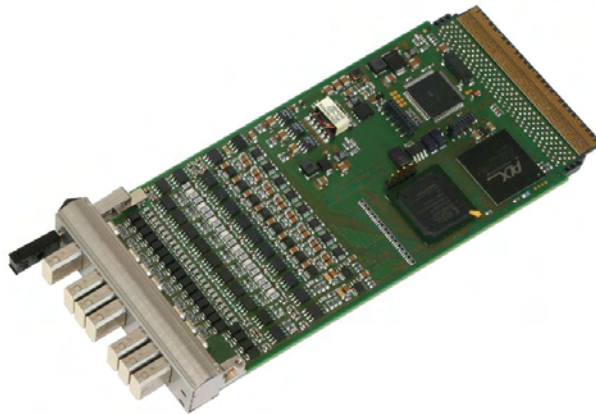




# AMC-ADIO24

## AMC Analog/Digital I/O Module



## Hardware Manual

to Product U.1001.01



## NOTE

The information in this document has been carefully checked and is believed to be entirely reliable. esd makes no warranty of any kind with regard to the material in this document, and assumes no responsibility for any errors that may appear in this document. esd reserves the right to make changes without notice to this, or any of its products, to improve reliability, performance or design.

esd assumes no responsibility for the use of any circuitry other than circuitry which is part of a product of esd gmbh.

esd does not convey to the purchaser of the product described herein any license under the patent rights of esd gmbh nor the rights of others.

esd electronic system design gmbh  
Vahrenwalder Str. 207  
30165 Hannover  
Germany

Phone: +49-511-372 98-0  
Fax: +49-511-372 98-68  
E-Mail: [info@esd.eu](mailto:info@esd.eu)  
Internet: [www.esd.eu](http://www.esd.eu)

USA / Canada:  
esd electronics Inc.  
525 Bernardston Road  
Suite 1  
Greenfield, MA 01301  
USA

Phone: +1-800-732-8006  
Fax: +1-800-732-8093  
E-mail: [us-sales@esd-electronics.com](mailto:us-sales@esd-electronics.com)  
Internet: [www.esd-electronics.us](http://www.esd-electronics.us)

### Trademark Notices

The PICMG® name and logo are registered trademarks of the PCI Industrial Computer Manufacturers Group. All other trademarks, product names, company names or company logos used in this manual are reserved by their respective owners.

<b>Document file:</b>	I:\Texte\Doku\MANUALS\AMC\AMC-ADIO24\Englisch\AMC-ADIO24_Hardware_en_10.odt
<b>Date of print:</b>	2011-01-10
<b>Document type number:</b>	DOC0800

<b>Firmware version:</b>	from FPGA release 2.0 on
--------------------------	--------------------------

## Document History

The changes in the document listed below affect changes in the hardware as well as changes in the description of the facts, only.

Revision	Chapter	Changes versus previous version	Date
1.0	-	First released version of English manual.	2011-01-10

Technical details are subject to change without further notice.



## Safety Instructions

- When working with AMC-ADIO24 follow the instructions below and read the manual carefully to protect yourself and the AMC-ADIO24 from damage.
- Protect the AMC-ADIO24 from dust, moisture and steam.
- Protect the AMC-ADIO24 from shocks and vibrations.
- The AMC-ADIO24 may become warm during normal use. Always allow adequate ventilation around the AMC-ADIO24 and use care when handling.
- Do not operate the AMC-ADIO24 adjacent to heat sources and do not expose it to unnecessary thermal radiation. Ensure an ambient temperature as specified in the technical data.
- Do not use damaged or defective cables to connect the AMC-ADIO24.

### Qualified Personal

This documentation is directed exclusively towards qualified personal in control and automation engineering.

The installation and commissioning of the product may only be carried out by qualified personal, which is authorized to put devices, systems and electric circuits into operation according to the applicable national standards of safety engineering.

### Intended Use

The intended use of the AMC-ADIO24 is the operation as AMC analog/digital I/O module in a  $\mu$ TCA system.

The guarantee given by esd does not cover damages which result from improper use, usage not in accordance with regulations or disregard of safety instructions and warnings.

- The AMC-ADIO24 is intended for installation in a MicroTCA-system only.
- The operation of the AMC-ADIO24 in hazardous areas, or areas exposed to potentially explosive materials is not permitted.
- The operation of the AMC-ADIO24 for medical purposes is prohibited.

### Service Note

The AMC-ADIO24 does not contain any parts that require maintenance by the user. The AMC-ADIO24 does not require any manual configuration of the hardware.

### Note on Environmental Protection

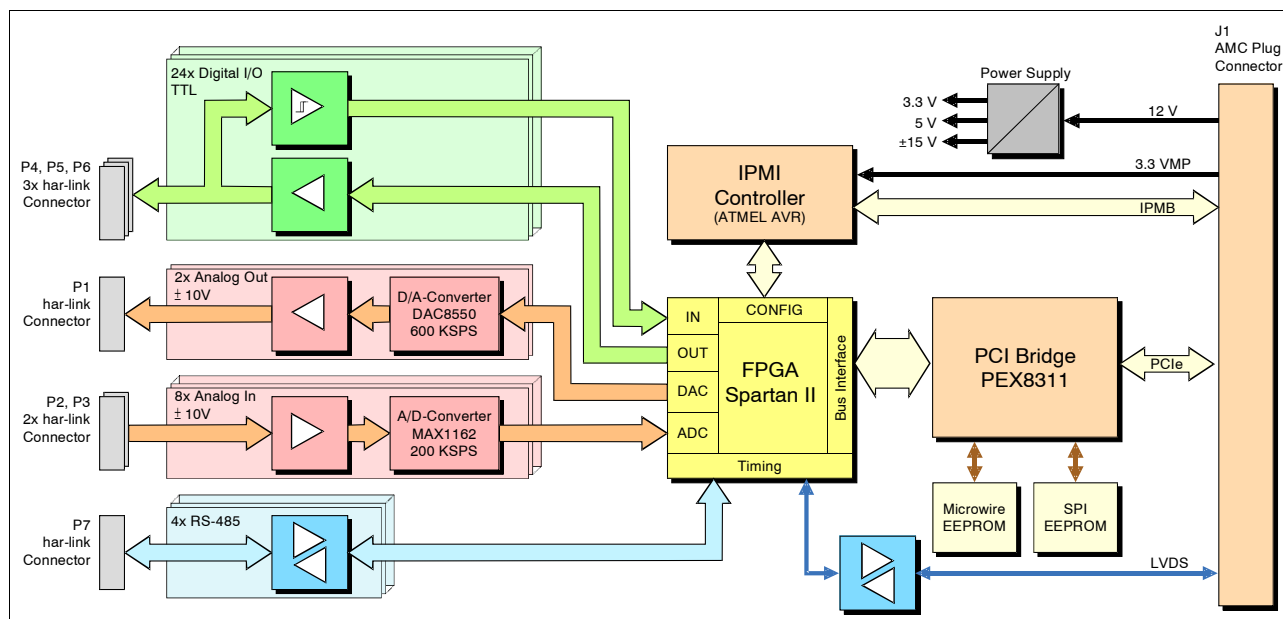
Devices which have become defective in the long run have to be disposed in an appropriate way or have to be returned to the manufacturer for proper disposal. Please, make a contribution to environmental protection.

