD12	
MGTXTXP1_213	MGTTX1	AE9 / AF9	
	MGTRX1	AE13 / AF13	
MGTXTXP2_213	MGTTX2	AC8 / AD8	
	MGTRX2	AC14 / AD14	
MGTXTXP3_213	MGTTX3	AE7 / AF7	
	MGTRX3	AE11 / AF11	
MGTXTXP0_216	PET_07	B7 / A7	used for PCI Express Endpoint Block and connected to XMC P15 Connector
	PER_07	B11 / A11	
MGTXTXP1_216	PET_06	D8 / C8	
	PER_06	D14 / C14	
MGTXTXP2_216	PET_05	B9 / A9	
	PER_05	B13 / A13	
MGTXTXP3_216	PET_04	D10 / C10	
	PER_04	D12 / C12	

Table 7-3 : User FPGA MGT Connections

The MGT clock MGTREFCLK0_216 (PCI Express Endpoint Block clock reference) of 100 MHz is generated by the PI7C9X2G312GP PCIe Switch. The MGTREFCLK0_213 is connected to a 156.25 MHz clock output of the Si5338 low jitter clock generator. MGTREFCLK1_213 and MGTREFCLK1_216 are not used on the TXMC637.

MGT	TXMC637 Signal	FPGA Pins	Connected to
MGTREFCLK0_213	CLK_MGT	AA13 / AB13	156.25 MHz Si5338 Clock Generator
MGTREFCLK1_213	not used	AA11 / AB11	not connected
MGTREFCLK0_216	REFCLK_O2	F11 / E11	100 MHz PI7C9X2G312GP PCIe Switch
MGTREFCLK1_216	not used	F13 / E13	not connected

Table 7-4 : User FPGA MGT Reference Clocks

7.4 User FPGA Configuration

The Artix-7 can be configured by the following interfaces:

- Master Serial SPI Flash Configuration Interface
- JTAG Interface via FPGA JTAG Connector
- PCIe Interface via BCC FPGA Slave Select Map Interface Configuration

The change of the configuration mode is done with a configuration register of the BCC FPGA.

At Power-up, the TXMC637 User FPGA (Artix-7) always configures via x4 SPI Interface by "Master Serial / SPI" mode. At factory default the SPI Flash contains the TEWS Board Reference Design application for the TXMC637 User FPGA device.

7.4.1 Master Serial SPI Flash Configuration

It is important for User FPGA Configuration via SPI Master Mode that the ISP Mode Enable (ISP_EN) bit is clear to disable the ISP Mode. This is also the default value after Power Up.
See also Register Description of TXMC637 Configuration Device.

To comply with the PCI-Express specification it is necessary to perform the configuration as quickly as possible. The PCIe specification demands that a PCI device must be accessible after 100ms (120ms). To speed up the SPI Configuration the following points must be taken into account for SPI Bitstream generation.

- External Clock Master (53.2MHz) should be used.
- If external Clock Master is used, also the SPI Falling Edge Option must be used.
- SPI Configuration Bus Width should be set to X4.
- Xilinx Tandem Configuration Feature could be used for full PCI-Express specification compliance. Already during PCI-Express IP Core generation this configuration feature must be included. (For more information see: Xilinx XAPP1179).
- If the Tandem Configuration feature is used, the Persist Option is mandatory.
- For smaller FPGA content, it is sometimes also possible to comply with the PCI-Express specification, when only Bitstream Compression is used.

To avoid damage on the BCC or User FPGA (Artix-7) if Tandem configuration or the Persist Option is used, the User FPGA must be set into reconfigure Mode by using the "FP_RE_CFG" Bit of the User FPGA Configuration Control/Status Register before Programming or Clearing the SPI Flash.

All required general settings, bitstream setting and additional constraints for a compressed tandem SPI X4 configuration could be found in the TXMC637 Board Reference Design application.

Note: Changing a configuration to tandem configuration results in a change in the timing of the design.

7.4.2 Manually User FPGA SPI Flash Reconfiguration

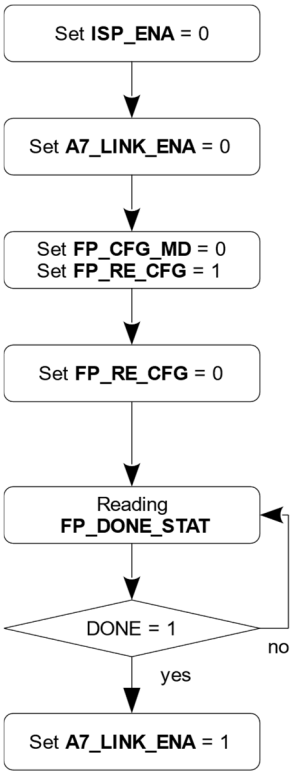
A manually User FPGA Reconfiguration can be performed with the User FPGA Reconfigure Command in the Global Configuration Register.

Set the User FPGA Reconfigure Command to set the User FPGA to configuration state with all FPGA I/O pins are High-Z.

Use the following procedure to perform a User FPGA SPI Reconfiguration

- Assure that ISP Mode Enable is disabled.
- By Reconfiguring the Artix-7 the XILINX PCIe endpoint is reloaded and is temporarily not available on the PCI bus. To avoid error messages of the PCIe switch the link between the PCIe Switch and the Artix-7 is disabled.
- Set the User FPGA Configuration Mode (FP_CFG_MD) to Master Serial / SPI and prepare the FPGA Reconfiguration.
- Start the FPGA Reconfiguration by setting the FP_RE_CFG bit of the User FPGA Configuration Control/Status Register to 0.
- Assure that the FPGA DONE Pin status shows a successful FPGA Configuration.

  0: FPGA DONE Pin Level is Low (FPGA is not configured)
  1: FPGA DONE Pin Level is High (FPGA is configured)
- The link between the PCIe Switch and the Artix-7 must be enabled.



A successful User FPGA configuration is indicated with FPGA_DONE status in the Global Status Register and the on-board User FPGA Done LED.

It must be considered in any case, that the Reconfiguration of the User FPGA also reconfigures the PCIe Endpoint of the User FPGA. This leads to the consequence that the PCI Header of the User FPGA PCIe Endpoint no longer exists. For this purpose it is necessary to disable the link between the PCIe switch and the User FPGA PCIe Endpoint before preparing the FPGA Reconfiguration and to enable the link again after reconfiguration.

Additionally, after FPGA Reconfiguration the User FPGA PCIe Endpoint PCI Header must be configured again. If the PCIe interface of the User FPGA PCIe Endpoint does not change. Device ID, Vendor ID, Class Code and PCI Bars do not change, the PCI header could be saved before the FPGA Reconfiguration and written back to configuration space after the Reconfiguration.

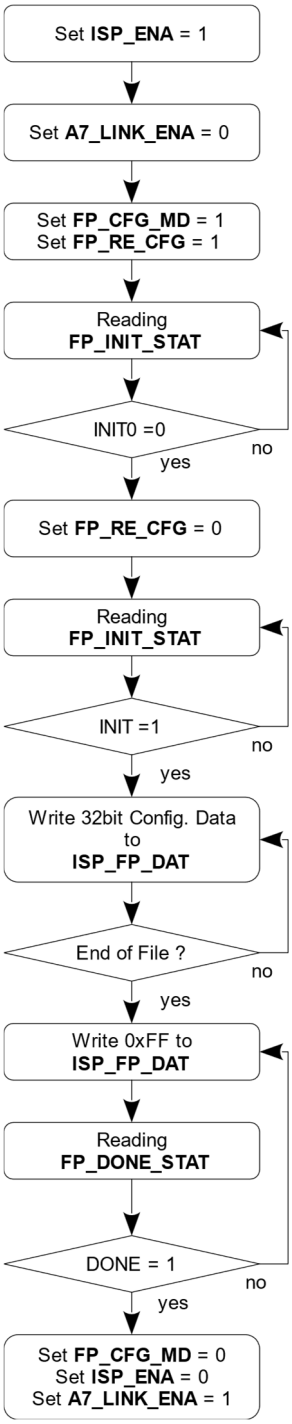
7.4.3 Slave Select Map Configuration

For direct User FPGA configuration via PCIe Interface the **User FPGA Configuration Mode** must be set to **Slave SelectMap** Mode. The on-board logic sets the User FPGA in configuration state with all FPGA I/O pins switches to High-Z. User FPGA is now ready for new configuration data.

The following procedure is required for Select Map Mode User FPGA configuration / reconfiguration.

- First the In System Program (ISP) Mode must be enabled.
- By reconfiguring the Artix-7 the XILINX PCIe endpoint is reloaded and is temporarily not available on the PCI bus. To avoid error messages of the PCIe switch the link between the PCIe Switch and the Artix-7 is disabled.
- Check response of the Artix-7 by reading the FPGA INIT_B pin value. If the Level is low the Artix-7 FPGA is in Reset Mode, and then configuration process could be continued.
- Start the FPGA Reconfiguration by setting the FP_RE_CFG bit of the User FPGA Configuration Control/Status Register to 0.
- Check response of the Artix-7 by reading the FPGA INIT_B pin value. While the FPGA INIT_B pin Level is low the Artix-7 isn't ready for configuration.
- If FPGA INIT_B pin high then the configuration data must be continually written to the ISP SelectMap Data Register. Typically 2860903 PCI write accesses are required to configure a Artix-7 325T.
- Dummy Write accesses to create configuration clock cycles while FP_DONE_STAT is low.
- A successful configuration of the User FPGA is indicated with FP_DONE_STAT in the User FPGA Configuration Control/Status Register and the on-board User FPGA Done LED.

  0: FPGA DONE Pin Level is Low (FPGA is not configured)
  1: FPGA DONE Pin Level is High (FPGA is configured)
- After reconfiguration was successful the User FPGA Configuration Mode and the ISP Mode could be disabled. Also the link between the PCIe Switch and the Artix-7 must be enabled.



If not all configuration data bytes are written the User FPGA is not configured correctly.

The number of bytes that must be written corresponds to the size of the XILINX configuration files. Typically the .bin or the .bit file could be used as data source.

The .bit file is the standard generated programming file. This is a binary configuration data file which contains header information that does not need to be downloaded to the FPGA. For generating the .bin file the BitGen option must be used. This is also a binary configuration data file but without header information. For configuration of the Artix-7 FPGA on the TXMC637 both files could be used. Both binary configuration data files have additional data to the actual configuration data.

See also the XILINX User Guide (ug470) “7 Series FPGAs Configuration” for more information about Configuration Details and Configuration Data File Formats.

The following BitGen options are mandatory for the Slave Select Map Configuration via BBC.

- External Clock Master (53.2MHz) must be used.
- In contrast to SPI Configuration Mode, the Falling Edge Option must be switched off.

Additional important BitGen Options:

- For a faster configuration the Bitstream Compression could be used.
- The Persist Option is not needed. But if this option is used, the User FPGA must be set into reconfigure Mode by using the “FP_RE_CFG” Bit of the User FPGA Configuration Control/Status Register before Programming or Clearing the SPI Flash.

Xilinx Tandem Configuration Feature cannot be used for Slave Select Map Configuration. It is therefore necessary to remove the Tandem Configuration Feature from the PCIe IP Core.

A design that is intended for the SPI configuration cannot be used by Slave Select Map configuration and vice versa.

7.4.4 Configuration via JTAG

The TXMC637 provides two JTAG chains which are accessible by the following connector options:

7.4.4.1 User JTAG Chain

For direct FPGA configuration, FPGA read back or in-system diagnostics with Vivado Logic Analyzer, the Molex Debug Connector can be used to access the JTAG-chain. Also an indirect SPI – PROM programming is possible via the User JTAG Chain. This Connector provides a direct connection to a Xilinx USB Programmer II compatible 2 mm shrouded header.

The second possible source is the BCC internal JTAG TAB controller. See chapter 7.12 JTAG Controller to A7 JTAG Interface.

The Molex JTAG connector has default priority. If a cable is connected here, the BCC switches this interface active. However, each of the sources can be selected via the User FPGA JTAG Control and Status Register - 0x80 in BCC.

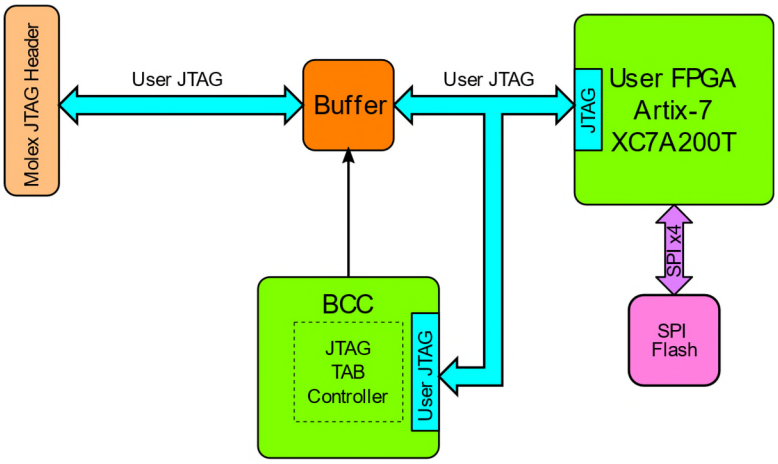


Figure 7-3 : User JTAG-Chain

7.4.4.2 TEWS Factory JTAG Chain

The TEWS Factory JTAG Chain is accessible from the PCIe Edge Connector.

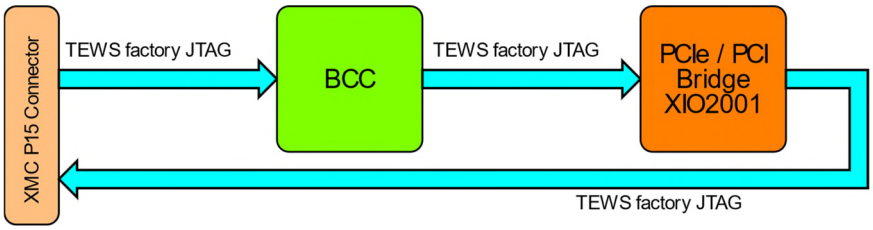


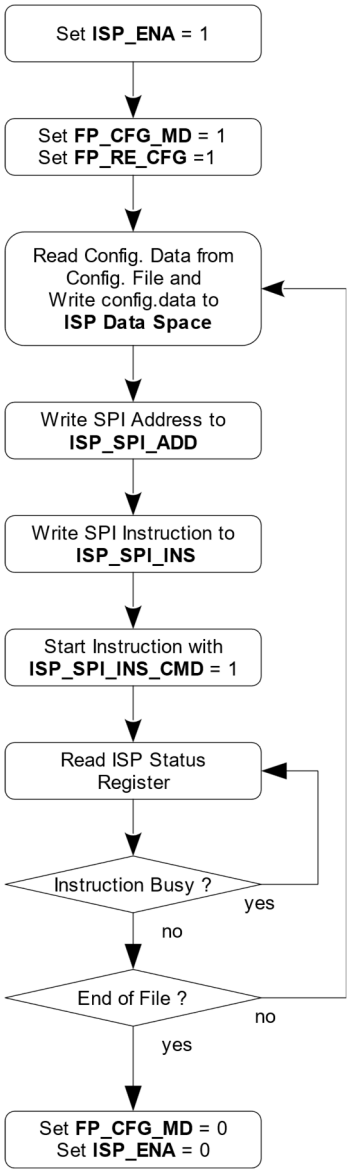
Figure 7-4 : TEWS Factory JTAG-Chain

7.4.5 Programming User FPGA SPI Configuration Flash

To program the User FPGA SPI Configuration Flash the **User FPGA Configuration Mode** must be set to **Master Serial / SPI** and the ISP Mode must be enabled.

The following procedure is required for User FPGA SPI Configuration Flash programming and subsequent reconfiguration of the User FPGA.

- Enable the ISP Mode in the ISP Mode Enable Register.
- Assure that User FPGA Configuration Mode is set to SPI Flash. If the FPGA is not configured or if it is possible that the FPGA accesses the SPI flash during BCC access set FP_RE_CFG = 0b1. Link must be set to disable previously!
- Read Configuration data from Configuration File and write Data to the In Circuit Programming Data Space. 256Byte (1 SPI Flash page) each time can be programmed maximally.
- Set the programming start address and write instruction in the ISP Configuration Register.
- Start the Instruction with ISP Command Register
- Wait on ISP SPI Instruction Done or ISP SPI Page Data Done for next write instruction.
- Process should be repeated until all configuration data is written to the SPI Flash
- After completion of the data programming, the ISP Mode bit must be cleared to set configuration path to User FPGA and a Reconfiguration can be performed.



A successful configuration of the User FPGA is indicated with FP_DONE_STAT in the User FPGA Configuration Control/Status Register and the on-board User FPGA Done LED.

The Programming Instruction always starts at address 0x00 to write data from the ISP Programming Data Space to the SPI flash.

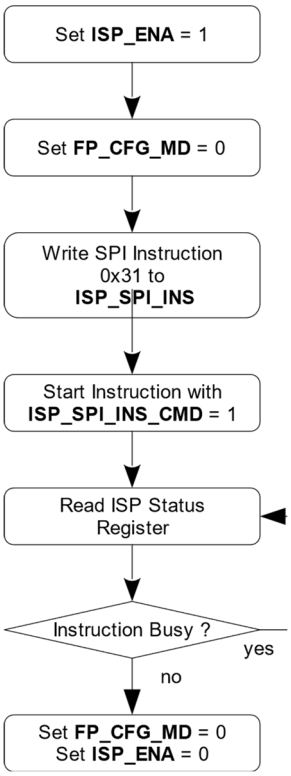
If not all configuration data bytes are written, the User FPGA is not configured correctly.

The source for the User FPGA SPI Configuration Flash data could be the .bin file. This file format can be created from the .bit file by using the XILINX Vivado software.

7.4.6 Erasing User FPGA SPI Configuration Flash

For Chip Erase of the User FPGA SPI Configuration Flash the **User FPGA Configuration Mode** must be set to **Master Serial / SPI** and the ISP Mode must be enabled.

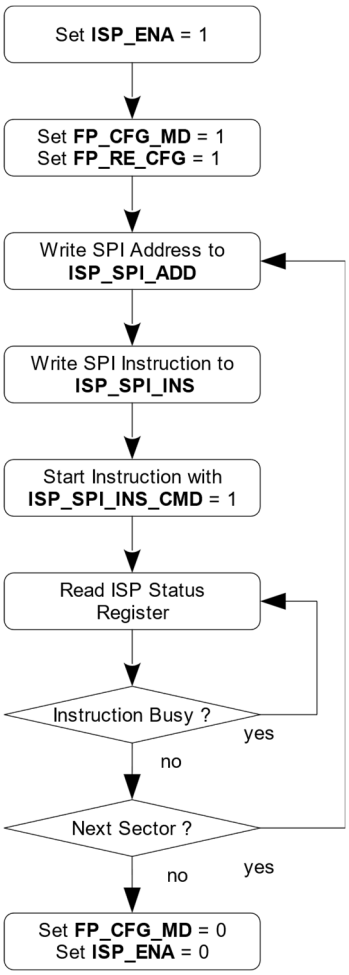
- Enable the ISP Mode in the ISP Mode Enable Register.
- Assure that User FPGA Configuration Mode is set to SPI Flash. If the FPGA is not configured or if it is possible that the FPGA accesses the SPI flash during BCC access set FP_RE_CFG = 0b1. Link must be set to disable previously!
- Set the Chip Erase instruction in the ISP Configuration Register.
- Start the Instruction with ISP Command Register
- Wait on ISP SPI Instruction Done or ISP SPI Page Data Done for erasing process end.
- After completion of the erasing process, the ISP Mode bit should be cleared to set configuration path to User FPGA or a User FPGA SPI Configuration Flash programming process could be done.



7.4.7 Sector Erasing User FPGA SPI Configuration Flash

For Sector Erase of the User FPGA SPI Configuration Flash the **User FPGA Configuration Mode** must be set to **Master Serial / SPI** and the ISP Mode must be enabled.

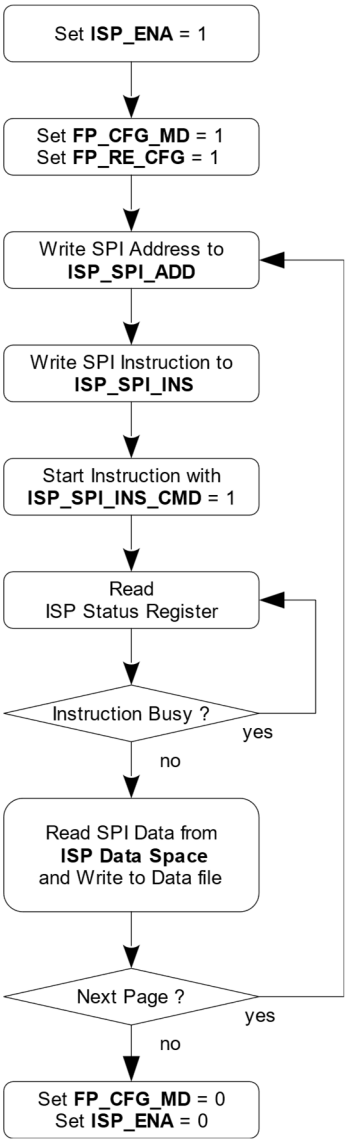
- Enable the ISP Mode in the ISP Mode Enable Register.
- Assure that User FPGA Configuration Mode is set to SPI Flash. If the FPGA is not configured or if it is possible that the FPGA accesses the SPI flash during BCC access set FP_RE_CFG = 0b1. Link must be set to disable previously!
- Write the Sector Address to the ISP Configuration Register
- Set the Chip Erase instruction in the ISP Configuration Register.
- Start the Instruction with ISP Command Register
- Wait on ISP SPI Instruction Done or ISP SPI Page Data Done for erasing process end.
- Process could be repeated for other sectors.
- After completion of the erasing process, the ISP Mode bit should be cleared to set configuration path to User FPGA or a User FPGA SPI Configuration Flash programming process could be done.



7.4.8 Reading User FPGA SPI Configuration Flash

To read the User FPGA SPI Configuration Flash the **User FPGA Configuration Mode** must be set to **Master Serial / SPI** and the ISP Mode must be enabled.

- Enable the ISP Mode in the ISP Mode Enable Register.
- Assure that User FPGA Configuration Mode is set to SPI Flash. If the FPGA is not configured or if it is possible that the FPGA accesses the SPI flash during BCC access set FP_RE_CFG = 0b1. Link must be set to disable previously!
- Set the reading start address and write instruction in the ISP Configuration Register.
- Start the Instruction with ISP Command Register
- Wait on ISP SPI Instruction Done or ISP SPI Page Data Done for next write instruction.
- Read one page of SPI Data from In Circuit Programming Data Space and write to Data file
- Process could be repeated until all needed data is written to the Data file.
- After completion of the reading process, the ISP Mode bit must be cleared to set configuration path back to User FPGA.



7.5 BCC (Board Configuration Controller) FPGA

The Board Configuration FPGA is factory configured, and handles the basic board setup and User FPGA (Artix-7) Configuration.

Changing or erasing the BCF (Board Configuration Firmware) content leads to an inoperable TXMC637 FPGA configuration.

7.5.1 I2C Interface to BCC Register

The TXMC637 BCC provides an I2C Interface to the User FPGA (Artix-7). Via this I2C Interface the TXMC637 Serial Number Register from the BCC Local Configuration Register Space could be read.

For the User FPGA (Artix-7) an I2C Master interface is required to use this interface. For this purpose the BCC provides an I2C slave interface.

Signal	Bank	V_{CCO}	Pin	Description
FPGA_SCL	15	3.3V	T24	Serial Clock Output A negative edge clock data out.
FPGA_SDA	15	3.3V	T25	Bisectional Serial Data

Table 7-5: User FPGA I2C Interface to BCC

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.6 Clocking

7.6.1 FPGA Clock Sources

As a central clock generator of the TXMC637 the Si5338 clock generator is used. This provides all necessary clocks for the User FPGA and the Configuration FPGA.

The following figure depicts an abstract User FPGA clock flow.

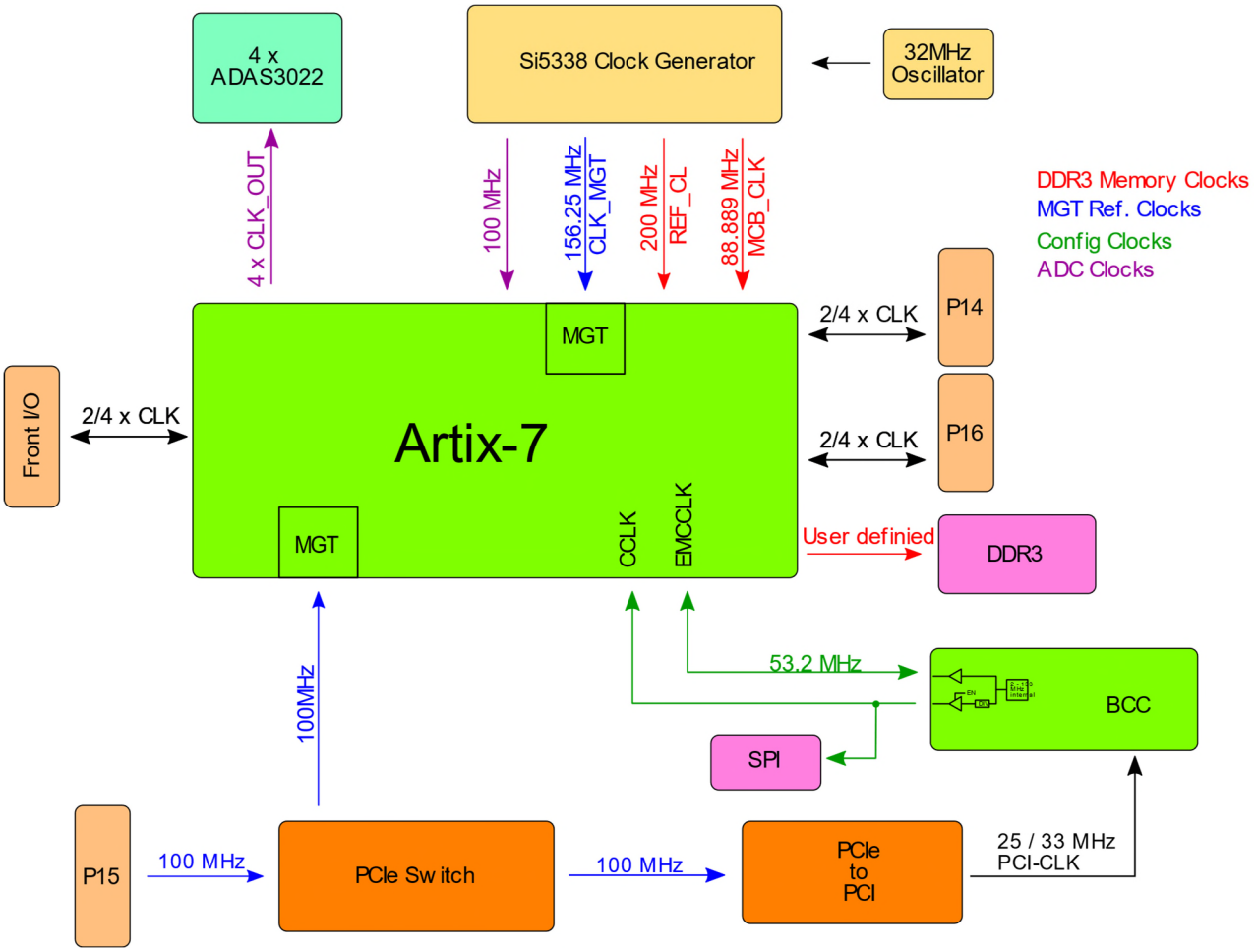


Figure 7-5 : FPGA Clock Sources

The following table lists the available clock sources on the TXMC637:

FPGA Clock Signal Name	FPGA Pin	Source	Description
CLK_MGT±	AA13 / AB13	SI5338 low-jitter clock generator	156.25 MHz differential MGT Reference clock for P16 Rear I/O Interface
REFCLKO2±	F11 / E11	PCIe Switch PI7C9X2G312GP	100 MHz differential PCIe Reference clock input
MCB_CLK±	R3 / P3	SI5338 low-jitter clock generator	88.889 MHz differential MCB CLK
REF_CLK±	P4 / N4	SI5338 low-jitter clock generator	200 MHz differential Reference clock
USER_CLKA	M21	SI5338 low-jitter clock generator	100 MHz Clock Input This clock is designated for ADC/DAC interface clock source.
TTL_IO_0	F5	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_1	G5	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_2	F4	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_3	G4	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_31	D19	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_30	C19	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_29	E20	Front I/O Line	Single ended TTL based I/O Line
TTL_IO_28	D20	Front I/O Line	Single ended TTL based I/O Line
REAR_IO0±	U22 / V22	Back I/O Connector	Diff. Back I/O Line from P14
REAR_IO1±	Y22 / Y23	Back I/O Connector	Diff. Back I/O Line from P14
REAR_IO19±	AC19 / AD19	Back I/O Connector	Diff. Back I/O Line from P14
REAR_IO20±	AA19 / AB19	Back I/O Connector	Diff. Back I/O Line from P14
REAR_IO23±	AA20 / AB20	Back I/O Connector	Diff. Back I/O Line from P14
REAR_IO25±	AB21 /AC21	Back I/O Connector	Diff. Back I/O Line from P14
DIG_IO_01±	W21 / Y21	Back I/O Connector	Diff. Back I/O Line from P16
DIG_IO_03±	U21 / V21	Back I/O Connector	Diff. Back I/O Line from P16
K7_EMCCLK	P16	BCC	53.2 MHz used for external configuration clock (CCLK)
CCLK	H13	BCC	external Artix -7 configuration clock

Table 7-6 : Available FPGA clocks

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.7 Serial ADC Interface

7.7.1 Overview

The 32 analog inputs of the TXMC637 are implemented with four ADAS3022 ADC devices. Each of these four SAR-ADCs has an 8 channel input MUX with integrated PGA. The eight analog input channels could be used as either eight single-ended or four differential inputs.

All channels of an ADC device are always operating in the same input mode (differential or single ended). However, the input mode may be different for the on-board ADC devices.

The ADAS3022 provides a Serial Peripheral Interface (SPI) for the digital connection to the User-FPGA. In addition, each device provides four additional control lines (RESET, Power-Down, Busy and CNV) for general and conversion process control.

Every single ADC is connected via a dedicated SPI bus, conversion control line CNV and the status line BUSY to the User-FPGA. This make possible to operate on all four ADCs simultaneously, thus exploiting the minimum conversion time.

Only the Power-Down and RESET control are shared by all four ADC devices.

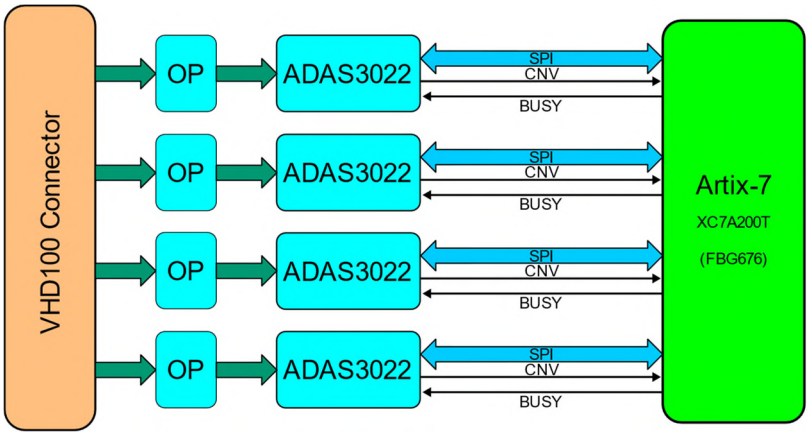


Figure 7-6 : ADC Input Structure

The ADAS3022 provides on all analog input lines diodes for protection against electrostatic charging.

7.7.2 Analog Input Stage

The TXMC637 Analog Input Stage provides

- Input Filter

  The TXMC637 provides a 1st order low-pass filter at the analog inputs for suppressing noise on the analog input lines (e.g. coupled noise from switching digital I/O lines running in the same I/O cable).

  The -3dB cutoff frequency of the input filter is approx. 105 kHz.

- Input Protection

  When the analog supply rails (approx. ±15V) are on, in case of applied analog input over-voltages of up to ±35V, the analog input is clamped at/to a level slightly below the supply rails. Operating range for the analog inputs is ±12.288V (to GND).

  When the supply rails are off, analog input voltages up to ±40V are tolerated.

- Input Buffer

  The TXMC637 provides analog input OPAMP buffers for decoupling the analog signal source.

- ADC Device Internal Input Path

  The ADAS3022 is an ADC with multiplexed analog inputs. Inside the ADAS3022 chip the analog input channels (pins) are connected to a single internal gain amplifier and a single internal SAR ADC unit by way of an internal analog multiplexer. Therefore, the ADAS3022 does not support real simultaneous sampling of its analog input channels.