# Table of contents

1. Overview.....	7
2. PCB View with Connectors.....	8
3. Hardware Installation .....	9
3.1 Front-Panel View.....	10
3.1.1 Front Panel LEDs and Connectors.....	10
3.1.2 LED Indication.....	10
4. PCIe Device Access.....	11
4.1 PCI Address Space.....	11
4.2 Software Development Kit for the PCI Bridge PEX8311.....	12
4.3 Data Access.....	13
4.3.1 Direct I/O Access.....	13
4.3.2 Setting Output Data.....	14
4.3.3 Periodic Pattern Generation.....	14
4.4 Streaming I/O and Trigger Selection.....	15
4.4.1 Streaming I/O Block Diagram.....	15
4.4.2 Trigger Selection.....	16
4.4.2.1 Overview.....	16
4.4.2.2 Timing Routing Pool.....	16
4.5 Store/Restore Default Parameters (Calibration, FRU, FPGA-Image).....	17
4.6 FPGA Memory Map.....	18
4.7 Register Description.....	21
4.7.1 Version.....	21
4.7.2 Status.....	22
4.7.3 DDI (Direct Digital In, Inclusive Inversion).....	24
4.7.4 DDO (Direct Digital Out, No Inversion).....	25
4.7.5 DIOMode (Trigger Mode Din/Dout).....	26
4.7.6 DOut Invert (Digital Output Inverted).....	27
4.7.7 HSE/LSE (HighSide Enable / LowSide Enable).....	28
4.7.8 FGENAB (DDFS Frequency Generator).....	29
4.7.9 FGENCD (Clock Divider).....	30
4.7.10 SyncOut (External Trigger Output/Input).....	31
4.7.11 DORout (Trigger Output/Input, DigOut-Route).....	32
4.7.12 DInInvert (Digital Input Inverted).....	33
4.7.13 DInFilter (Filter Digital Input).....	34
4.7.14 ADCMode (Trigger Mode ADC7...0).....	35
4.7.15 DIVMode (DMA-Mode, Timer Select, Interrupt Enable).....	36
4.7.16 Auxreg (Trigger Mode DAC A/B).....	39
4.7.17 QDac (Direct Read Back DAC A/B).....	40
4.7.18 DACCorr (Gain/Offset Correction DAC B/A).....	41
4.7.19 EVENT.....	42
4.7.20 DMAMode (DMA enable/disable).....	44
4.7.21 Compare.....	46
4.7.22 ADC1Corr ... ADC8Corr (ADC Gain and Offset).....	47
4.7.23 DOWriteX (Digital Out Mask).....	48
4.7.24 DACWriteBA (Write DAC).....	49
4.7.25 ChanWrite.....	50
4.7.26 Current Register Values.....	51
4.7.26.1 ADCxAct (Current Analog Input Values).....	51
4.7.26.2 DInAAct, DInBAAct (Current Digital Input Values).....	52
4.7.26.3 DOutAAAct, DOutBAAct (Current Digital Output Values).....	52
4.7.26.4 DACAAAct, DACBAAct (Current Analog Output Values).....	53
4.7.26.5 TSAct (Current Timestamp Counter Value).....	54

4.7.27	Last DMA Values.....	55
4.7.27.1	ADC1Last... ADC8Last (Last Analog Input DMA Values).....	55
4.7.27.2	DInALast, DInBLast (Last Digital Input DMA Values).....	56
4.7.27.3	DOutALast, DOutBLast (Last Digital Output DMA Values).....	56
4.7.27.4	DACALast, DACALast (Last Analog Output DMA Values).....	57
4.7.27.5	TSLast (Last TS-Counter DMA Values).....	57
4.7.28	Difference Values for Trigger Conditions .....	58
4.7.28.1	ADC1Delta ... ADC8Delta.....	58
4.7.28.2	DInA/BMask.....	59
4.7.28.3	DOutA/BMask.....	60
4.7.28.4	DACA/BDelta.....	61
4.7.28.5	TSDelta.....	62
4.7.29	Command Register.....	63
4.8	I/O Data Processing.....	65
4.9	DMA Access.....	66
4.9.1	DMA Input Data.....	66
4.9.1.1	Read Data FIFO.....	66
4.9.1.2	Setting DMA Operation.....	68
4.9.2	DMA Output Data.....	69
5.	Technical Data.....	70
5.1	General Technical Data.....	70
5.2	MicroTCATM /AMC Standards.....	71
5.3	Analog Outputs.....	71
5.4	Analog Inputs.....	72
5.5	Digital Inputs/Outputs.....	73
5.6	Trigger Ports.....	73
6.	Interfaces and Connector Assignments.....	74
6.1	Analog Out.....	74
6.2	Analog In.....	75
6.3	Digital I/O.....	76
6.4	Trigger Ports.....	77
7.	AMC-ADIO-Adapter (Order no. U1001.10).....	78
7.1	AMC-ADIO-Adapter View.....	79
7.2	Technical Data AMC-ADIO-Adapter.....	80
7.3	Connector Assignment.....	81
8.	Order Information.....	83
9.	Abbreviations.....	84

# 1. Overview



**Figure 1:** Block circuit diagram

The AMC-ADIO24 is an AMC analog/digital I/O module.

It is equipped with a Spartan FPGA, which manages the I/O data exchange in cooperation with the PCIe bridge. FIFOs for input and output direction and DMA to the PCIe host CPU's memory minimizes undesired latency during PCIe read cycles at higher data rates. Read cycles of the PCIe CPU are reduced to setup and diagnosis tasks.

The eight overvoltage protected analog inputs are connected to eight 16 bit A/D converters with a sampling rate of up to 200 kHz each.