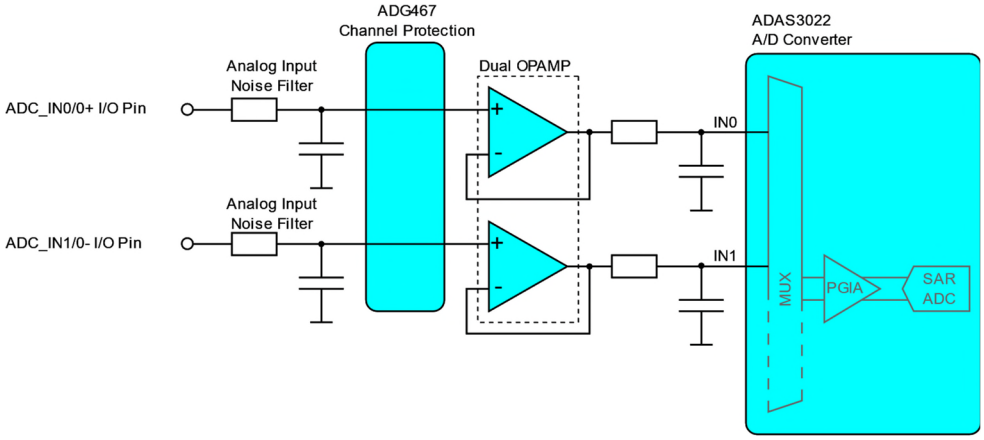


Figure 7-7 : Analog Input Stage

7.7.3 Analog Input Mode

The general input voltage mode (differential or single-ended) is configurable per on-board ADC device (not per A/D channel). Default input mode is Single-Ended.

For each ADC device operating in differential mode, the differential A/D channels 0 …3 are available (I/O signals ADC#_IN[0:3]+ and ADC#_IN[0:3]-).

For each ADC device operating in single-ended mode, the single-ended A/D channels 0 …7 are available (I/O signals ADC#_IN[0:7] and GND).

For more detail see ADAS3022 Datasheet.

7.7.4 Analog Input Range

An ADAS3022 integrated programmable gain amplifier is used to adapt the analog input voltage range to the internal (fix) SAR ADC voltage range. The analog input voltage range (i.e. the gain setting as shown in the overview table) is configurable for each individual A/D channel.

Note that the ADAS3022 ADC device does not provide unipolar input voltage ranges. For unipolar input signals, the appropriate (bipolar) input voltage range must be used.

The following table gives an overview of the supported analog input ranges for both analog input modes.

ADC Channel Configuration Option			**Differential Mode Analog Input Range** [2]		**Single-Ended Mode Analog Input Range**
Gain	**Differential Input Voltage Range**	**LSB**	**V_{IN+} Range** (to GND) **V_{IN-} Range** (to GND)	**V_{IN_DIFF} Range** ($V_{IN\_DIFF} = V_{IN+} - V_{IN-}$)	**V_{IN} Range** (to GND)
0.1666	±24.576V	750uV	±12.288V	±24.576V	±12.288V [3]
0.2	±20.48V	625uV	±10.24V	±20.48V	±10.24V [3]
0.4	±10.24V	312.5uV	±10.24V	±10.24V	±10.24V
0.8	±5.12V	156.3uV	±10.24V	±5.12V	±5.12V
1.6	±2.56V	78.13uV	±10.24V	±2.56V	±2.56V
3.2	±1.28V	39.06uV	±10.24V	±1.28V	±1.28V
6.4	±0.64V	19.53uV	±10.24V	±0.64V	±0.64V

1) ± indicates a voltage range (e.g. ±5.12V is used as a short form for -5.12V … +5.12V)
2) The V_{IN+} and V_{IN-} voltage levels must comply with both given ranges (that is the voltage levels must meet the given range when referenced to ground and must meet the given differential voltage range as well)
3) Only half of the available digital codes are useable for these Single-Ended configurations

Table 7-7 : Analog Input Ranges

The signal level on any Analog Input Pin on the I/O connector must not exceed 12.288V (referenced to ground)!

7.7.5 A/D Data Coding

The A/D data coding is always Two's Complement.

Available GAIN settings per A/D channel are: 0.16, 0.2, 0.4, 0.8, 1.6, 3.2 and 6.4.

Digital Output Code	Analog Input Voltage (V_{REF} = 4.096V)	Description
0x7FFF	(32767/32768) x (V_{REF}/GAIN)	Pos. Full Scale – 1 LSB
0x0001	(1/32768) x V_{REF}/GAIN	Mid-Scale + 1 LSB
0x0000	0V	Mid-Scale
0xFFFF	– (1/32768) x V_{REF}/GAIN	Mid-Scale – 1 LSB
0x8001	– (32767/32768) x V_{REF}/GAIN	Neg. Full-Scale + 1 LSB
0x8000	–V_{REF} / GAIN	Neg. Full-Scale

Table 7-8 : A/D Data Coding

7.7.6 User FPGA Pinning

Each ADC is connected to the User FPGA (Artix-7) via a dedicated serial clocked Interface. Each ADC device has one input clock, one output clock and one conversion signal. For each ADC channel there is a respective data output line, so both ADC channel transfers data at the same time.

Signal	Bank	VCCO	Pin	Description
ADC1_CS#	16	3.3V	A24	ADC active low chip select
ADC1_SDIN	16		B24	ADC serial data input
ADC1_SDO	16		A25	ADC serial data output
ADC1_SCK	16		D24	ADC serial clock input
ADC1_CNV	16		B26	ADC conversion input
ADC1_BUSY	16		B25	ADC busy output
ADC1_RESET	16		D23	ADC reset input
ADC1_PD	16		C24	ADC power down input

Signal	Bank	VCCO	Pin	Description
ADC2_CS#	16	3.3V	C26	ADC active low chip select
ADC2_SDIN	15		F22	ADC serial data input
ADC2_SDO	15		E25	ADC serial data output
ADC2_SCK	15		D25	ADC serial clock input
ADC2_CNV	15		E26	ADC conversion input
ADC2_BUSY	15		D26	ADC busy output
ADC2_RESET	15		E23	ADC reset input
ADC2_PD	15		F23	ADC power down input

Signal	Bank	VCCO	Pin	Description
ADC3_CS#	15	3.3V	G22	ADC active low chip select
ADC3_SDIN	15		H22	ADC serial data input
ADC3_SDO	15		G24	ADC serial data output
ADC3_SCK	15		F25	ADC serial clock input
ADC3_CNV	15		G26	ADC conversion input
ADC3_BUSY	15		G25	ADC busy output
ADC3_RESET	15		F24	ADC reset input
ADC3_PD	15		H23	ADC power down input

Signal	Bank	VCCO	Pin	Description
ADC4_CS#	15	3.3V	H26	ADC active low chip select
ADC4_SDIN	15		H24	ADC serial data input
ADC4_SDO	15		J23	ADC serial data output
ADC4_SCK	15		J24	ADC serial clock input
ADC4_CNV	15		K22	ADC conversion input
ADC4_BUSY	15		K23	ADC busy output
ADC4_RESET	15		J26	ADC reset input
ADC4_PD	15		J25	ADC power down input

Table 7-9: ADC Interface Connections

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

For using the serial SPI interface between the User FPGA (Artix-7) and one of the four ADAS3022 ADC devices please use the ADAS3022 data sheet which describes the communication process.

7.8 Parallel DAC Interface

7.8.1 Overview

The 8 analog DAC outputs of the TXMC637 are implemented with four AD5547 16bit Dual-Current DAC devices. Each of these DACs has two DAC channels. Thus, a total of 8 DAC channels are available on the TXMC637. Because of the fact that these DACs have a current output it is necessary to use operational amplifier for each DAC output channel to generate an output voltage range up to ±10V.

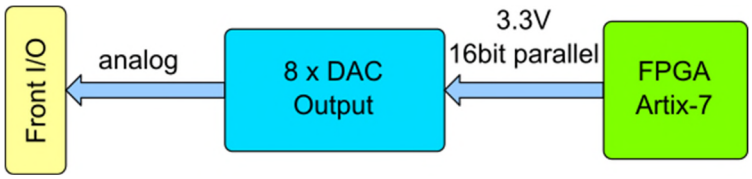


Figure 7-8 : Analog Output Section

As programming interface, a 16bit parallel Bus is implemented. Respectively all four DAC devices (8 DAC channels) share this Artix-7 16bit data bus. To set each output individually each DAC device has its control interface.

The following figure shows the structure and principle of two DAC outputs. Both are connected via independent operational amplifiers to the TXMC637 I/O Connector.

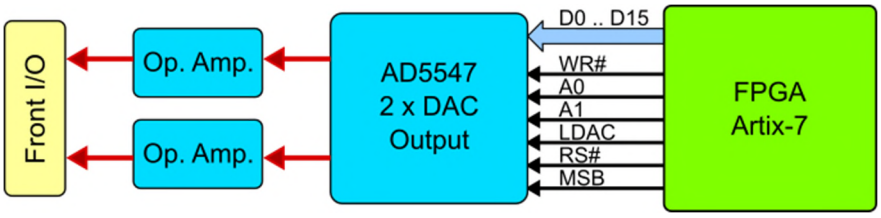


Figure 7-9 : Analog Output Section

The key-features of the TXMC637 DAC Interface are:

- 16bit Resolution
- Built-in 4-quadrant resistors in combination with an operational amplifier allow ±10V outputs
- Outputs Drive ±10mA per channel
- Capacitive Load Driving = 1000pF

7.8.2 Output Voltage Range

The output voltage ranges of the TXMC637 DAC outputs are set via BCC Register - *DAC Control / Status Register* and *DAC Output Voltage Range Register*.

There are three predefined output voltage ranges ±10V, ±5V, ±2,5V and a fourth mode in which the high and low voltage range can be set individually.

For the first three predefined fixed stress ranges, the corecture data were determined during the TEWS factory test. Determined correction values were stored in an I2C EEPROM. There are no correction values for the individually adjustable voltage range mode.

Due to tolerances of the reference voltage generation, basic tolerances of the DAC components and temperature dependence, it may happen that the limits for the output voltage cannot be reached.

7.8.3 User FPGA Pinning

First DAC configuration interface is for controlling the DAC channel 0 up to 7.

Signal	Bank	VCCO	Pin	Description
DAC_D0	16	3.3V	A23	First DAC Data Bus to DAC Device 1 .. 4 with DAC Channel 1 .. 8
DAC_D1	16	3.3V	C23	
DAC_D2	16	3.3V	A22	
DAC_D3	16	3.3V	B22	
DAC_D4	16	3.3V	C22	
DAC_D5	16	3.3V	B21	
DAC_D6	16	3.3V	C21	
DAC_D7	16	3.3V	B20	
DAC_D8	16	3.3V	A20	
DAC_D9	16	3.3V	B19	
DAC_D10	16	3.3V	A19	
DAC_D11	16	3.3V	C18	
DAC_D12	16	3.3V	D18	
DAC_D13	16	3.3V	A17	
DAC_D14	16	3.3V	B17	
DAC_D15	16	3.3V	C17	
DAC_ADR0	16	3.3V	A18	DAC Address Line to select the DAC channel A or B from one DAC Device.
DAC_'WR00_01'	35	3.3V	D4	A low active WR transfers data to DAC input register. One write signal for each DAC device respectively for two DAC channel.
DAC_'WR02_03'	15	3.3V	J16	
DAC_'WR04_05'	35	3.3V	G7	
DAC_'WR06_07'	15	3.3V	J15	
DAC_LDAC00_01	35	3.3V	C4	Load the DAC output register with contents of the input register. One write signal for each DAC device respectively for two DAC channel.
DAC_LDAC02_03	16	3.3V	G15	
DAC_LDAC04_05	35	3.3V	K6	
DAC_LDAC06_07	15	3.3V	M16	
DAC_RS#	16	3.3V	E22	Active low resets all 16 input and output DAC registers. Value depends on DAC_MSB line.
DAC_MSB	35	3.3V	J8	**MSB Power-On Reset State.** DAC_MSB = 0 corresponds to zero-scale reset; DAC_MSB = 1 corresponds to midscale reset

Table 7-10: TXCM637 parallel DAC Interface

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.8.4 Programming Hints for AD5547

TXMC637 DAC Channel write to DAC Input Register Decoding.

ADR0	'WR00_01'	'WR02_03'	'WR04_05'	'WR06_07'	DAC Channel
0	0	1	1	1	Channel 1
1	0	1	1	1	Channel 2
0	1	0	1	1	Channel 3
1	1	0	1	1	Channel 4
0	1	1	0	1	Channel 5
1	1	1	0	1	Channel 6
0	1	1	1	0	Channel 7
1	1	1	1	0	Channel 8

TXMC637 DAC Channel LOAD_DAC Register Decoding.

ADR0	'LDAC00_01'	'LDAC02_03'	'LDAC04_05'	'LDAC06_07'	DAC Channel
0	1	0	0	0	Channel 1
1	1	0	0	0	Channel 2
0	0	1	0	0	Channel 3
1	0	1	0	0	Channel 4
0	0	0	1	0	Channel 5
1	0	0	1	0	Channel 6
0	0	0	0	1	Channel 7
1	0	0	0	1	Channel 8

A detailed description of the AD5547 parallel interface and the AD5547 function please use the data sheet which describes the whole data transfer, data register and output process and all special characteristics of the DAC.

7.9 TTL I/O Interface

Each of the 32 Front I/O TTL lines are designed with a 74LVC2G241 dual buffer as an interface to the User FPGA (Artix-7) pins. The logic levels of the buffers are TTL compatible, meaning that the minimum high level is 2.0V and the maximum low level is 0.8V. The nominal output high voltage is 3.3V.

The buffer outputs are followed by 47Ω serial resistors for signal integrity and safety reasons and a 4.7kΩ pull resistors. The pull resistors guaranty a TTL compatible logic level when outputs are tristate and not driven externally.

As an option the pull up voltage can be set to 5V by an analog multiplexer to (weakly) drive a higher voltage than 3.3V by setting the output to tristate. This means, instead of changing the logical output value (high and low), the output enable is set to logic low for an output high level or logic high to use the pull functionality. The logical output value must be logic low to ensure the low-level at the output while the buffer is active (output enable is active).

For example when connecting to a standard 5V CMOS logic input (not TTL compatible levels), a high level of minimum 3.5V is required.

A second option is to set the pull voltage to GND for pull-down functionality. This means, instead of changing the logical output value (high and low), the output enable is set to logical low for an output low level or logical high to drive the output high. The logical output value must be logic high to ensure the high-level at the output while the buffer is active (output enable is active).

Please note that the pull-up or pull-down resistor can only drive high impedance inputs. In low impedance input cases there is a reduced high-level voltage at the I/O pin when the output buffer sources a noticeable current to the external load while driving a high-level. There is also an increased low-level voltage at the I/O pin when the output buffer sinks a noticeable current from the external load while driving a low-level. The maximum current needs to be limited in both cases to ensure proper signal level.

A TVS array protects against ESD shocks.

See the following figure for more information of the TTL I/O circuitry.

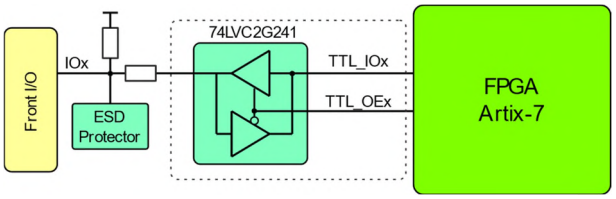


Figure 7-10 : One Channel TTL I/O Interface

Due to on placement restrictions, groups are needed for the pull voltage.

If the Pull-Resistors is set to floats (possible selection), the user should keep in mind that the I/O Lines of one group are connected via their Pull-Resistors.

7.9.1 User FPGA Pinning

Signal Name	Pin Number	VCCO	Direction
TTL_IO<0>	F5	3.3V	IN/OUT
TTL_IO<1>	G5	3.3V	IN/OUT
TTL_IO<2>	F4	3.3V	IN/OUT
TTL_IO<3>	G4	3.3V	IN/OUT
TTL_IO<4>	J4	3.3V	IN/OUT
TTL_IO<5>	F3	3.3V	IN/OUT
TTL_IO<6>	E3	3.3V	IN/OUT
TTL_IO<7>	D3	3.3V	IN/OUT
TTL_IO<8>	J6	3.3V	IN/OUT
TTL_IO<9>	H6	3.3V	IN/OUT
TTL_IO<10>	D6	3.3V	IN/OUT
TTL_IO<11>	D5	3.3V	IN/OUT
TTL_IO<12>	E5	3.3V	IN/OUT
TTL_IO<13>	F8	3.3V	IN/OUT
TTL_IO<14>	G8	3.3V	IN/OUT
TTL_IO<15>	K8	3.3V	IN/OUT
TTL_IO<16>	E18	3.3V	IN/OUT
TTL_IO<17>	F15	3.3V	IN/OUT
TTL_IO<18>	K21	3.3V	IN/OUT
TTL_IO<19>	F20	3.3V	IN/OUT
TTL_IO<20>	L8	3.3V	IN/OUT
TTL_IO<21>	A5	3.3V	IN/OUT
TTL_IO<22>	A3	3.3V	IN/OUT
TTL_IO<23>	B1	3.3V	IN/OUT
TTL_IO<24>	H16	3.3V	IN/OUT
TTL_IO<25>	H15	3.3V	IN/OUT
TTL_IO<26>	G21	3.3V	IN/OUT
TTL_IO<27>	L18	3.3V	IN/OUT
TTL_IO<28>	D20	3.3V	IN/OUT
TTL_IO<29>	E20	3.3V	IN/OUT
TTL_IO<30>	C19	3.3V	IN/OUT
TTL_IO<31>	D19	3.3V	IN/OUT
TTL_OE<0>	G1	3.3V	OUTPUT
TTL_OE<1>	E1	3.3V	OUTPUT
TTL_OE<2>	G2	3.3V	OUTPUT
TTL_OE<3>	D1	3.3V	OUTPUT
TTL_OE<4>	H4	3.3V	OUTPUT

TTL_OE<5>	E2	3.3V	OUTPUT
TTL_OE<6>	C2	3.3V	OUTPUT
TTL_OE<7>	B2	3.3V	OUTPUT
TTL_OE<8>	G6	3.3V	OUTPUT
TTL_OE<9>	E6	3.3V	OUTPUT
TTL_OE<10>	J5	3.3V	OUTPUT
TTL_OE<11>	B5	3.3V	OUTPUT
TTL_OE<12>	B4	3.3V	OUTPUT
TTL_OE<13>	K7	3.3V	OUTPUT
TTL_OE<14>	H7	3.3V	OUTPUT
TTL_OE<15>	H8	3.3V	OUTPUT
TTL_OE<16>	G17	3.3V	OUTPUT
TTL_OE<17>	L14	3.3V	OUTPUT
TTL_OE<18>	J21	3.3V	OUTPUT
TTL_OE<19>	J19	3.3V	OUTPUT
TTL_OE<20>	F7	3.3V	OUTPUT
TTL_OE<21>	A4	3.3V	OUTPUT
TTL_OE<22>	A2	3.3V	OUTPUT
TTL_OE<23>	C1	3.3V	OUTPUT
TTL_OE<24>	M17	3.3V	OUTPUT
TTL_OE<25>	J14	3.3V	OUTPUT
TTL_OE<26>	G20	3.3V	OUTPUT
TTL_OE<27>	H18	3.3V	OUTPUT
TTL_OE<28>	D16	3.3V	OUTPUT
TTL_OE<29>	H14	3.3V	OUTPUT
TTL_OE<30>	G19	3.3V	OUTPUT
TTL_OE<31>	F18	3.3V	OUTPUT

Table 7-11 : User FPGA TTL IO and OE Pins

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.10 RS422 Interface

The TXMC637 provides a multiplex functionality between TTL and RS422 for 16 of the 32 ESD-protected TTL lines so that 16 TTL can alternatively be 8 RS422 channels.

The specification for TTL interfaces and RS422 interfaces specifies very different maximum voltage levels or common mode voltages, so this switchover involves some risks.

The switching itself is done via 8 Texas Instruments TMUX1072 2-Channel 2:1 Analog Multiplexers with Overvoltage Detection and Protection. These multiplexers make simultaneous use as TTL I/O and RS422 impossible and providing an overvoltage protection to prevent the TTL buffers from being destroyed by RS422 levels.

Every Multiplexer switches two TTL pins to one differential RS422 pair.

Switching is done via the BCC Digital I/O Interface Configuration / Status Register.

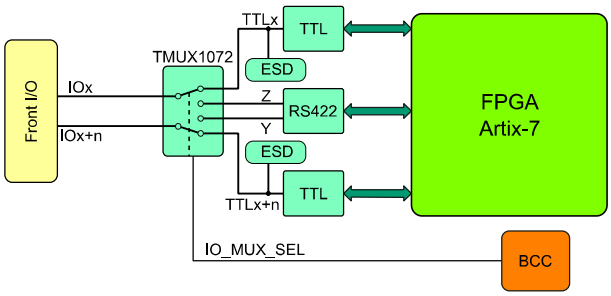


Figure 7-11 : 1 Channel Digital I/O Multiplexer Block Diagram

TTL channels 16 to 31 can be switched.

7.10.1 RS422 FPGA Interface

The Half Duplex RS485/RS422 transceiver LTC2854 from Linear technology is used an the TXMC637 for the differential I/O Interface. The transceiver is controlled via the following pins.

- RO - Receiver Output
- RE# - Receiver Enable
- DE - Driver Enable
- DI - Diver Input
- TE - Termination Enable (internal)

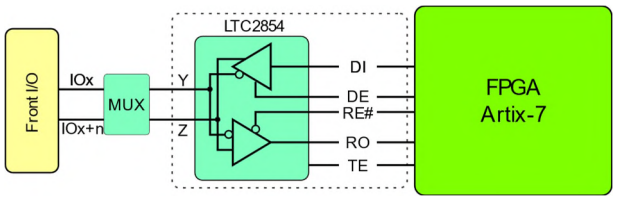


Figure 7-12 : One Channel RS422 I/O Interface

For further information, please use the LTC2854 data sheet.

7.10.2 User FPGA Pinning

Signal Name	Pin Number	VCCO	Direction
Receiver Output			
RS_RO<0>	H17	3.3V	IN
RS_RO<1>	H21	3.3V	IN
RS_RO<2>	H9	3.3V	IN
RS_RO<3>	C3	3.3V	IN
RS_RO<4>	G16	3.3V	IN
RS_RO<5>	K20	3.3V	IN
RS_RO<6>	F17	3.3V	IN
RS_RO<7>	F19	3.3V	IN
Receiver Enable			
RS_RE#<0>	Y20	2.5V	OUT
RS_RE#<1>	U25	2.5V	OUT
RS_RE#<2>	U20	2.5V	OUT
RS_RE#<3>	V19	2.5V	OUT
RS_RE#<4>	W23	2.5V	OUT
RS_RE#<5>	V26	2.5V	OUT
RS_RE#<6>	W24	2.5V	OUT
RS_RE#<7>	Y25	2.5V	OUT
Driver Enable			
RS_DE<0>	AA22	2.5V	OUT
RS_DE<1>	W25	2.5V	OUT
RS_DE<2>	T20	2.5V	OUT
RS_DE<3>	U19	2.5V	OUT
RS_DE<4>	V23	2.5V	OUT
RS_DE<5>	W26	2.5V	OUT
RS_DE<6>	V24	2.5V	OUT
RS_DE<7>	Y26	2.5V	OUT
Diver Input			
RS_DI<0>	W20	2.5V	OUT
RS_DI<1>	U24	2.5V	OUT
RS_DI<2>	T19	2.5V	OUT
RS_DI<3>	W19	2.5V	OUT
RS_DI<4>	AA23	2.5V	OUT
RS_DI<5>	U26	2.5V	OUT
RS_DI<6>	AA24	2.5V	OUT
RS_DI<7>	AA25	2.5V	OUT

Termination Enable			
RS_TE<0>	E16	3.3V	OUT
RS_TE<1>	J20	3.3V	OUT
RS_TE<2>	G9	3.3V	OUT
RS_TE<3>	F2	3.3V	OUT
RS_TE<4>	L15	3.3V	OUT
RS_TE<5>	H19	3.3V	OUT
RS_TE<6>	K15	3.3V	OUT
RS_TE<7>	J18	3.3V	OUT

Table 7-12 : User FPGA RS422 Interface Pins

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.11 I/O Pull Configuration

Each TTL I/O Line has a 4k7 Pull-Resistor. The 32 I/O Lines are arranged in two groups which can be configured for 3.3V pull-up, 5V pull-up or pull-down. In addition, the Pull-Resistors can float.

If the Pull-Resistors float, the user should keep in mind that the 16 I/O Lines of each group are connected via their Pull-Resistors to the others in the group.

The I/O Pull configuration can be done via the User FPGA or the BCC. The normal behaviour is that the User FPGA firmware controls the I/O Pull Configuration depending on the User FPGA I/O functionality.

The User FPGA IO_PULL Interface configuration signals are connected to three analog multiplexers via the BCC. With these multiplexers, the desired pull resistor reference can be adjusted directly from the User FPGA. There are four I/O lines in total for controlling the three analog multiplexer regarding the I/O Pull Voltage.

CNT **Lines**	**concerns I/O Lines**	**Description**
PULL_1_GRP [1:0]	TTL_IO[16:31]	0b11 : pull-down 0b10 : pull-up to 3.3V 0b01 : pull-up to 5V 0b00 : No pull-up or pull-down (default)
PULL_0_GRP[1:0]	TTL_IO[0 :15]	

Table 7-13: I/O Pull Configuration

Signal Name	**Pin Number**	**Direction**	**IO Standard**	**Drive [mA]**	**Slew Rate**
PULL_0_GRP<0>	P20	OUT	LVCMOS33	8	SLOW
PULL_0_GRP<1>	T22	OUT	LVCMOS33	8	SLOW
PULL_1_GRP<0>	N19	OUT	LVCMOS33	8	SLOW
PULL_1_GRP<1>	P21	OUT	LVCMOS33	8	SLOW

Table 7-14 : User FPGA Pins

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

The BCC offers an additional option for setting the I/O Pull voltage. Through its Digital I/O Interface Configuration / Status Register the User FPGA control can be revoked or rather dominated. (Use this Register to set the desired pull resistor reference.)

7.12 Memory

The TXMC637 is equipped with a 512MB, 16bit wide DDR3 SDRAM and a 128Mbit non-volatile SPI-Flash. The SPI-Flash can also be used as the User FPGA configuration memory.

7.12.1 DDR3 SDRAM

The TXMC637 provides one MT41… (96-ball) DDR3 memory devices. The memory is accessible through a Memory Interface Controller Block IP in bank 33 and 34 of the User FPGA.

Signal	FPGA Pin Number	Termination	Memory Devices	
			Pin	Name
A0	M1	49.9Ω VTT	N3	A0
A1	N1	49.9Ω VTT	P7	A1
A2	K2	49.9Ω VTT	P3	A2
A3	L3	49.9Ω VTT	N2	A3
A4	J1	49.9Ω VTT	P8	A4
A5	K1	49.9Ω VTT	P2	A5
A6	M6	49.9Ω VTT	R8	A6
A7	N6	49.9Ω VTT	R2	A7
A8	N7	49.9Ω VTT	T8	A8
A9	K5	49.9Ω VTT	R3	A9
A10	L5	49.9Ω VTT	L7	A10
A11	L7	49.9Ω VTT	R7	A11
A12	M7	49.9Ω VTT	N7	A12
A13	J3	49.9Ω VTT	T3	NC/A13
A14	K3	49.9Ω VTT	T7	NC/A14
BA0	M2	49.9Ω VTT	M2	BA0
BA1	H1	49.9Ω VTT	N8	BA1
BA2	H2	49.9Ω VTT	M3	BA2
RAS#	L2	49.9Ω VTT	J3	RAS#
CAS#	N3	49.9Ω VTT	K3	CAS#
WE#	N2	49.9Ω VTT	L3	WE#
RESET#	AA2	4.7kΩ GND	T2	RESET#
CKE[0]	P1	4.7kΩ GND	K9	CKE
ODT[0]	T4	4.7kΩ GND	K1	ODT
CS#	R1	49.9Ω VTT	L2	CS#
DM_0	P6	ODT	E7	LDM
DM_1	AB1	ODT	D3	UDM
DQ0	T5	ODT	E3	DQ0
DQ1	R5	ODT	F7	DQ1
DQ2	R8	ODT	F2	DQ2
DQ3	P8	ODT	F8	DQ3
DQ4	R7	ODT	H3	DQ4

Signal	FPGA Pin Number	Termination	Memory Devices	
			Pin	Name
DQ5	R6	ODT	H8	DQ5
DQ6	T8	ODT	G2	DQ6
DQ7	T7	ODT	H7	DQ7
DQ8	AC1	ODT	D7	DQ8
DQ9	Y2	ODT	C3	DQ9
DQ10	Y1	ODT	C8	DQ10
DQ11	AE2	ODT	C2	DQ11
DQ12	AF2	ODT	A7	DQ12
DQ13	AB2	ODT	A2	DQ13
DQ14	AC2	ODT	B8	DQ14
DQ15	AA3	ODT	A3	DQ15
CK_p	M4	100Ω	J7	CK
CK_n	L4		K7	CK#
DQS_0_p	U6	ODT	F3	LDQS
DQS_0_n	U5	ODT	G3	LDQS#
DQS_1_p	AD1	ODT	C7	UDQS
DQS_1_n	AE1	ODT	B7	UDQS#

Table 7-15 : DDR3 SDRAM Interface

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

For details regarding the DDR3 SDRAM interface, please refer to XILINX Memory Interface Generator Documentation. Xilinx UG586: *Zynq-7000 AP SoC and 7 Series Devices Memory Interface Solutions v4.2*.

7.12.2 SPI-Flash

The TXMC637 provides a Micron MT25QL128ABA 128-Mbit serial Flash memory. This Flash is used as FPGA configuration source (default configuration source).

After configuration, it is always accessible from the FPGA, so it also can be used for code or user data storage.

The SPI-EEPROM is connected via Quad (x4) SPI interface to the User FPGA (Artix-7) configuration interface.

SPI-PROM Signal	Bank	V_{CCO}	Pin	Description / Artix-7
CLK	0	3.3V	H13	Serial Clock (CCLK_B)
CS#	14	3.3V	P18	Chip Select (FCS_B)
DI (bit0)	14	3.3V	R14	Serial Data input (MOSI) / MISO[0]
DO (bit1)	14	3.3V	R15	Serial Data output (DIN) / MISO[1]
WP# (bit2)	14	3.3V	P14	MISO[2] – D02
HOLD# (bit3)	14	3.3V	N14	MISO[3] – D03

Table 7-16 : FPGA SPI-Flash Connections

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.12.3 I2C - EEPROM

The TXMC637 provides an STMicroelectronics M24128 (128-Kbit) I2C-Compatible (2-wire) Serial EEPROM.

This EEPROM is used as ADC and DAC correction data memory. In addition, for each correction data set ADC Input and DAC Output the corresponding voltage range of ADC inputs and DAC outputs are also stored.

During factory test the analog input and output channel gain error and offset error are determined. For each device, 16bit correction values are stored to the I2C EEPROM. These correction values have been determined with TEWS test environment. If system specific correction values are needed, the determination of the correcture values of the entire system can be done by the user and the I2C EEPROM could be used as a possible memory to store the custom correction values.

The I2C EEPROM is connected via 2-wire interface to User FPGA (Artix-7). As usual for the I2C interface the two pins must be realized as open drain buffers.

SPI-PROM Signal	Bank	V_{CCO}	Pin	Description / Artix-7
USER_SCL	14	3.3V	R25	Serial clock
USER_SDA	14	3.3V	P24	Serial data

Table 7-17: FPGA I2C EEPROM Connections

For using the serial I2C interface between the USER FPGA (Artix-7) and the I2C EEPROM please see the STMicroelectronics M24128 data sheet which describes the serial communication process.

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.12.3.1 ADC and DAC Correctable Errors

There are two errors affecting the accuracy of the ADC and DAC that can be corrected using the factory determined correction values. The correction values are obtained during factory test and are stored in an on-board I2C EEPROM as 2-complement 16bit values in the range from -32768 to +32767. To achieve a higher accuracy, they are scaled to ¼LSB.

DAC Offset Error:

For the DAC, this is the data value that is required to produce a zero voltage output signal. This error is corrected by subtracting the offset from the DAC data value.

ADC Offset Error:

The offset error is the data value when converting with the input connected to its own ground in single-ended mode, or with shorted inputs in differential mode. This error is corrected by subtracting the known error from the reading.

DAC Gain Error:

The gain error is the difference between the ideal gain and the actual gain of the DAC. It is corrected by multiplying the DAC data value with a correction factor.

ADC Gain Error:

The gain error is the difference between the ideal gain and the actual gain of the programmable gain amplifier and the ADC. This error is corrected by multiplying the reading with a correction factor.

7.12.3.2 ADC Correction Values

The 16 ADC channels are realized with four octal ADAS3022 ADC devices.

For each ADC channel, each gain, differential and single ended mode a 16bit Offset correction and a 16bit gain correction value is stored.

Exceptions are the two single-ended mode ranges (Gain 0.2 / ±20.48 and Gain 0.16666 / ±24.576).

These two voltage ranges are limited by the "architecture" of the ADAS3022 to half of the digital values. In addition, the maximum range is limited by the Supply Voltage. Since these ranges are almost indistinguishable from the correction values, only the values for range Gain 0.16666 were measured. But correction values can be used for both voltage ranges.