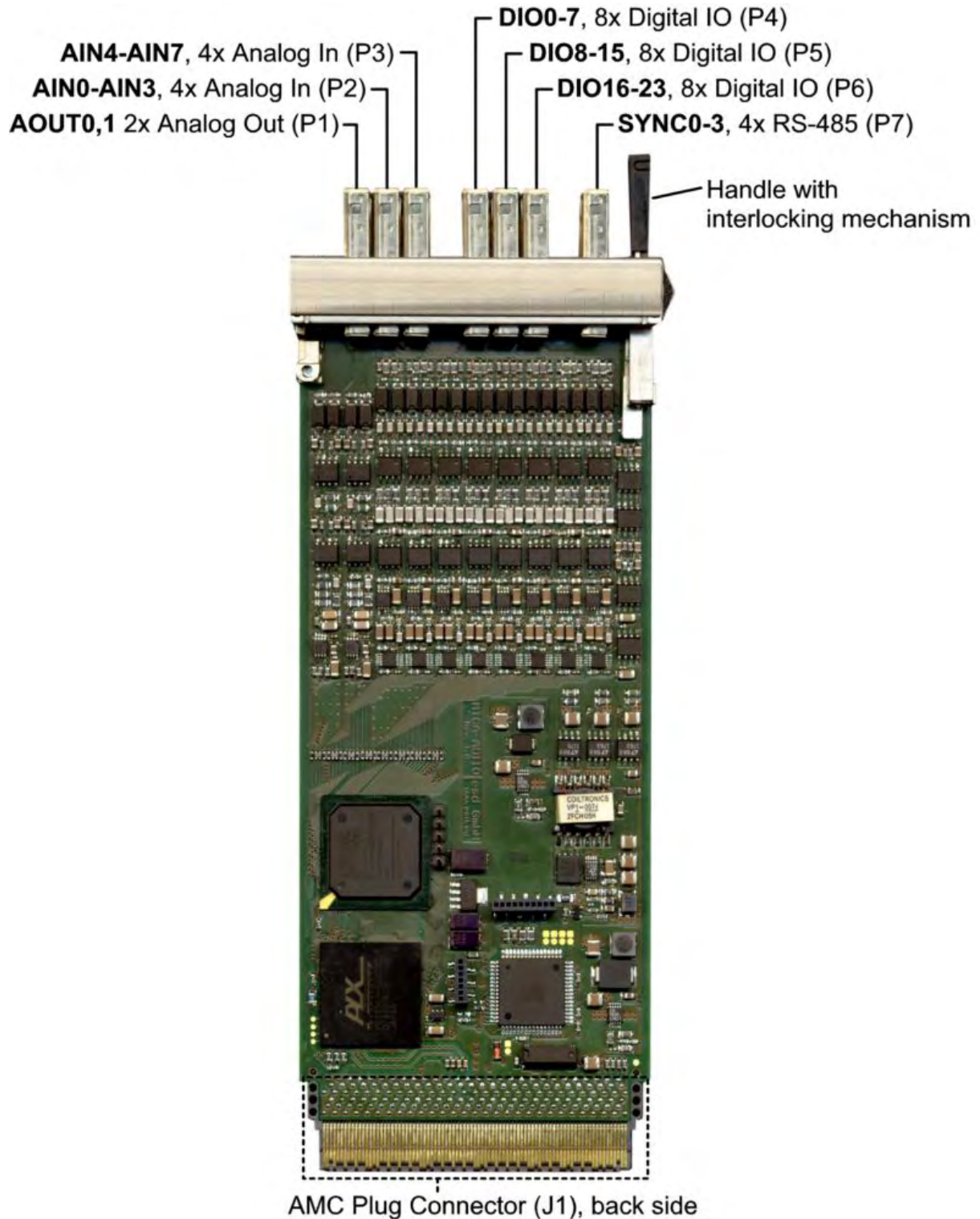
Both 16 bit analog outputs have a differential and a single ended output circuit, accessible at separate pins at a Harting® har-link® connector. The outputs are transient and short circuit protected. The maximum update rate of the analog outputs is 600 KSPS (kilo samples per second).

Each of the 24 TTL-level I/Os can be separately configured as input or output. The outputs can be configured as high side driver, low side driver or both (sink/source). The I/O port state can be read back in any configuration via a comparator with hysteresis.

For the trigger I/Os for synchronisation the firmware offers a so called 'Timing-Routing-Pool' with various trigger conditions (RS-485 trigger input, timer, software, free run) that can be individually evaluated for each I/O or I/O group.

For setup and I/O data exchange a comprehensive register description is accessible via PCIe (see chapter 4).

## 2. PCB View with Connectors



**Figure 2:** PCB top view

See also page 74 et seqq. for signal assignments of the connectors.

### 3. Hardware Installation



Read the safety instructions at the beginning of this document carefully, before you start with the hardware installation!



**Attention !**

Electrostatic discharges may cause damage to electronic components. To avoid this, please discharge the static electricity from your body by touching the metal case of the  $\mu$ TCA system *before* you touch the AMC-ADIO24.

**Procedure:**

1. The AMC-ADIO24 is hot-pluggable.  
Insert the AMC-ADIO24 into a free slot in your  $\mu$ TCA system.
2. Fix the AMC-ADIO24 by pushing the handle with interlocking mechanism (see figure 2).
3. Connect the signal lines to the har-link-connectors (P1 - P7) in the front panel (see figure 2) e.g. via the har-link® cable (see page 83). The external signal-lines are 'hot-pluggable' and can be connected or disconnected at discretion.
4. End of hardware installation.
5. Set the interface properties in your operating system. For further information refer to the documentation of the operating system.