I2C EEPROM Address	Description	Size (Bit)
ADC Differential Mode A/D Channel Gain 0.1666 / ±24.576V Range		
0x000	ADC 1 Differential Ch. 0, Gain 0.1666, $Offset_{Corr}$ High Byte	8
0x001	ADC 1 Differential Ch. 0, Gain 0.1666, $Offset_{Corr}$ Low Byte	8
0x002	ADC 1 Differential Ch. 0, Gain 0.1666, $Gain_{Corr}$ High Byte	8
0x003	ADC 1 Differential Ch. 0, Gain 0.1666, $Gain_{Corr}$ Low Byte	8
0x004	ADC 1 Differential Ch. 1, Gain 0.1666, $Offset_{Corr}$ High Byte	8
0x005	ADC 1 Differential Ch. 1, Gain 0.1666, $Offset_{Corr}$ Low Byte	8
0x006	ADC 1 Differential Ch. 1, Gain 0.1666, $Gain_{Corr}$ High Byte	8
0x007	ADC 1 Differential Ch. 1, Gain 0.1666, $Gain_{Corr}$ Low Byte	8
. . .		
0x03C	ADC 4 Differential Ch. 3, Gain 0.1666, $Offset_{Corr}$ High Byte	8

0x03D	ADC 4 Differential Ch. 3, Gain 0.1666, $Offset_{Corr}$ Low Byte	8
0x03E	ADC 4 Differential Ch. 3, Gain 0.1666, $Gain_{Corr}$ High Byte	8
0x03F	ADC 4 Differential Ch. 3, Gain 0.1666, $Gain_{Corr}$ Low Byte	8
0x040 .. 0x07E	Reserved	512
ADC Differential Mode A/D Channel Gain 0.2 / ±20.48V Range		
0x080	ADC 1 Differential Ch. 0, Gain 0.2, $Offset_{Corr}$ High Byte	8
0x081	ADC 1 Differential Ch. 0, Gain 0.2, $Offset_{Corr}$ Low Byte	8
0x082	ADC 1 Differential Ch. 0, Gain 0.2, $Gain_{Corr}$ High Byte	8
0x083	ADC 1 Differential Ch. 0, Gain 0.2, $Gain_{Corr}$ Low Byte	8
0x084	ADC 1 Differential Ch. 1, Gain 0.2, $Offset_{Corr}$ High Byte	8
0x085	ADC 1 Differential Ch. 1, Gain 0.2, $Offset_{Corr}$ Low Byte	8
0x086	ADC 1 Differential Ch. 1, Gain 0.2, $Gain_{Corr}$ High Byte	8
0x087	ADC 1 Differential Ch. 1, Gain 0.2, $Gain_{Corr}$ Low Byte	8
…		
0x0BC	ADC 4 Differential Ch. 3, Gain 0.2, $Offset_{Corr}$ High Byte	8
0xBD	ADC 4 Differential Ch. 3, Gain 0.2, $Offset_{Corr}$ Low Byte	8
0x0BE	ADC 4 Differential Ch. 3, Gain 0.2, $Gain_{Corr}$ High Byte	8
0x0BF	ADC 4 Differential Ch. 3, Gain 0.2, $Gain_{Corr}$ Low Byte	8
0x0C0 .. 0x0FE	Reserved	512
ADC Differential Mode A/D Channel Gain 0.4 / ±10.24V Range		
0x100	ADC 1 Differential Ch. 0, Gain 0.4, $Offset_{Corr}$ High Byte	8
0x101	ADC 1 Differential Ch. 0, Gain 0.4, $Offset_{Corr}$ Low Byte	8
0x102	ADC 1 Differential Ch. 0, Gain 0.4, $Gain_{Corr}$ High Byte	8
0x103	ADC 1 Differential Ch. 0, Gain 0.4, $Gain_{Corr}$ Low Byte	8
0x104	ADC 1 Differential Ch. 1, Gain 0.4, $Offset_{Corr}$ High Byte	8
0x105	ADC 1 Differential Ch. 1, Gain 0.4, $Offset_{Corr}$ Low Byte	8
0x106	ADC 1 Differential Ch. 1, Gain 0.4, $Gain_{Corr}$ High Byte	8
0x107	ADC 1 Differential Ch. 1, Gain 0.4, $Gain_{Corr}$ Low Byte	8
…		
0x13C	ADC 4 Differential Ch. 3, Gain 0.4, $Offset_{Corr}$ High Byte	8
0x13D	ADC 4 Differential Ch. 3, Gain 0.4, $Offset_{Corr}$ Low Byte	8
0x13E	ADC 4 Differential Ch. 3, Gain 0.4, $Gain_{Corr}$ High Byte	8
0x13F	ADC 4 Differential Ch. 3, Gain 0.4, $Gain_{Corr}$ Low Byte	8
0x140 .. 0x17E	Reserved	512
ADC Differential Mode A/D Channel Gain 0.8 / ±5.12V Range		
0x180	ADC 1 Differential Ch. 0, Gain 0.8, $Offset_{Corr}$ High Byte	8
0x181	ADC 1 Differential Ch. 0, Gain 0.8, $Offset_{Corr}$ Low Byte	8
0x182	ADC 1 Differential Ch. 0, Gain 0.8, $Gain_{Corr}$ High Byte	8
0x183	ADC 1 Differential Ch. 0, Gain 0.8, $Gain_{Corr}$ Low Byte	8
0x184	ADC 1 Differential Ch. 1, Gain 0.8, $Offset_{Corr}$ High Byte	8
0x185	ADC 1 Differential Ch. 1, Gain 0.8, $Offset_{Corr}$ Low Byte	8

0x186	ADC 1 Differential Ch. 1, Gain 0.8, $Gain_{Corr}$ High Byte	8
0x187	ADC 1 Differential Ch. 1, Gain 0.8, $Gain_{Corr}$ Low Byte	8
…		
0x1BC	ADC 4 Differential Ch. 3, Gain 0.8, $Offset_{Corr}$ High Byte	8
0x1BD	ADC 4 Differential Ch. 3, Gain 0.8, $Offset_{Corr}$ Low Byte	8
0x1BE	ADC 4 Differential Ch. 3, Gain 0.8, $Gain_{Corr}$ High Byte	8
0x1BF	ADC 4 Differential Ch. 3, Gain 0.8, $Gain_{Corr}$ Low Byte	8
0x1C0 .. 0x1FE	Reserved	512
ADC Differential Mode A/D Channel Gain / 1.6 / ±2.56V Range		
0x200	ADC 1 Differential Ch. 0, Gain 1.6, $Offset_{Corr}$ High Byte	8
0x201	ADC 1 Differential Ch. 0, Gain 1.6, $Offset_{Corr}$ Low Byte	8
0x202	ADC 1 Differential Ch. 0, Gain 1.6, $Gain_{Corr}$ High Byte	8
0x203	ADC 1 Differential Ch. 0, Gain 1.6, $Gain_{Corr}$ Low Byte	8
0x204	ADC 1 Differential Ch. 1, Gain 1.6, $Offset_{Corr}$ High Byte	8
0x205	ADC 1 Differential Ch. 1, Gain 1.6, $Offset_{Corr}$ Low Byte	8
0x206	ADC 1 Differential Ch. 1, Gain 1.6, $Gain_{Corr}$ High Byte	8
0x207	ADC 1 Differential Ch. 1, Gain 1.6, $Gain_{Corr}$ Low Byte	8
0x23C	ADC 4 Differential Ch. 3, Gain 1.6, $Offset_{Corr}$ High Byte	8
0x23D	ADC 4 Differential Ch. 3, Gain 1.6, $Offset_{Corr}$ Low Byte	8
0x23E	ADC 4 Differential Ch. 3, Gain 1.6, $Gain_{Corr}$ High Byte	8
0x23F	ADC 4 Differential Ch. 3, Gain 1.6, $Gain_{Corr}$ Low Byte	8
0x240 .. 0x27E	Reserved	512
ADC Differential Mode A/D Channel Gain 3.2 / ±1.28V Range		
0x280	ADC 1 Differential Ch. 0, Gain 3.2, $Offset_{Corr}$ High Byte	8
0x281	ADC 1 Differential Ch. 0, Gain 3.2, $Offset_{Corr}$ Low Byte	8
0x282	ADC 1 Differential Ch. 0, Gain 3.2, $Gain_{Corr}$ High Byte	8
0x283	ADC 1 Differential Ch. 0, Gain 3.2, $Gain_{Corr}$ Low Byte	8
0x284	ADC 1 Differential Ch. 1, Gain 3.2, $Offset_{Corr}$ High Byte	8
0x285	ADC 1 Differential Ch. 1, Gain 3.2, $Offset_{Corr}$ Low Byte	8
0x286	ADC 1 Differential Ch. 1, Gain 3.2, $Gain_{Corr}$ High Byte	8
0x287	ADC 1 Differential Ch. 1, Gain 3.2, $Gain_{Corr}$ Low Byte	8
…		
0x2BC	ADC 4 Differential Ch. 3, Gain 3.2, $Offset_{Corr}$ High Byte	8
0x2BD	ADC 4 Differential Ch. 3, Gain 3.2, $Offset_{Corr}$ Low Byte	8
0x2BE	ADC 4 Differential Ch. 3, Gain 3.2, $Gain_{Corr}$ High Byte	8
0x2BF	ADC 4 Differential Ch. 3, Gain 3.2, $Gain_{Corr}$ Low Byte	8
0x2C0 .. 0x2FE	Reserved	512
ADC Differential Mode A/D Channel Gain 6.4 / ±0.64V Range		
0x300	ADC 1 Differential Ch. 0, Gain 6.4, $Offset_{Corr}$ High Byte	8
0x301	ADC 1 Differential Ch. 0, Gain 6.4, $Offset_{Corr}$ Low Byte	8

0x302	ADC 1 Differential Ch. 0, Gain 6.4, $Gain_{Corr}$ High Byte	8
0x303	ADC 1 Differential Ch. 0, Gain 6.4, $Gain_{Corr}$ Low Byte	8
0x304	ADC 1 Differential Ch. 1, Gain 6.4, $Offset_{Corr}$ High Byte	8
0x305	ADC 1 Differential Ch. 1, Gain 6.4, $Offset_{Corr}$ Low Byte	8
0x306	ADC 1 Differential Ch. 1, Gain 6.4, $Gain_{Corr}$ High Byte	8
0x307	ADC 1 Differential Ch. 1, Gain 6.4, $Gain_{Corr}$ Low Byte	8
0x33C	ADC 4 Differential Ch. 3, Gain 6.4, $Offset_{Corr}$ High Byte	8
0x33D	ADC 4 Differential Ch. 3, Gain 6.4, $Offset_{Corr}$ Low Byte	8
0x33E	ADC 4 Differential Ch. 3, Gain 6.4, $Gain_{Corr}$ High Byte	8
0x33F	ADC 4 Differential Ch. 3, Gain 6.4, $Gain_{Corr}$ Low Byte	8
0x340 .. 0x37E	Reserved	512
ADC Single-Ended Mode A/D Channel Gain 0.1666 (half resolution) ±12.288V Range		
0x380	ADC 1 Single-Ended Ch. 0, Gain 0.1666, $Offset_{Corr}$ High Byte	8
0x381	ADC 1 Single-Ended Ch. 0, Gain 0.1666, $Offset_{Corr}$ Low Byte	8
0x382	ADC 1 Single-Ended Ch. 0, Gain 0.1666, $Gain_{Corr}$ High Byte	8
0x383	ADC 1 Single-Ended Ch. 0, Gain 0.1666, $Gain_{Corr}$ Low Byte	8
0x384	ADC 1 Single-Ended Ch. 1, Gain 0.1666, $Offset_{Corr}$ High Byte	8
0x385	ADC 1 Single-Ended Ch. 1, Gain 0.1666, $Offset_{Corr}$ Low Byte	8
0x386	ADC 1 Single-Ended Ch. 1, Gain 0.1666, $Gain_{Corr}$ High Byte	8
0x387	ADC 1 Single-Ended Ch. 1, Gain 0.1666, $Gain_{Corr}$ Low Byte	8
0x3FC	ADC 4 Single-Ended Ch. 7, Gain 0.1666, $Offset_{Corr}$ High Byte	8
0x3FD	ADC 4 Single-Ended Ch. 7, Gain 0.1666, $Offset_{Corr}$ Low Byte	8
0x3FE	ADC 4 Single-Ended Ch. 7, Gain 0.1666, $Gain_{Corr}$ High Byte	8
0x3FF	ADC 4 Single-Ended Ch. 7, Gain 0.1666, $Gain_{Corr}$ Low Byte	8
ADC Single-Ended Mode A/D Channel Gain 0.4 / ±10.24V Range		
0x400	ADC 1 Single-Ended Ch. 0, Gain 0.4, $Offset_{Corr}$ High Byte	8
0x401	ADC 1 Single-Ended Ch. 0, Gain 0.4, $Offset_{Corr}$ Low Byte	8
0x402	ADC 1 Single-Ended Ch. 0, Gain 0.4, $Gain_{Corr}$ High Byte	8
0x403	ADC 1 Single-Ended Ch. 0, Gain 0.4, $Gain_{Corr}$ Low Byte	8
0x404	ADC 1 Single-Ended Ch. 1, Gain 0.4, $Offset_{Corr}$ High Byte	8
0x405	ADC 1 Single-Ended Ch. 1, Gain 0.4, $Offset_{Corr}$ Low Byte	8
0x406	ADC 1 Single-Ended Ch. 1, Gain 0.4, $Gain_{Corr}$ High Byte	8
0x507	ADC 1 Single-Ended Ch. 1, Gain 0.4, $Gain_{Corr}$ Low Byte	8
…		
0x47C	ADC 4 Single-Ended Ch. 7, Gain 0.4, $Offset_{Corr}$ High Byte	8
0x47D	ADC 4 Single-Ended Ch. 7, Gain 0.4, $Offset_{Corr}$ Low Byte	8
0x47E	ADC 4 Single-Ended Ch. 7, Gain 0.4, $Gain_{Corr}$ High Byte	8
0x47F	ADC 4 Single-Ended Ch. 7, Gain 0.4, $Gain_{Corr}$ Low Byte	8

ADC Single-Ended Mode A/D Channel Gain 0.8 / ±5.12V Range		
0x480	ADC 1 Single-Ended Ch. 0, Gain 0.8, $Offset_{Corr}$ High Byte	8
0x481	ADC 1 Single-Ended Ch. 0, Gain 0.8, $Offset_{Corr}$ Low Byte	8
0x482	ADC 1 Single-Ended Ch. 0, Gain 0.8, $Gain_{Corr}$ High Byte	8
0x483	ADC 1 Single-Ended Ch. 0, Gain 0.8, $Gain_{Corr}$ Low Byte	8
0x484	ADC 1 Single-Ended Ch. 1, Gain 0.8, $Offset_{Corr}$ High Byte	8
0x485	ADC 1 Single-Ended Ch. 1, Gain 0.8, $Offset_{Corr}$ Low Byte	8
0x486	ADC 1 Single-Ended Ch. 1, Gain 0.8, $Gain_{Corr}$ High Byte	8
0x487	ADC 1 Single-Ended Ch. 1, Gain 0.8, $Gain_{Corr}$ Low Byte	8
…		
0x4FC	ADC 4 Single-Ended Ch. 7, Gain 0.8, $Offset_{Corr}$ High Byte	8
0x4FD	ADC 4 Single-Ended Ch. 7, Gain 0.8, $Offset_{Corr}$ Low Byte	8
0x4FE	ADC 4 Single-Ended Ch. 7, Gain 0.8, $Gain_{Corr}$ High Byte	8
0x4FF	ADC 4 Single-Ended Ch. 7, Gain 0.8, $Gain_{Corr}$ Low Byte	8
ADC Single-Ended Mode A/D Channel Gain 1.6 / ±2.56V Range		
0x500	ADC 1 Single-Ended Ch. 0, Gain 1.6, $Offset_{Corr}$ High Byte	8
0x501	ADC 1 Single-Ended Ch. 0, Gain 1.6, $Offset_{Corr}$ Low Byte	8
0x502	ADC 1 Single-Ended Ch. 0, Gain 1.6, $Gain_{Corr}$ High Byte	8
0x503	ADC 1 Single-Ended Ch. 0, Gain 1.6, $Gain_{Corr}$ Low Byte	8
0x504	ADC 1 Single-Ended Ch. 1, Gain 1.6, $Offset_{Corr}$ High Byte	8
0x505	ADC 1 Single-Ended Ch. 1, Gain 1.6, $Offset_{Corr}$ Low Byte	8
0x506	ADC 1 Single-Ended Ch. 1, Gain 1.6, $Gain_{Corr}$ High Byte	8
0x507	ADC 1 Single-Ended Ch. 1, Gain 1.6, $Gain_{Corr}$ Low Byte	8
…		
0x57C	ADC 4 Single-Ended Ch. 7, Gain 1.6, $Offset_{Corr}$ High Byte	8
0x57D	ADC 4 Single-Ended Ch. 7, Gain 1.6, $Offset_{Corr}$ Low Byte	8
0x57E	ADC 4 Single-Ended Ch. 7, Gain 1.6, $Gain_{Corr}$ High Byte	8
0x57F	ADC 4 Single-Ended Ch. 7, Gain 1.6, $Gain_{Corr}$ Low Byte	8
ADC Single-Ended Mode A/D Channel Gain 3.2 / ±1.28V Range		
0x580	ADC 1 Single-Ended Ch. 0, Gain 3.2, $Offset_{Corr}$ High Byte	8
0x581	ADC 1 Single-Ended Ch. 0, Gain 3.2, $Offset_{Corr}$ Low Byte	8
0x582	ADC 1 Single-Ended Ch. 0, Gain 3.2, $Gain_{Corr}$ High Byte	8
0x583	ADC 1 Single-Ended Ch. 0, Gain 3.2, $Gain_{Corr}$ Low Byte	8
0x584	ADC 1 Single-Ended Ch. 1, Gain 3.2, $Offset_{Corr}$ High Byte	8
0x585	ADC 1 Single-Ended Ch. 1, Gain 3.2, $Offset_{Corr}$ Low Byte	8
0x586	ADC 1 Single-Ended Ch. 1, Gain 3.2, $Gain_{Corr}$ High Byte	8
0x587	ADC 1 Single-Ended Ch. 1, Gain 3.2, $Gain_{Corr}$ Low Byte	8
…		
0x5FC	ADC 4 Single-Ended Ch. 7, Gain 3.2, $Offset_{Corr}$ High Byte	8
0x5FD	ADC 4 Single-Ended Ch. 7, Gain 3.2, $Offset_{Corr}$ Low Byte	8
0x5FE	ADC 4 Single-Ended Ch. 7, Gain 3.2, $Gain_{Corr}$ High Byte	8