### 3.1 Front-Panel View

#### 3.1.1 Front Panel LEDs and Connectors

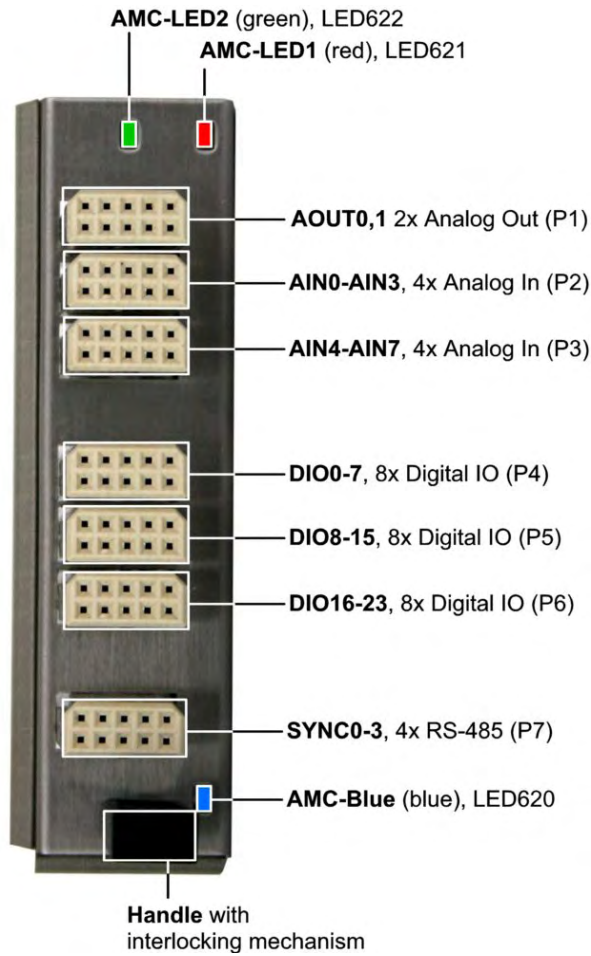


Figure 3: Connectors and LEDs

#### 3.1.2 LED Indication

Name	Colour	Description	LED name in schematics diagram	
AMC-LED2	green	Controlled by IPMI. Local function: Lit if FPGA is booted correctly.	LED622	
AMC-LED1	red	Controlled by IPMI.	LED621	
AMC-Blue	blue	Controlled by IPMI.	LED620	
		Off		in operation
		Blinking		preparing for hot-plug (in transition)
		Lit		powered off, hot-plug allowed

## 4. PCIe Device Access

A 64K address space is required for the AMC-ADIO24. The address space must be accessible by the host CPU of the PCIe bus.

### 4.1 PCI Address Space

The PCI base address of the AMC-ADIO24 depends on the PCI host that assigns the PCI bus memory addresses to the connected PCI bus boards.

The following listing shows an example of the memory assignment at a x86 Linux system, after calling the command 'lspci -vv'.

Note that the PCIe bridge of the AMC-ADIO24 (PEX8311) internally carries two PCI-bridges. therefore the PCI host detects and displays two memory spaces:

```

...
...
06:00.0 PCI bridge: PLX Technology, Inc. PEX 8111 PCI Express-to-PCI Bridge (rev 21) (prog-if 00 [Normal decode])
  Control: I/O+ Mem+ BusMaster+ SpecCycle- MemWINV+ VGASnoop- ParErr- Stepping- SERR- FastB2B-
  Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort- <MAbort- >SERR- <PERR-
  Latency: 0, Cache Line Size: 32 bytes
  Region 0: Memory at fea00000 (64-bit, prefetchable) [size=64K]
  Bus: primary=06, secondary=07, subordinate=07, sec-latency=36
  I/O behind bridge: 00002000-00002fff
  Memory behind bridge: fe600000-fe6fffff
  Secondary status: 66MHz- FastB2B- ParErr- DEVSEL=medium >TAbort- <TAbort- <MAbort+ <SERR- <PERR-
  BridgeCtl: Parity- SERR- NoISA+ VGA- MAbort- >Reset- FastB2B-
  Capabilities: [50] Message Signalled Interrupts: Mask- 64bit+ Queue=0/0 Enable-
    Address: 0000000000000000 Data: 0000
  Capabilities: [60] Express PCI/PCI-X Bridge IRQ 0
    Device: Supported: MaxPayload 128 bytes, PhantFunc 0, ExtTag-
    Device: Latency L0s <64ns, L1 <1us
    Device: AtnBtn- AtnInd- PwrInd-
    Device: Errors: Correctable- Non-Fatal- Fatal- Unsupported-
    Device: RlxdOrd- ExtTag- PhantFunc- AuxPwr- NoSnoop-
    Device: MaxPayload 128 bytes, MaxReadReq 4096 bytes
    Link: Supported Speed 2.5Gb/s, Width x1, ASPM L0s L1, Port 0
    Link: Latency L0s <1us, L1 <16us
    Link: ASPM Disabled CommClk- ExtSynch-
    Link: Speed 2.5Gb/s, Width x1
  Capabilities: [100] Power Budgeting

07:04.0 Signal processing controller: PLX Technology, Inc. Francois (rev ba)
  Subsystem: ESD Electronic System Design GmbH Unknown device 0600
  Control: I/O+ Mem+ BusMaster+ SpecCycle- MemWINV+ VGASnoop- ParErr- Stepping- SERR- FastB2B-
  Status: Cap+ 66MHz+ UDF- FastB2B+ ParErr- DEVSEL=medium >TAbort- <TAbort- <MAbort- >SERR- <PERR-
  Latency: 32, Cache Line Size: 32 bytes
  Interrupt: pin A routed to IRQ 7
  Region 0: Memory at fe610000 (32-bit, non-prefetchable) [size=512]
  Region 1: I/O ports at 2000 [size=256]
  Region 2: Memory at fe600000 (32-bit, non-prefetchable) [size=64K]
  Capabilities: [40] Power Management version 2
    Flags: PMEClk- DSI- D1- D2- AuxCurrent=0mA PME(D0-,D1-,D2-,D3hot-,D3cold-)
    Status: D0 PME-Enable- DSel=0 DScale=0 PME-
  Capabilities: [48] #06 [0000]
  Capabilities: [4c] Vital Product Data
...
...

```

## PCIe Device Access

---

For the user the memory base address of the **FPGA** at the AMC-ADIO24 is most important. It is always placed in the PCI bridge called

**'Subsystem: ESD Electronic System Design GmbH...'**

after the I/O-ports and is displayed as

**'Region 2: Memory at fe600000 (32-bit, non-prefetchable) [size=64K]'**,

i.e. the base address is 0xFE600000 in this example.

The memory space size reserved for the FPGA registers always is defined as 64k.

## 4.2 Software Development Kit for the PCI Bridge PEX8311

For an easier start we recommend to use the software development kit (SDK) for the PEX8311, called 'PlxSDK'. This kit contains an universal driver for Windows and Linux (with source code) that allows the access to the user memory space of the bridge.

The PlxSDK can be downloaded from the PLX homepage:

<http://www.plxtech.com/products/sdk/pde>

### 4.3 Data Access

The AMC-ADIO24 offers simple I/O access via PCIe. But attention should be paid to the duration of a read cycle via PCIe, which can last up to 2...3  $\mu$ s.