0x5FF	ADC 4 Single-Ended Ch. 7, Gain 3.2, $Gain_{Corr}$ Low Byte	8
ADC Single-Ended Mode A/D Channel Gain 6.4 / ±0.64V Range		
0x600	ADC 1 Single-Ended Ch. 0, Gain 6.4, $Offset_{Corr}$ High Byte	8
0x601	ADC 1 Single-Ended Ch. 0, Gain 6.4, $Offset_{Corr}$ Low Byte	8
0x602	ADC 1 Single-Ended Ch. 0, Gain 6.4, $Gain_{Corr}$ High Byte	8
0x603	ADC 1 Single-Ended Ch. 0, Gain 6.4, $Gain_{Corr}$ Low Byte	8
0x604	ADC 1 Single-Ended Ch. 1, Gain 6.4, $Offset_{Corr}$ High Byte	8
0x605	ADC 1 Single-Ended Ch. 1, Gain 6.4, $Offset_{Corr}$ Low Byte	8
0x606	ADC 1 Single-Ended Ch. 1, Gain 6.4, $Gain_{Corr}$ High Byte	8
0x607	ADC 1 Single-Ended Ch. 1, Gain 6.4, $Gain_{Corr}$ Low Byte	8
…		
0x67C	ADC 4 Single-Ended Ch. 7, Gain 6.4, $Offset_{Corr}$ High Byte	8
0x67D	ADC 4 Single-Ended Ch. 7, Gain 6.4, $Offset_{Corr}$ Low Byte	8
0x67E	ADC 4 Single-Ended Ch. 7, Gain 6.4, $Gain_{Corr}$ High Byte	8
0x67F	ADC 4 Single-Ended Ch. 7, Gain 6.4, $Gain_{Corr}$ Low Byte	8

Table 7-18: ADC Correction Values

7.12.3.3 DAC Correction Values

The 16 DAC channels are realized with eight dual AD5547 DAC devices.

For each DAC channel three correction value sets are stored. One set for each DAC Output Voltage Range Selection (±10V range (default value), ±5V range and ±2.5V range).

Each correction value set consists of a 16bit value for Offset correction and Gain correction for each channel.

I2C EEPROM Address	Description	Size (Bit)
0x680	DAC Range 0 (±10V) Channel 0 $Offset_{corr}$ High Byte	8
0x681	DAC Range 0 (±10V) Channel 0 $Offset_{corr}$ Low Byte	8
0x682	DAC Range 0 (±10V) Channel 0 $Gain_{corr}$ High Byte	8
0x683	DAC Range 0 (±10V) Channel 0 $Gain_{corr}$ Low Byte	8
0x684	DAC Range 0 (±10V) Channel 1 $Offset_{corr}$ High Byte	8
0x685	DAC Range 0 (±10V) Channel 1 $Offset_{co}$ Low Byte	8
0x686	DAC Range 0 (±10V) Channel 1 $Gain_{corr}$ High Byte	8
0x0687	DAC Range 0 (±10V) Channel 1 $Gain_{corr}$ Low Byte	8
…		
0x69C	DAC Range 0 (±10V) Channel 7 $Offset_{corr}$ High Byte	8
0x69E	DAC Range 0 (±10V) Channel 7 $Offset_{corr}$ Low Byte	8
0x69E	DAC Range 0 (±10V) Channel 7 $Gain_{corr}$ High Byte	8
0x69F	DAC Range 0 (±10V) Channel 7 $Gain_{corr}$ Low Byte	8
0x6A0 .. 0x6BE	Reserved	256
0x6C0	DAC Range 1 (±5V) Channel 0 $Offset_{corr}$ High Byte	8
0x6C1	DAC Range 1 (±5V) Channel 0 $Offset_{corr}$ Low Byte	8
0x6C2	DAC Range 1 (±5V) Channel 0 $Gain_{corr}$ High Byte	8
0x6C3	DAC Range 1 (±5V) Channel 0 $Gain_{corr}$ Low Byte	8
0x6C4	DAC Range 1 (±5V) Channel 1 $Offset_{corr}$ High Byte	8
0x6C5	DAC Range 1 (±5V) Channel 1 $Offset_{corr}$ Low Byte	8
0x6C6	DAC Range 1 (±5V) Channel 1 $Gain_{corr}$ High Byte	8
0x6C7	DAC Range 1 (±5V) Channel 1 $Gain_{corr}$ Low Byte	8
…		
0x6DC	DAC Range 1 (±5V) Channel 7 $Offset_{corr}$ High Byte	8
0x6CD	DAC Range 1 (±5V) Channel 7 $Offset_{corr}$ Low Byte	8
0x6DE	DAC Range 1 (±5V) Channel 7 $Gain_{corr}$ High Byte	8
0x6CF	DAC Range 1 (±5V) Channel 7 $Gain_{corr}$ Low Byte	8
0x6E0 .. 0x6FE	Reserved	256
0x700	DAC Range 2 (±2.5V) Channel 0 $Offset_{corr}$ High Byte	8
0x701	DAC Range 2 (±2.5V) Channel 0 $Offset_{corr}$ Low Byte	8
0x702	DAC Range 2 (±2.5V) Channel 0 $Gain_{corr}$ High Byte	8
0x703	DAC Range 2 (±2.5V) Channel 0 $Gain_{corr}$ Low Byte	8
0x704	DAC Range 2 (±2.5V) Channel 1 $Offset_{corr}$ High Byte	8

0x705	DAC Range 2 (±2.5V) Channel 1 $Offset_{corr}$ Low Byte	8
0x706	DAC Range 2 (±2.5V) Channel 1 $Gain_{corr}$ High Byte	8
0x707	DAC Range 2 (±2.5V) Channel 1 $Gain_{corr}$ Low Byte	8
…		
0x71C	DAC Range 2 (±2.5V) Channel 7 $Offset_{corr}$ High Byte	8
0x71D	DAC Range 2 (±2.5V) Channel 7 $Offset_{corr}$ Low Byte	8
0x71E	DAC Range 2 (±2.5V) Channel 7 $Offset_{corr}$ High Byte	8
0x71F	DAC Range 2 (±2.5V) Channel 7 $Offset_{corr}$ Low Byte	8
0x720 .. 0x73E	Reserved	256

Table 7-19: DAC Correction Values

7.12.3.4 ADC Data Correction Formula

Please use the total 16 bit data register value for the ADC correction formula.

The basic formula for correcting any ADC reading for the TXMC627 (bipolar input voltage range) is:

$$Value = Reading \cdot \left(1 - \frac{Gain_{corr}}{131072}\right) - \frac{Offset_{corr}}{4}$$

Value is the corrected result.

Reading is the data read from the ADC Data Register.

$Gain_{corr}$ and $Offset_{corr}$ are the ADC correction factors from the Correction Data ROM stored for each programmable gain factor.

The correction values are stored as two's complement 16 bit values in the range -32768 to 32767. For higher accuracy they are scaled to ¼ LSB.

Floating point arithmetic or scaled integer arithmetic is necessary to avoid rounding error while computing above formula.

7.12.3.5 DAC Data Correction Formula

The basic formula for correcting any DAC value is:

$$Data = Value \cdot \left(1 - \frac{Gain_{corr}}{131072}\right) - \frac{Offset_{corr}}{4}$$

Value is the desired DAC value.

Data is the corrected DAC value that must be sending to the DAC.

$Gain_{corr}$ and $Offset_{corr}$ are the DAC correction values from the Correction Data ROM. They are stored separately for each of the 8 DAC channels.

The correction values are stored as two's complement byte wide values in the range from –32768 to +32767. For higher accuracy they are scaled to ¼ LSB.

Floating point arithmetic or scaled integer arithmetic must be used to avoid rounding errors in computing above formula.

7.12.3.6 DAC voltage ranges

The voltage range for each DAC output is composed of two bytes, one high byte and one low byte. Together they provide the 16bit voltage range in mV.

I2C EEPROM Address	Description	Size (Bit)
0x740	DAC Output Range 0, Channel 0 High Byte	8
0x741	DAC Output Range 0, Channel 0 Low Byte	8
0x742	DAC Output Range 0, Channel 1 High Byte	8
0c743	DAC Output Range 0, Channel 1 Low Byte	8
…		
0x74E	DAC Output Range 0, Channel 7 High Byte	8
0c74F	DAC Output Range 0, Channel 7 Low Byte	8
0x750 .. 0x75E	Reserved	
0x760	DAC Output Range 1, Channel 0 High Byte	8
0x761	DAC Output Range 1, Channel 0 Low Byte	8
0x762	DAC Output Range 1, Channel 1 High Byte	8
0x763	DAC Output Range 1, Channel 1 Low Byte	8
…		
0x76E	DAC Output Range 1, Channel 7 High Byte	8
0x76F	DAC Output Range 1, Channel 7 Low Byte	8
0x770 .. 0x77E	Reserved	128
0x780	DAC Output Range 2, Channel 0 High Byte	8
0x781	DAC Output Range 2, Channel 0 Low Byte	8
0x782	DAC Output Range 2, Channel 1 High Byte	8
0x783	DAC Output Range 2, Channel 1 Low Byte	8
…		
0x78E	DAC Output Range 2, Channel 7 High Byte	8
0x78F	DAC Output Range 2, Channel 7 Low Byte	8
0x790 .. 0x79E	reserved	128

Table 7-20: ADC and DAC voltage ranges

7.12.3.7 Version of EEPROM data structure

In the first versions of TXMC637 only the Correction data were stored. A version register is used to identify whether the voltage ranges are also present.

I2C EEPROM Address	Description	Size (Bit)
0x7A0 .. 0x7FE	not used	-
0x7FF	EEPROM data structure Version 0x00: Only the ADC and DAC correction data are available. 0x01: ADC and DAC correction data and voltage range data are available.	8

Table 7-21: Version of EEPROM data structure

7.13 Rear I/O Interface

The Rear I/O Pins of the TXMC637 are directly routed to the User FPGA (Artix-7). The I/O functions of these FPGA pins are directly dependent on the configuration of the FPGA.

The Artix-7 VCCO voltage is set to 2.5V, so only the 2.5V I/O standards LVCMOS25, LVTTL25 and LVDS_25 are possible when using the TXMC637 rear I/O interface.

Signal Name	Pin Number	Direction	IO Standard for BRD
REAR_IO0+	U22	IN/OUT	LVDS_25
REAR_IO0-	V22	IN/OUT	LVDS_25
REAR_IO1+	Y22	IN/OUT	LVDS_25
REAR_IO1-	Y23	IN/OUT	LVDS_25
REAR_IO2+	U14	IN/OUT	LVDS_25
REAR_IO2-	V14	IN/OUT	LVDS_25
REAR_IO3+	T14	IN/OUT	LVDS_25
REAR_IO3-	T15	IN/OUT	LVDS_25
REAR_IO4+	U15	IN/OUT	LVDS_25
REAR_IO4-	U16	IN/OUT	LVDS_25
REAR_IO5+	V16	IN/OUT	LVDS_25
REAR_IO5-	V17	IN/OUT	LVDS_25
REAR_IO6+	T17	IN/OUT	LVDS_25
REAR_IO6-	T18	IN/OUT	LVDS_25
REAR_IO7+	V18	IN/OUT	LVDS_25
REAR_IO7-	W18	IN/OUT	LVDS_25
REAR_IO8+	W14	IN/OUT	LVDS_25
REAR_IO8-	W15	IN/OUT	LVDS_25
REAR_IO9+	Y15	IN/OUT	LVDS_25
REAR_IO9-	AA15	IN/OUT	LVDS_25
REAR_IO10+	AB16	IN/OUT	LVDS_25
REAR_IO10-	AC16	IN/OUT	LVDS_25
REAR_IO11+	Y16	IN/OUT	LVDS_25
REAR_IO11-	Y17	IN/OUT	LVDS_25
REAR_IO12+	AE17	IN/OUT	LVDS_25
REAR_IO12-	AF17	IN/OUT	LVDS_25
REAR_IO13+	AC17	IN/OUT	LVDS_25
REAR_IO13-	AD17	IN/OUT	LVDS_25
REAR_IO14+	AE18	IN/OUT	LVDS_25
REAR_IO14-	AF18	IN/OUT	LVDS_25
REAR_IO15+	AA17	IN/OUT	LVDS_25
REAR_IO15-	AB17	IN/OUT	LVDS_25

REAR_IO16+	AC18	IN/OUT	LVDS_25
REAR_IO16-	AD18	IN/OUT	LVDS_25
REAR_IO17+	AF19	IN/OUT	LVDS_25
REAR_IO17-	AF20	IN/OUT	LVDS_25
REAR_IO18+	Y18	IN/OUT	LVDS_25
REAR_IO18-	AA18	IN/OUT	LVDS_25
REAR_IO19+	AC19	IN/OUT	LVDS_25
REAR_IO19-	AD19	IN/OUT	LVDS_25
REAR_IO20+	AA19	IN/OUT	LVDS_25
REAR_IO20-	AB19	IN/OUT	LVDS_25
REAR_IO21+	AD20	IN/OUT	LVDS_25
REAR_IO21-	AE20	IN/OUT	LVDS_25
REAR_IO22+	AD21	IN/OUT	LVDS_25
REAR_IO22-	AE21	IN/OUT	LVDS_25
REAR_IO23+	AA20	IN/OUT	LVDS_25
REAR_IO23-	AB20	IN/OUT	LVDS_25
REAR_IO24+	AE22	IN/OUT	LVDS_25
REAR_IO24-	AF22	IN/OUT	LVDS_25
REAR_IO25+	AB21	IN/OUT	LVDS_25
REAR_IO25-	AC21	IN/OUT	LVDS_25
REAR_IO26+	AE23	IN/OUT	LVDS_25
REAR_IO26-	AF23	IN/OUT	LVDS_25
REAR_IO27+	AC22	IN/OUT	LVDS_25
REAR_IO27-	AC23	IN/OUT	LVDS_25
REAR_IO28+	AD23	IN/OUT	LVDS_25
REAR_IO28-	AD24	IN/OUT	LVDS_25
REAR_IO29+	AF24	IN/OUT	LVDS_25
REAR_IO29-	AF25	IN/OUT	LVDS_25
REAR_IO30+	AE25	IN/OUT	LVDS_25
REAR_IO30-	AE26	IN/OUT	LVDS_25
REAR_IO31+	AD25	IN/OUT	LVDS_25
REAR_IO31-	AD26	IN/OUT	LVDS_25

Table 7-22 : Digital Rear I/O Interface

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.14 Digital Interface to XMC P16 Connector

Eight additional Pins of the TXMC637 User FPGA (Artix-7) are directly routed to the XMC P16 rear I/O connector. The I/O functions of these FPGA pins are directly dependent on the configuration of the User FPGA.