Therefore DMA operation to access the input process data, supported by the PCI bridge PEX8311, is highly recommended to achieve higher data rates. The DMA is assisted by a local FIFO in the FPGA, designed as a circular buffer with a depth of 256 entries.

For more information on the DMA mode read chapter 4.9 from page 66 on.

#### 4.3.1 Direct I/O Access

A direct access to the I/Os can be achieved by FPGA registers. The following table displays examples of I/O registers for direct access:

I/O Port	Access	Register Name	See Page	Notes
Digital Inputs	read	DInAAct, DInBAct	52	read latched values of digital inputs
	read	DDI	24	read transparent values of digital inputs
Digital Outputs	write	DOWrite32	48	write all digital outputs at once
	write	DOWriteA, DOWriteB	48	write digital outputs
	read	DOutAAct, DOutBAct	52	read latched values of digital outputs
	read	DDO	25	read transparent values of digital outputs
Analog Inputs	read	ADC0Act ... ADC7Act	51	read latched values of analog inputs
Analog Outputs	write	DACWriteBA	49	write analog outputs
	read	DACAAct, DACBAct	53	read latched values of analog outputs

The current timing of setting and reading the I/Os depends on the SYNC timing which strongly depends on the settings of of the so called 'Timing Routing Pool' (see page 15 et seq.).

### 4.3.2 Setting Output Data

The output data is combined in six groups with 16 bits each.

Group	Outputs
1	Dout07 ... Dout00 (Enable + Data)
2	Dout15 ... Dout08 (Enable + Data)
3	Dout23 ... Dout16 (Enable + Data)
4	Dout31 ... Dout24 (internal use only!)
5	AOUT1
6	AOUT2

Each of these groups has FIFOs with a depth of 256 entries. If a new entry is generated by the bus side (host CPU), the physical output is set with the next valid trigger condition.

A selective bit access to the digital outputs without reading the current output state can be executed by validation of a write access within an output group (see register 'DOWriteA/B').

### 4.3.3 Periodic Pattern Generation

The output-FIFO memory can be setup to generate periodical signals at the outputs (DOut-Pattern, SIN-Wave, ...).

Is the according bit in parameter *Output-Table Mode* in register 'DIVMode' (reg. no. 0xF, page 36) set to '1', the last 256 entries of the FIFO are periodically output controlled by the according trigger event.

## 4.4 Streaming I/O and Trigger Selection

### 4.4.1 Streaming I/O Block Diagram

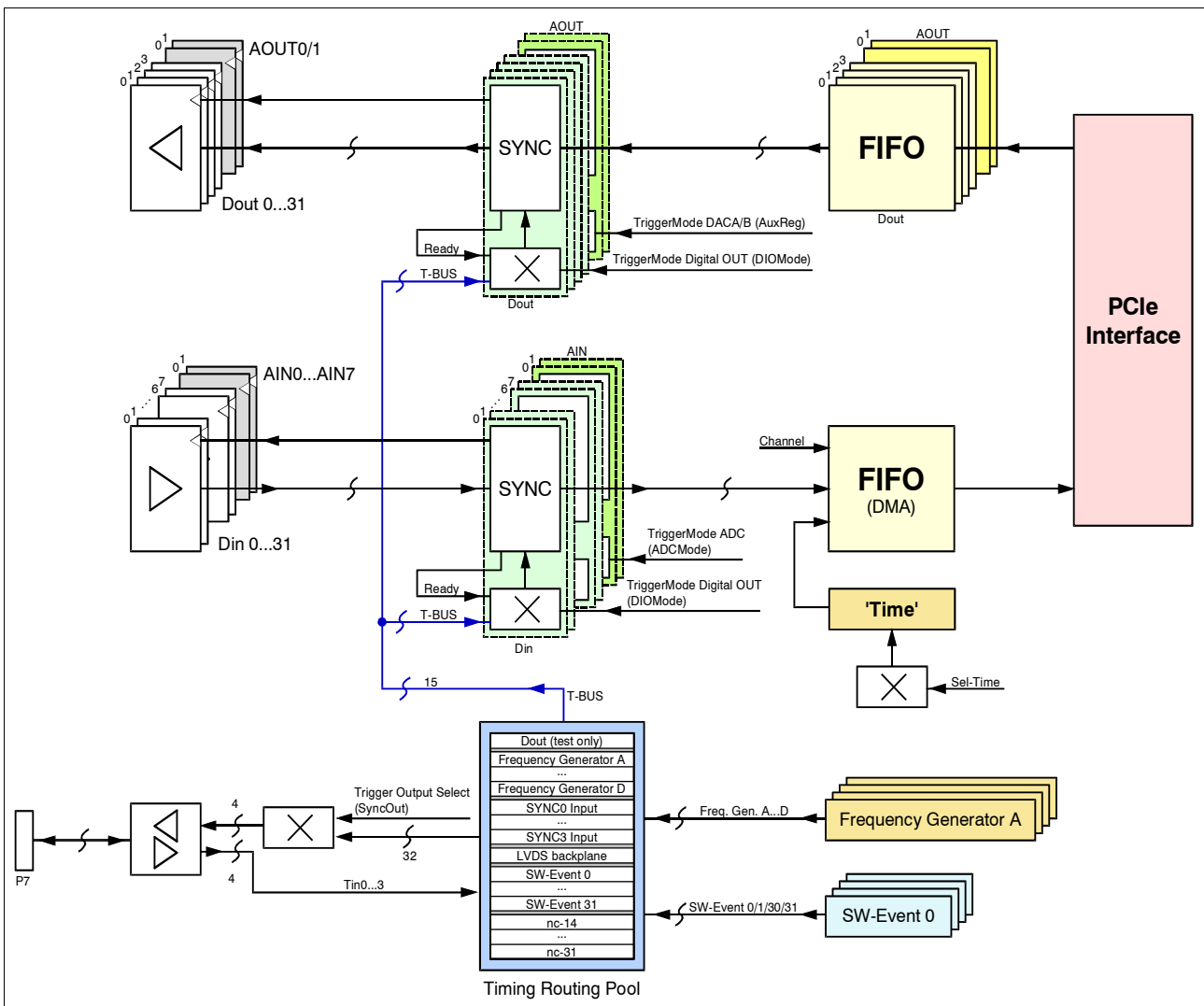


Figure 4: Streaming I/O block diagram

## 4.4.2 Trigger Selection

### 4.4.2.1 Overview

All input and output groups are featured with an independent trigger selector. For the outputs the trigger selector controls the reading of the FIFOs and the setting of the physical outputs. For the inputs the trigger selector controls the start of sampling.

For each group 16 different trigger sources are available:

- 0: channel-specific: 'done' generated next trigger ('free run')
- 1...15: trigger source is taken from Timing Routing Pool (1...15) 'rising edge'

### 4.4.2.2 Timing Routing Pool

The AMC-ADIO24 offers an internal routing pool for the trigger input and trigger output signals (T-Bus). This pool offers 32 independent signals. The trigger signals of the routing pool can be used to update the I/O signals and can be output at the SYNC0...SYNC3 interfaces as well.

Line	Trigger Source
0	DigOut(DORout) One of Dout31..Dout00 - For Test only
1	Frequency Generator A
2	Frequency Generator B
3	Frequency Generator C
4	Frequency Generator D
5	SYNC0 Input
6	SYNC1 Input
7	SYNC2 Input
8	SYNC3 Input
9	LVDSIn (Backplane, not yet implemented!)
10	SW-Event 0
11	SW-Event 1
12	SW-Event 30
13	SW-Event 31
14...31	reserved

**Table:** Assignment of the TRP (Timing Routing Pool)

To generate local trigger events the following options are available:

1. By software via generation of a software event  
(SW-Event 0, SW-Event 1, SW-Event 30, SW-Event 31; see page 42)
2. By timing generator (register FGENAB, FGENCD; see page 29)
3. By external trigger signal (SYNC0...SYNC3)

Each of the 32 local timing signals can be output at the according 'SYNCX' channel. Where applicable the polarity can be inverted. All output signals are available as 'SYNCXin' in the routing pool as well.

## 4.5 Store/Restore Default Parameters (Calibration, FRU, FPGA-Image)

Beside the classic functions of an IPMI controller the controller at the AMC-ADIO24 offers some application-specific features:

1. Factory calibration data  
The analog inputs and outputs are calibrated at the factory during the end test of the module. During this calibration gain and offset values are determined and stored in the IPMI controller's EEPROM. At start-up the calibration data is loaded into the FPGA.  
The gain and offset values can be read and can be overwritten in the according FPGA registers (ADC0Corr...ADC7Corr, DACCor).  
Via a command register the user can store the current calibration data in the controller's EEPROM.
2. Reset to factory settings  
The calibration data, the FPGA-image and the FRU data can be reset to factory default via the command register.

A detailed register description is printed from page 63 on.

### 4.6 FPGA Memory Map

Addr. ... address  
 Attr. ... attribute  
 Acc. ... access  
 RO ... read only (writes are ignored)  
 RW ... read and write  
 R(W) ... writable, but will be overwritten by local functions  
 WO ... write only  
 b / w / l ... permitted width for write access (b: byte, w: word, l: long)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
0	0000	RO	n/a	Version	abcdEEEE	11000000	a = type, b = FPGA-version, cd = PCB-version, eeee = feature flags
1	0004	RO	n/a	Status	vvvvvvvv	xxxxxxxx	miscellaneous status bits
2	0008	RO	n/a	DDI	vvvvvvvv	xxxxxxxx	DirectDigitalIn 32 bit
3	000C	RO	n/a	DDO	vvvvvvvv	00000000	DirectDigitalOut 32 bit
4	0010	RW	bwl	DIOMode	hgfedcba	00000000	TriggerMode Digital In/Out
5	0014	RW	bwl	DOutInvert	vvvvvvvv	00000000	DigOut InversionMask 32 bit
6	0018	RW	bwl	HSE	vvvvvvvv	00000000	DigOut High_Side_Enable 32 bit
7	001C	RW	bwl	LSE	vvvvvvvv	00000000	DigOut Low_Side_Enable 32 bit
8	0020	RW	wl	FGENAB	bbbbaaaa	00000000	FreqGenA/B Value 2x16 bit
9	0024	RW	wl	FGENCD	ddddcccc	00000000	FreqGenC/D Value 2x16 bit
A	0028	RW	bwl	SyncOut	ddccbbaa	00000000	Trigger-Output-Select 4x8 bit (RS-485)
B	002C	RW	bwl	DORout	rrrraaee	00000000	aa = Selector DigOut to TRP0, ee = TriggerOut-Select (LVDS Backplane) rrrr = reserved
C	0030	RW	bwl	DInInvert	vvvvvvvv	00000000	DigIn InversionMask 32 bit
D	0034	RW	bwl	DInFilter	vvvvvvvv	00000000	DigIn FilterEnable 32 bit
E	0038	RW	bwl	ADCMode	hgfedcba	00000000	TriggerMode for ADC7...ADC0
F	003C	RW	bwl	DIVMode	missoxtt	00000000	m = DMA-Mode i = IrqEnable3...0 ss = TS-Counter-Select o = Master of DMA-Out x = unused tt = Output-Table Mode
10	0040	RW	bwl	AuxReg	rrrrdcba	00000000	ba = TriggerMode DACA/B, dc = Enable-Bits Out-FIFO-Irq 5...0 rrrr = reserved
11	0044	RO	n/a	QDac	bbbbaaaa	00000000	DirectDacValue DAC_B/A 2x16 bit
12	0048	RW	bwl	DACCorr	ddccbbaa	00000000	Gain/Offset-Correction DAC_B/A
13	004C	RW	bwl	EVENT	vvvvvvvv	00000000	PCI-Write: Set SW-Event 31...0 SPI-Write: Ack SPI-Irq All_Read: SPI-Irq 15...0, DMA-Pointer
14	0050	RW	bwl	DMAMode	ddddeeee	00000000	dddd = DMA-Write Disable eeee = DMA-TimeEnable 16-Bit