The Artix-7 VCCO voltage for these pins is set to 2.5V, so only the 2.5V I/O standards LVCMOS25, LVTTL25 and LVDS25 are possible when using this TXMC637 XMC P16 Rear I/O interface.

Signal Name	Bank	VCCO	Pin	IO Standard for example
DIG_IO_00+	13	2.5V	AB26	LVDS25
DIG_IO_00-	13		AC26	LVDS25
DIG_IO_01+	13		W21	LVDS25
DIG_IO_01-	13		Y21	LVDS25
DIG_IO_02+	13		AB24	LVDS25
DIG_IO_02-	13		AC24	LVDS25
DIG_IO_03+	13		U21	LVDS25
DIG_IO_04-	13		V21	LVDS25

Table 7-23 : XMC P16 digital I/O Interface

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

For the pin assignment at the XMC P16 connector see chapter I/O Description.

7.15 JTAG Controller to User FPGA JTAG Interface

The BCC provides two methods for accessing the User FPGA via JTAG: Bit-I/O and Vector-I/O.

7.15.1 Bit-IO

The Bit-I/O Interface is a bit-centric interface that allows stimulating the JTAG interface by setting and reading each line (TCK, TMS, TDI and TDO) of the JTAG interface independently.

All Bit-I/O register bits (JTAG_BIO_*) are directly linked/connected to the BCC JTAG Interface. If the Bit-I/O Interface is enabled (JTAG_BIO_EN = 1), these bits drive the corresponding JTAG lines.

Note that the flexibility comes along with a higher I/O effort since every signal constellation must be set and emitted on the JTAG interface via a TCK high and low sequence with separated register accesses.

7.15.2 Vector-IO

The Vector-I/O Interface focuses on a high abstract level where operations are performed semi-automatically based on vectors and transfer lengths.

Via the register interface TMS (JTAG_VIO_TMS_DATA) and TDI data (JTAG_VIO_TDI_DATA) vectors are defined that shall be shifted-out onto the JTAG interface. These vectors are coupled in a way that the data (bit-information) is shifted-out at the same time.

Note that bit #0 is the first one that appears on the JTAG interface.

Shift operations require the transfer length (JTAG_VIO_XFER_LEN) information. The value has to be set in accordance to the size of the TMS/TDI data vectors or TDO read size and defines the number of transfer cycles or rather the number of TCK cycles (rising and falling edges with 50% duty factor) occurring on the JTAG interface.
The actual TCK I/O clock rate is adjustable (JTAG_VIO_TCK_CLK_DIV).

If the Vector-I/O Interface is enabled (JTAG_VIO_EN = 1) and while the Vector-I/O Controller is idle (JTAG_VIO_CTRL_STAT = 0b01), write (JTAG_VIO_SHIFT_REQ) or read (JTAG_VIO_GET_REQ) operations can be initiated. Requests initiated while the Vector-I/O Controller is not idle are lost.

Note that there is no check regarding the number of bits shifted-out or read-in. Hence illegal data can be shifted-out or read-in if the transfer length is not set appropriately.

Bits of the output data vectors are aligned to the beginning of a TCK cycle and are hence updated after a falling edge. If more data is shifted-out than defined, zero bits (‘0’) are transferred, which is useful in JTAG IR- or DR-Shift States.
Data is read-in with every rising edge during shift-out and get-requests. This also allows obtaining the bit-information provided in response of a TMS/TDI operation.

Read-in updates appear immediately on the JTAG Vector-I/O TDO data vector (JTAG_VIO_TDO_DATA).

Note that bit #0 is the last one that has been read-in from the JTAG interface.

7.16 I2C Bridge

The I2C bridge mode allows a unidirectional I2C bus BCC reach-through from the User FPGA I2C Bus onto the BCCs isolated management BCC I2C bus. In consequence of that, the two-independent busses behave like one.

Unidirectional means in this context that only the SDA signal state is bidirectional (transferred to through the BCC) but not the SCL. SCL only traverses from User FPGA to management BCC I2C bus.

Having this feature active, allows accessing the both TMP441 temperature sensor and the SI5338 clock generator device. Both are accessible via their normal (original) I2C addresses.

Note that the Automatic Temperature Read Mode (TMP441_AUTO_TRD_EN) must be disabled before the bridge mode can be enabled.

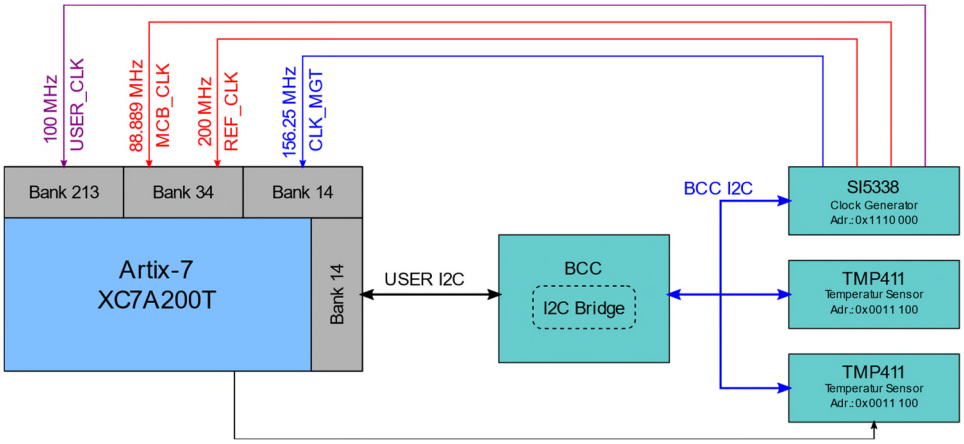


Figure 7-13 : User FPGA I2C to BCC I2C Bridge

For function description and register contents of the clock generator Si5338 and the temperature sensor TMP411 please use the data sheets of the respective device.

Do not change the SI5338 Output driver properties like “Format and Voltage”

Channel 0 must be 3.3V LVDS

Channel 1 must be 3.3V SSTL on A and B

Channel 2 must be 3.3V CMOS on A and B

Channel 4 must be 3.3V SSTL on A and B

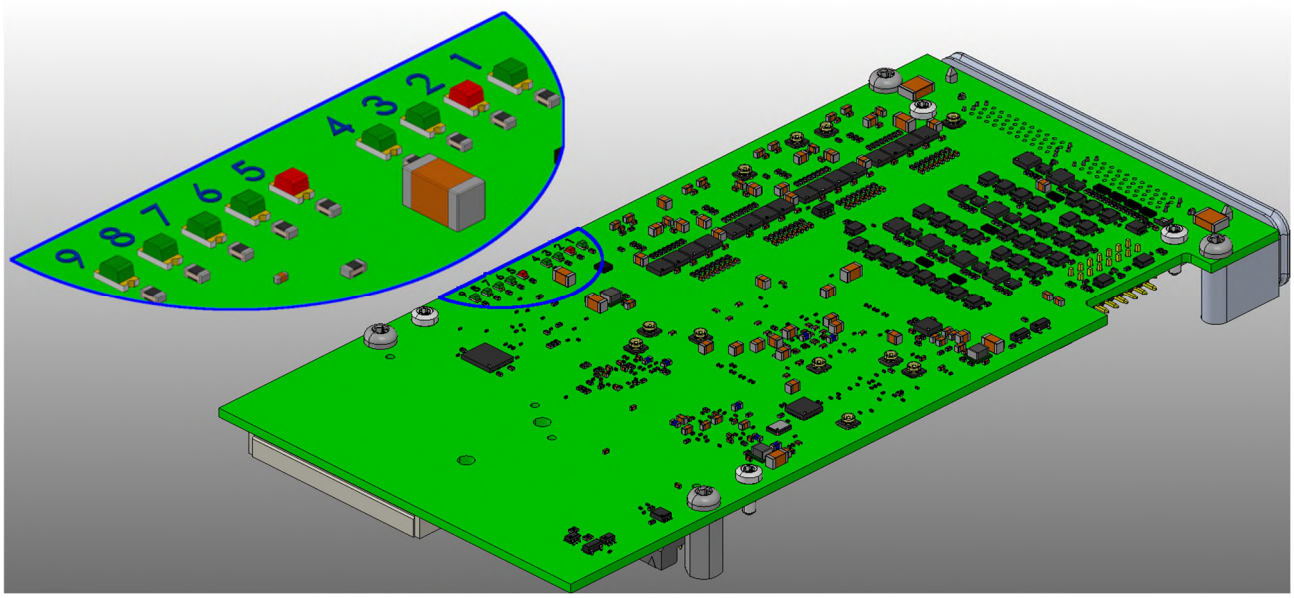
Do not change Si5338 Core VDD or I2C Bus Voltage

Core VDD = 3.3V

I2C Bus Voltage = 2.5V or 3.3V

7.17 On-Board Indicators

The TXMC637 provides a couple of board-status LEDs as shown below. These include Power-Good and FPGA configuration status indications as well as two general purpose LEDs.



No.	LED	Color	State		Description
1	Power Good	Green	on	On-Board Power Supplies are all ok	Power Good Signal for all on-board power supplies.
2		Red	on	On-Board Power Supplies are not ok	
3	USER LED 0	Green	-		Design dependent, can be controlled by the User FPGA. Refer to chapter “User-GPIO”
4	USER LED 1	Green	-		
5	FPGA INIT	Red	on	INIT state is active	User FPGA (Artix-7) INIT# - Pin LED. Indicates FPGA configuration
			off	Device is not configured	
6	BCC DONE	Green	flashing	Board initiation could not be completed	Board Configuration FPGA DONE-Pin LED Indicates successful FPGA configuration and initiation.
			on	Device is completely configured	
			off	Device is not configured	
7	FPGA DONE	Green	on	Device is completely configured	User FPGA (Artix-7) DONE-Pin LED. Indicates successful FPGA configuration
			off	General State is not IDLE	
8	GPIO LED 0	Green	on	General State is IDLE	Configuration FPGA depends.
			off	PCI Reset is active	
9	GPIO LED 1	Green	on	PCI Reset is inactive	

Table 7-24: Board-Status and User LEDs

7.17.1 User FPGA Pinning

General purpose I/O connected to the User FPGA Artix-7.

Signal	Bank	VCCO	Pin	Description
USER_LED0	14	3.3V	N26	2x green on-board LEDs
USER_LED1			M26	

Table 7-25: TXMC637 User On-Board Indicators

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

7.18 User FPGA Reset Inputs

General purpose Reset input connected to the User FPGA Artix-7.

Signal	Bank	VCCO	Pin	Description
DWNRST#	14	3.3V	R18	Reset from PCIe Switch Based directly on the downstream reset of the PCIe switch on the TXMC637. See the PCIe switch data sheet for more information.
FPGA_RST	14	3.3V	P23	Reset Input from BCC The reset signal is valid as long as the BCC is in its configuration phase. When all configuration processes on the TXMC637 are completed, this signal is deactivated.

Table 7-26: User FPGA Reset Inputs

For pin assignment, I/O standard, slew rate and driver performance please use the XDC files of the TXMC637 BRD Project.

8 Design Help

8.1 Board Reference Design

User applications for the TXMC637 may be developed by using the TXMC637 FPGA Board Reference Design.

TEWS offers this Board Reference Design as a well-documented basic example. It includes an .xdc constrain file with all necessary pin assignments and basic timing constraints. The example design covers the main functionalities of the TXMC637. It implements a PCIe endpoint with interrupt support, register mapping, DDR3 memory access and basic I/O functions. It comes as a Xilinx Vivado 2020.1 project with source code and as a ready-to-download bit stream. This example design can be used as a starting point for own projects.

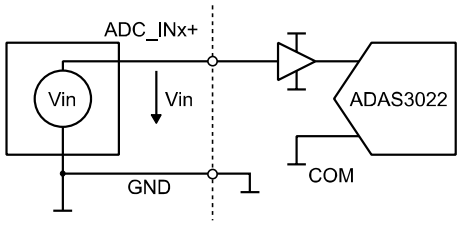
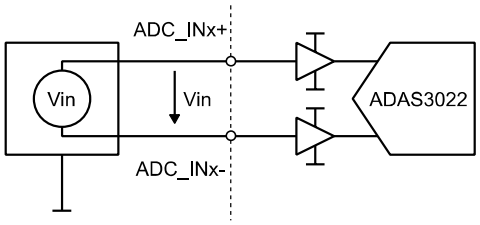
The TXMC637 FPGA Application design can be developed using the design software Vivado Design Suite. Licenses for design tools are required.

For TXMC637 FPGA Board Reference Design also see the included BRD User Manual.

9 I/O Interfaces

9.1.1 Front I/O - ADC Analog Input Level

All analog inputs lines are routed through an over voltage protection devices and an additional operational amplifier. The protection devices protect the TXMC637 ADC inputs against fault voltage above ±30V. The operational amplifier is used as an impedance converter that optimizes the input characteristics of the ADAS3022 ADC device. (Eliminates the ADC feedback on the measured source impedance.)

Single-Ended ADC Input	Diff. Source with Ground Reference
	
Maximum Vin = ±12.288V	Maximum Vin = ±24.576V

For a Single-Ended input a maximum ADC setting of PGIA = ±10.24V is useful.

If the inputs are used as differential all ADC PGIA settings are possible.

Unused ADC inputs must always be pulled to GND to avoid excessive heating of the TXMC637.

If signals without a ground reference should be connected, connect VIN+ and VIN- to GND with a resistor to prevent the signal source from floating out of the ADC's common-mode range. In most cases, a weak resistor to ground at the VIN- connection suffices.

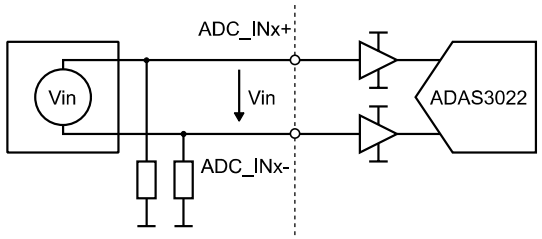
Differential without Ground Reference

Maximum Vin = ±24.576V

Table 9-1 : ADC Input without Ground Reference

An integrated programmable gain amplifier is used to adapt the analog input voltage range to the internal (static) SAR ADC voltage range. The analog input voltage range (i.e. the gain setting as shown in the overview table) is configurable for each individual A/D channel.

Note that the ADAS3022 ADC device does not provide unipolar input voltage ranges. For unipolar input signals, the appropriate (bipolar) input voltage range must be used.

The following table gives an overview of the supported analog input ranges for both analog input modes.

ADC Channel Configuration Option			Differential Mode Analog Input Range [2]		Single-Ended Mode Analog Input Range
Gain	**Differential Input Voltage Range**	**LSB**	**V_{IN+} Range (to GND) V_{IN-} Range (to GND)**	**V_{IN_DIFF} Range (V_{IN_DIFF} = V_{IN+} - V_{IN-})**	**V_{IN} Range (to GND)**
0.16	±24.576V	750uV	±12.288V	±24.576V	±12.288V [3]
0.2	±20.48V	625uV	±10.24V	±20.48V	±10.24V [3]
0.4	±10.24V	312.5uV	±10.24V	±10.24V	±10.24V
0.8	±5.12V	156.3uV	±10.24V	±5.12V	±5.12V
1.6	±2.56V	78.13uV	±10.24V	±2.56V	±2.56V
3.2	±1.28V	39.06uV	±10.24V	±1.28V	±1.28V
6.4	±0.64V	19.53uV	±10.24V	±0.64V	±0.64V

[1] ± indicates a voltage range (e.g. ±5.12V is used as a short form for -5.12V … +5.12V)

[2] The V_{IN+} and V_{IN-} voltage levels must comply with both given ranges (that is the voltage levels must meet the given range when referenced to ground and must meet the given differential voltage range as well)

[3] Only half of the available digital codes are useable for these Single-Ended configurations

Table 9-2 : Analog Input Ranges

The signal level on any Analog Input Pin on the I/O connector must not exceed 12.288V (referenced to ground)!

9.1.2 Front I/O – Analog Output Level

All analog outputs of the AD5547 DAC are routed through operational amplifiers to front I/O connector.

Outputs Drive Current	**±10mA**
Capacitive Load	1000pF

Table 9-3 : DAC Electrical Interface

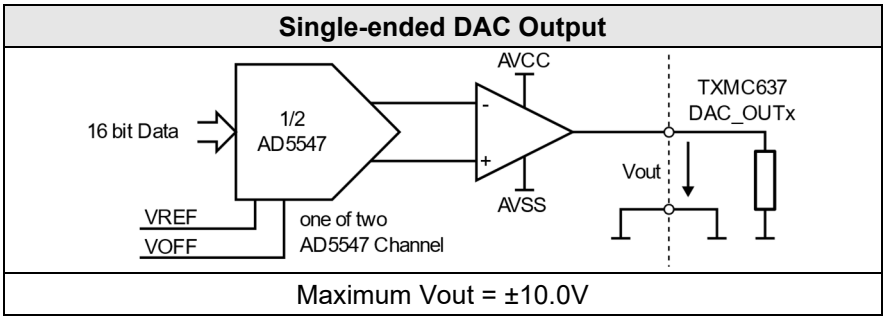


Figure 9-1 : DAC Output Interface

The output voltage range of each DAC channel can be set via VREF and VOFF.