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
15	0054	R	-	Compare Value	ttttcccc	00000000	tttt = Current Timestamp Counter cccc = Current Compare Value
		W	wl	Next	rrrrrnnnnn	00000000	nnnn = unsigned Delta Compare Value rrrr = reserved
..	..	..	..	..	..	..	..
30	00C0	RW	wl	ADC0Corr	uuuuggoo	ffff0000	ADC0: Gain/Offset-Correction gg = GainCorr, oo = OffsCorr uuuu = unused
31	00C4	RW	wl	ADC1Corr	uuuuggoo	ffff0000	ADC1: see ADC0 above
..	..	..	..	..	..	..	..
37	00DC	RW	wl	ADC7Corr	uuuuggoo	ffff0000	ADC7: see ADC0 above
38	00E0	RW	I	DOWrite32	vvvvvvvvv	00000000	Write: 32 bit DigitalOut Read: 32 bit DDO
39	00E4	RW	wl	DOWriteA	mmvvmvmmv	00000000	Write: mask/value Dout 31...24 / Dout 23...16 Read: 32 bit DDO
3A	00E8	RW	wl	DOWriteB	mmvvmvmmv	00000000	Write: mask/value Dout 15...8 / Dout 7...0 Read: 32 bit DDO
3B	00EC	RO	n/a	----	---	---	Read: 32 bit DDO
3C	00F0	RW	wl	DACWriteBA	bbbbaaaa	00000000	Write: value DACB/DACA Read: 2x16 bit QDAC
3D	00F4	..	..	reserved		rrrrrrrrr	
3E	00F8	..	..	reserved		rrrrrrrrr	
3F	00FC	RW	I	ChanWrite	rrrcvwww	uuuuuuuuu	write: value (channel) c = channel www = value rrrr = reserved
40	0100	R(W)	wl	ADC0Act	ttttvvvv	xxxxxxxxx	ADC0 current value
41	0104	R(W)	wl	ADC1Act	ttttvvvv	xxxxxxxxx	ADC1 current value
...	...	..	..	..	..	..	..
47	011C	R(W)	wl	ADC7Act	ttttvvvv	xxxxxxxxx	ADC7 current value
48	0120	R(W)	wl	DInAAct	ttttvvvv	xxxxxxxxx	Din 15...0 current value
49	0124	R(W)	wl	DInBAct	ttttvvvv	xxxxxxxxx	Din 31...16 current value
4A	0128	R(W)	wl	DOutAAct	ttttvvvv	xxxxxxxxx	Dout 15...0 current value
4B	012C	R(W)	wl	DOutBAct	ttttvvvv	xxxxxxxxx	Dout 31...16 current value
4C	0130	R(W)	wl	DACAAct	ttttvvvv	xxxxxxxxx	DACA current value
4D	0134	R(W)	wl	DACBAct	ttttvvvv	xxxxxxxxx	DACB current value
4E	0138	R(W)	wl	TSAct	ttttvvvv	xxxxxxxxx	Timestamp current value
4F	013C	..	..	reserved		rrrrrrrrr	
50	0140	R(W)	wl	ADC0Last	ttttvvvv	xxxxxxxxx	ADC0 LastDMAValue
51	0144	R(W)	wl	ADC1Last	ttttvvvv	xxxxxxxxx	ADC1 LastDMAValue
...	...	..	..	..	..	..	..

## PCIe Device Access

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
57	015C	R(W)	wl	ADC7Last	ttttvvvv	xxxxxxxxxx	ADC7 LastDMAValue
58	0160	R(W)	wl	DInALast	ttttvvvv	xxxxxxxxxx	Din 15...0 LastDMAValue
59	0164	R(W)	wl	DInBLast	ttttvvvv	xxxxxxxxxx	Din 31...16 LastDMAValue
5A	0168	R(W)	wl	DOutALast	ttttvvvv	xxxxxxxxxx	Dout 15...0 LastDMAValue
5B	016C	R(W)	wl	DOutBLast	ttttvvvv	xxxxxxxxxx	Dout 31...16 LastDMAValue
5C	0170	R(W)	wl	DACALast	ttttvvvv	xxxxxxxxxx	DACA LastDMAValue
5D	0174	R(W)	wl	DACBLast	ttttvvvv	xxxxxxxxxx	DACB LastDMAValue
5E	0178	R(W)	wl	TSLast	ttttvvvv	xxxxxxxxxx	TimeStamp LastDMAValue
5F	017C	..	..	reserved		rrrrrrrrr	
60	0180	RW	wl	ADC0Delta	ddddvvvv	00000000	ADC0 MinDelta for DMA
61	0184	RW	wl	ADC1Delta	ddddvvvv	00000000	ADC1 MinDelta for DMA
...	...	..	..	..	..	..	..
67	019C	RW	wl	ADC7Delta	ddddvvvv	00000000	ADC7 MinDelta for DMA
68	01A0	RW	wl	DInAMask	ddddvvvv	00000000	Din 15...0 Mask for DMA
69	01A4	RW	wl	DInBMask	ddddvvvv	00000000	Din 31...16 Mask for DMA
6A	01A8	RW	wl	DOutAMask	ddddvvvv	00000000	Dout 15...0 Mask for DMA
6B	01AC	RW	wl	DOutBMask	ddddvvvv	00000000	Dout 31...16 Mask for DMA
6C	01B0	RW	wl	DACADelta	ddddvvvv	00000000	DACA MinDelta for DMA
6D	01B4	RW	wl	DACBDelta	ddddvvvv	00000000	DACB MinDelta for DMA
6E	01B8	RW	wl	TSDelta	ddddvvvv	00000000	TS MinDelta for DMA
6F	01BC	RW	wl	unused	-----	00000000	unused
...	...	..	..	..	..	..	..
77	01DC	RW	wl	unused	-----	00000000	unused
78	01E0	RW	wl	Command	cccccccc	00000000	Command
79	01E4	RW	wl	Parameter	pppppppp	00000000	Parameter
7A	01E8	RW	wl	reserved		rrrrrrrrr	
...	...	..	..	..	..	..	..
7E	01F8	RW	wl	reserved		rrrrrrrrr	
7F	01FC	RO	n/a	Result	vvvvvvvv	00000000	Return code
80	0200	RW	wl	reserved		rrrrrrrrr	
...	...	..	..	..	..	..	..
FF	03FC	RW	wl	reserved		rrrrrrrrr	
100	0400	RO	n/a	DMARead0	tttcaaaa	xxxxxxxxxx	Read FIFO ttt = TimeStamp c = channel aaaa = value
101	0404	RO	n/a	DMARead1	tttcaaaa	xxxxxxxxxx	see above
	...	..	..	..	..	..	..
1FF	07FC	RO	n/a	DMARead...	tttcaaaa	xxxxxxxxxx	see above

## 4.7 Register Description

### 4.7.1 Version

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
0	0000	RO	n/a	Version	abcd <sub>15</sub> eeee	12000003	a= type, b= FPGA-version, cd= PCB-version, eeee= feature

#### Description

Variables	Name	Meaning
a	<i>FPGA_Type</i>	1: TCA_ADIO24 0, 2 ... 15: reserved
b	<i>FPGA_Version</i>	1: Prototyp 2: Release 2 0, 3...15: reserved
cd	<i>PCB-Version</i>	PCB-Version (Is written by $\mu$ C on startup.)
eeee	<i>Feature-Flags</i>	see table below