See also the chapter about the configuration of the DAC output voltage in the BCC Register Description.

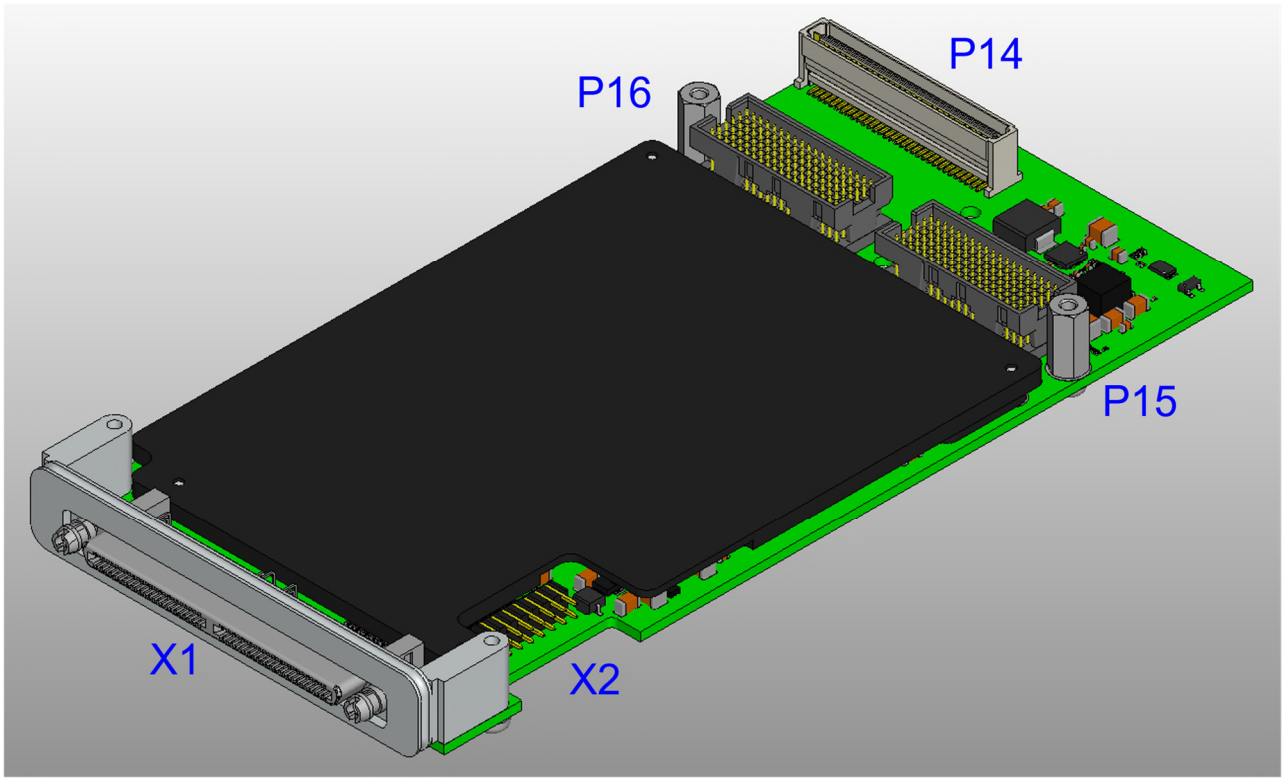
9.1.3 Rear I/O Interface

All 64 single-ended / 32 differential digital rear I/O Pins of the TXMC637 are directly routed from the User FPGA (Artix-7) to the 64 pin P14 XMC rear connector. The I/O functions of these FPGA pins are directly dependent on the configuration of the FPGA.

The Artix-7 VCCO voltage is set to 2.5V, so only the 2.5V I/O standards LVCMOS25 and LVDS25 are possible when using the TXMC637 rear I/O interface.

10 I/O Description

10.1 Overview



10.2 Front I/O Connector (X1)

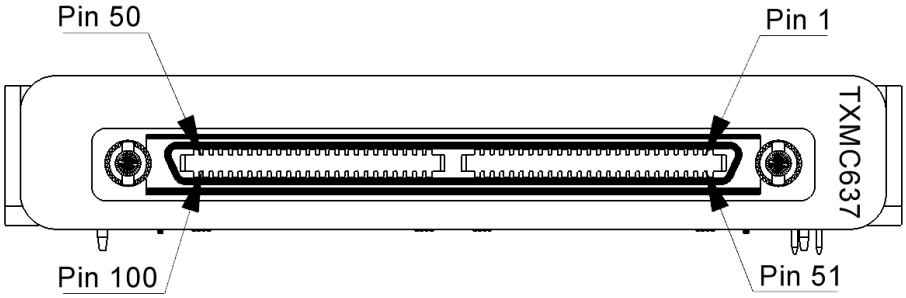


Figure 10-1 : Front Panel I/O Connector Numbering

Pin-Count	100
Connector Type	VHD100, 0.8mm Pitch Connector
Source & Order Info	HDRA-EC100LFDT-SL+ Honda

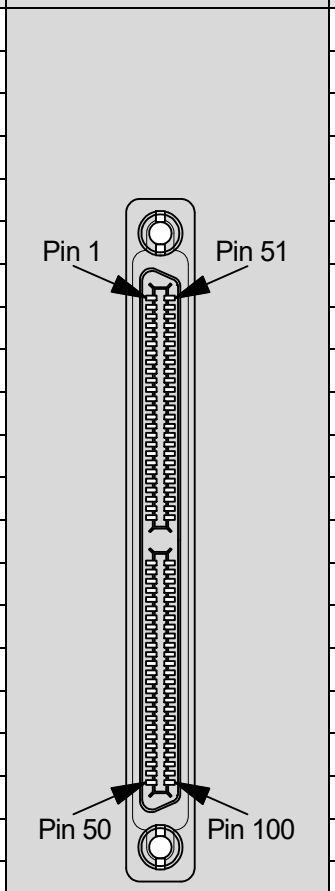
Pin	I/O	Connector View	Pin	I/O
1	IO0		51	GND
2	IO1		52	GND
3	IO2		53	GND
4	IO3		54	GND
5	IO4		55	GND
6	IO5		56	GND
7	IO6		57	GND
8	IO7		58	GND
9	IO8		59	GND
10	IO9		60	GND
11	IO10		61	GND
12	IO11		62	GND
13	IO12		63	GND
14	IO13		64	GND
15	IO14		65	GND
16	IO15		66	GND
17	IO16		67	IO17
18	IO18		68	IO19
19	IO20		69	IO21
20	IO22		70	IO23
21	IO24		71	IO25
22	IO26		72	IO27
23	IO28		73	IO29
24	IO30		74	IO31
25	GND		75	GND

Table 10-1: Pin Assignment Front Panel I/O Connector digital Part

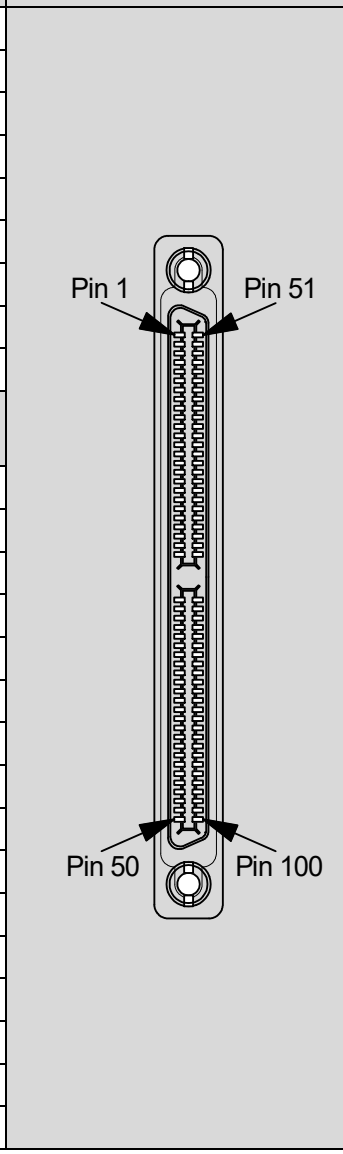
Pin	I/O single-ended		Connector View	Pin	I/O single-ended	
26	GND			76	GND	
27	DAC_OUT0			77	DAC_OUT4	
28	GND			78	GND	
29	DAC_OUT1			79	DAC_OUT5	
30	GND			80	GND	
31	DAC_OUT2			81	DAC_OUT6	
32	GND			82	GND	
33	DAC_OUT3			83	DAC_OUT7	
34	GND			84	GND	
Pin	**I/O single-ended**	**I/O differential**		**Pin**	**I/O single-ended**	**I/O differential**
35	ADC1_IN0	ADC1_IN0+		85	ADC3_IN0	ADC3_IN0+
36	ADC1_IN1	ADC1_IN0-		86	ADC3_IN1	ADC3_IN0-
37	ADC1_IN2	ADC1_IN1+		87	ADC3_IN2	ADC3_IN1+
38	ADC1_IN3	ADC1_IN1-		88	ADC3_IN3	ADC3_IN1-
39	ADC1_IN4	ADC1_IN2+		89	ADC3_IN4	ADC3_IN2+
40	ADC1_IN5	ADC1_IN2-		90	ADC3_IN5	ADC3_IN2-
41	ADC1_IN6	ADC1_IN3+		91	ADC3_IN6	ADC3_IN3+
42	ADC1_IN7	ADC1_IN3-		92	ADC3_IN7	ADC3_IN3-
43	ADC2_IN0	ADC2_IN0+		93	ADC4_IN0	ADC4_IN0+
44	ADC2_IN1	ADC2_IN0-		94	ADC4_IN1	ADC4_IN0-
45	ADC2_IN2	ADC2_IN1+		95	ADC4_IN2	ADC4_IN1+
46	ADC2_IN3	ADC2_IN1-		96	ADC4_IN3	ADC4_IN1-
47	ADC2_IN4	ADC2_IN2+		97	ADC4_IN4	ADC4_IN2+
48	ADC2_IN5	ADC2_IN2-		98	ADC4_IN5	ADC4_IN2-
49	ADC2_IN6	ADC2_IN3+		99	ADC4_IN6	ADC4_IN3+
50	ADC2_IN7	ADC2_IN3-		100	ADC4_IN7	ADC4_IN3-

Table 10-2: Pin Assignment Front Panel I/O Connector analog Part

10.3 Rear I/O Connector (P14)

Pin-Count	64
Connector Type	64pol. Mezzanine SMD Connector
Source & Order Info	Molex – 71436-2864

Pin	differential I/O	single ended I/O		Pin	differential I/O	single ended I/O
1	IO_R_0+	IO_R_0		33	IO_R_16+	IO_R_32
2	IO_R_0-	IO_R_1		34	IO_R_16-	IO_R_33
3	IO_R_1+	IO_R_2		35	IO_R_17+	IO_R_34
4	IO_R_1-	IO_R_3		36	IO_R_17-	IO_R_35
5	IO_R_2+	IO_R_4		37	IO_R_18+	IO_R_36
6	IO_R_2-	IO_R_5		38	IO_R_18-	IO_R_37
7	IO_R_3+	IO_R_6		39	IO_R_19+	IO_R_38
8	IO_R_3-	IO_R_7		40	IO_R_19-	IO_R_39
9	IO_R_4+	IO_R_8		41	IO_R_20+	IO_R_40
10	IO_R_4-	IO_R_9		42	IO_R_20-	IO_R_41
11	IO_R_5+	IO_R_10		43	IO_R_21+	IO_R_42
12	IO_R_5-	IO_R_11		44	IO_R_21-	IO_R_43
13	IO_R_6+	IO_R_12		45	IO_R_22+	IO_R_44
14	IO_R_6-	IO_R_13		46	IO_R_22-	IO_R_45
15	IO_R_7+	IO_R_14		47	IO_R_23+	IO_R_46
16	IO_R_7-	IO_R_15		48	IO_R_23-	IO_R_47
17	IO_R_8+	IO_R_16		49	IO_R_24+	IO_R_48
18	IO_R_8-	IO_R_17		50	IO_R_24-	IO_R_49
19	IO_R_9+	IO_R_18		51	IO_R_25+	IO_R_50
20	IO_R_9-	IO_R_19		52	IO_R_25-	IO_R_51
21	IO_R_10+	IO_R_20		53	IO_R_26+	IO_R_52
22	IO_R_10-	IO_R_21		54	IO_R_26-	IO_R_53
23	IO_R_11+	IO_R_22		55	IO_R_27+	IO_R_54
24	IO_R_11-	IO_R_23		56	IO_R_27-	IO_R_55
25	IO_R_12+	IO_R_24		57	IO_R_28+	IO_R_56
26	IO_R_12-	IO_R_25		58	IO_R_28-	IO_R_57
27	IO_R_13+	IO_R_26		59	IO_R_29+	IO_R_58
28	IO_R_13-	IO_R_27		60	IO_R_29-	IO_R_59
29	IO_R_14+	IO_R_28		61	IO_R_30+	IO_R_60
30	IO_R_14-	IO_R_29		62	IO_R_30-	IO_R_61
31	IO_R_15+	IO_R_30		63	IO_R_31+	IO_R_62
32	IO_R_15-	IO_R_31		64	IO_R_31-	IO_R_63

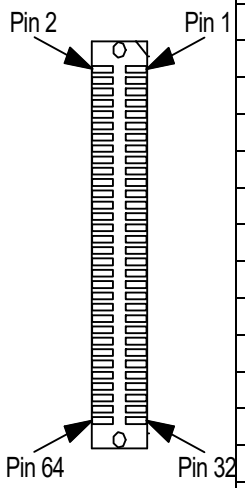


Figure 10-2 : Pin Assignment P14 Rear I/O Connector

10.4 Rear I/O Connector (P16)

Pin-Count	114
Connector Type	XMC Connector 114-pol Male
Source & Order Info	Samtec - ASP-105885-01

	A	B	C	D	E	F
1	Tx 0+	Tx 0-	-	Tx 1+	Tx 1-	-
2	GND	GND	-	GND	GND	-
3	Tx 2+	Tx 2-	-	Tx 3+	Tx 3-	-
4	GND	GND	-	GND	GND	-
5	-	-	-	-	-	-
6	GND	GND	-	GND	GND	-
7	-	-	-	-	-	-
8	GND	GND	-	GND	GND	-
9	Reserved	Reserved	-	Reserved	Reserved	-
10	GND	GND	-	GND	GND	-
11	Rx 0+	Rx 0-	-	Rx 1+	Rx 1-	-
12	GND	GND	-	GND	GND	-
13	Rx 2+	Rx 2-	-	Rx 3+	Rx 3-	-
14	GND	GND	-	GND	GND	-
15	-	-	-	-	-	-
16	GND	GND	DIG_IO_02-	GND	GND	DIG_IO_03-
17	-	-	DIG_IO_02+	-	-	DIG_IO_03+
18	GND	GND	DIG_IO_00-	GND	GND	DIG_IO_01-
19	-	-	DIG_IO_00+	-	-	DIG_IO_01+

Figure 10-3 : Pin Assignment P16 Rear I/O Connector

10.5 JTAG Header (X2)

This header is connected via multiplexers to the JTAG I/Os of the on-board User FPGA JTAG interface. The BCC can disconnect the header to become the JTAG master. The pinout of this header is designed in accordance with the Xilinx Platform Cable USB II. This allows the direct usage of Xilinx software-tools like Vivado Logic Analyzer or the Vivado Hardware Manager.

Pin-Count	14
Connector Type	2.00 mm Pitch Milli-Grid™ Header
Source & Order Info	Molex 877601416 or compatible

Pin	Signal	Description
1	NC	Not Connected
2	V_{REF}	JTAG Reference Voltage (3.3V)
3	GND	Ground
4	TMS	Test Mode Select Input
5	GND	Ground
6	TCK	Test Clock
7	GND	Ground
8	TDO	Test Data Output (TAP Controller: TDI)
9	GND	Ground
10	TDI	Test Data Input (TAP Controller: TDO)
11	GND	not connected on the TXMC637
12	NC	not connected on the TXMC637
13	PGND	Used on TXMC637 for XILINX Header present detection
14	NC	HALT_INIT_WP signal. Optional. Not connected on the TXMC637

Table 10-3: Pin Assignment JTAG Header