<i>Feature-Flags</i> Bits	Meaning
0	bit = 1: IRQ on FIFO-out
1	bit = 1: IRQ on Compare
2	reserved (bit = 0)
:	
15	reserved (bit = 0)

### 4.7.2 Status

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
1	0004	RO	n/a	Status	vvvvvvvvv	xxxxxxxxx	miscellaneous status bits

#### Description

Bit	Name		Meaning
0	<i>FIFO_Empty</i>	<i>Dout7...0</i>	bit = 1: empty
1	<i>FIFO_Status</i>		<i>FIFO_Status bit 0</i>
2	<i>FIFO_Status</i>		<i>FIFO_Status bit 1</i>
			meaning of <i>FIFO_Status</i> bits see table at page 23
3	<i>FIFO_Empty</i>	<i>Dout15...8</i>	bit = 1: empty
4	<i>FIFO_Status</i>		<i>FIFO_Status bit 0</i>
5	<i>FIFO_Status</i>		<i>FIFO_Status bit 1</i>
			meaning of <i>FIFO_Status</i> bits see table at page 23
6	<i>FIFO_Empty</i>	<i>Dout23...16</i>	bit = 1: empty
7	<i>FIFO_Status</i>		<i>FIFO_Status bit 0</i>
8	<i>FIFO_Status</i>		<i>FIFO_Status bit 1</i>
			meaning of <i>FIFO_Status</i> bits see table at page 23
9	<i>FIFO_Empty</i>	<i>Dout31...24</i>	bit = 1: empty
10	<i>FIFO_Status</i>		<i>FIFO_Status bit 0</i>
11	<i>FIFO_Status</i>		<i>FIFO_Status bit 1</i>
			meaning of <i>FIFO_Status</i> bits see table at page 23
12	<i>FIFO_Empty</i>	<i>DACA</i>	bit = 1: empty
13	<i>FIFO_Status</i>		<i>FIFO_Status bit 0</i>
14	<i>FIFO_Status</i>		<i>FIFO_Status bit 1</i>
			meaning of <i>FIFO_Status</i> bits see table at page 23
15	<i>FIFO_Empty</i>	<i>DACB</i>	bit = 1: empty
16	<i>FIFO_Status</i>		<i>FIFO_Status bit 0</i>
17	<i>FIFO_Status</i>		<i>FIFO_Status bit 1</i>
			meaning of <i>FIFO_Status</i> bits see table at page 23
18	<i>Irq_Status0</i>		bit = 1: active
19	<i>Irq_Status1</i>		bit = 1: active
20	<i>Irq_Status2</i>		bit = 1: active
21	<i>Irq_Status3</i>		bit = 1: active
22...24	unused		'0'
25	<i>DMAFIFFULL</i>		bit = 1: FIFO to local_bus Full
26	<i>DMAFIFEMPTY</i>		bit = 1: FIFO to local_bus Empty
27	<i>RS485InA</i>		Status of Input Signal (Test_only)
28	<i>RS485InB</i>		Status of Input Signal (Test_only)
29	<i>RS485InC</i>		Status of Input Signal (Test_only)
30	<i>RS485InD</i>		Status of Input Signal (Test_only)
31	<i>LVDSIn</i>		Status of Input Signal (Test_only)

The following table shows the meaning of the *FIFO\_Status* bits:

<i>FIFO_Status</i>		FIFO #Data Bytes	State
<i>bit 1</i>	<i>bit 0</i>		
0	0	$0 \leq \#Data < 64$	almost empty
0	1	$64 \leq \#Data < 224$	in range
1	0	$224 \leq \#Data < 255$	almost full
1	1	$\#Data = 255$	FIFO full

### 4.7.3 DDI (Direct Digital In, Inclusive Inversion)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
2	0008	RO	n/a	DDI	vvvvvvvv	xxxxxxxxxx	DirectDigitalIn 32-bit

**Description**

Read back of unsampled digital input port.

Bit no.	Digital Input
0	Din0
1	Din1
...	...
31	Din31

Din31 ... Din24 are internally hard wired to Dout31 .... Dout24.

### 4.7.4 DDO (Direct Digital Out, No Inversion)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
3	000C	RO	n/a	DDO	vvvvvvvv	00000000	DirectDigitalOut 32-bit

#### Description

Current output state.

Bit no.	Digital Output
0	Dout0
1	Dout1
...	...
31	Dout31

Dout31 ... Dout24 are internally hard wired to Din31 .... Din24.

### 4.7.5 DIOMode (Trigger Mode Din/Dout)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
4	0010	RW	bwl	DIOMode	hgfedcba	00000000	TriggerMode Digital In/Out

**Description**

Selection of trigger signals for digital input/output groups.

**Bit Assignment**

Bit no.	Mode
3 ... 0	<i>Mode_Dout 7...0</i>
7 ... 4	<i>Mode_Dout 15...8</i>
11 ... 8	<i>Mode_Dout 23...16</i>
15 ... 12	<i>Mode_Dout 31..24</i>
19 ... 16	<i>Mode_Din 7...0</i>
23 ... 19	<i>Mode_Din 15...8</i>
27 ... 24	<i>Mode_Din 23...16</i>
31 ... 28	<i>Mode_Din 31...24</i>

**Meaning of Parameter *Mode\_...*:**

Value of <i>Mode_...</i>	Meaning
0	'own trigger': trigger next when ready
15 ... 1	TRP 15...1 (timing routing pool line 15...1)

### 4.7.6 DOut Invert (Digital Output Inverted)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
5	0014	RW	bwl	DOutInvert	vvvvvvvv	00000000	DigOut InversionMask 32-bit

#### Description

Bit no.	Digital Output
0	bit = 1: <i>invert physical output Dout0</i>
1	bit = 1: <i>invert physical output Dout1</i>
...	...
31	bit = 1: <i>invert physical output Dout31</i>

### 4.7.7 HSE/LSE (HighSide Enable / LowSide Enable)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
6	0018	RW	bwl	HSE	vvvvvvvv	00000000	DigOut High_Side_Enable 32-bit
7	001C	RW	bwl	LSE	vvvvvvvv	00000000	DigOut Low_Side_Enable 32-bit

#### Description

Configuration of digital out characteristic.

Bit no. of HSE or LSE	Assigned to Digital Output
0	Dout0
1	Dout1
...	...
31	Dout31

HSE and LSE are evaluated together for each physical output:

Register Bits			Pin	Details
HSEx	LSEx	Doutx		
0	0	x	open	I/O is input only
1	0	0	open	High Side Switch: OFF
1	0	1	5V	High Side Switch: ON
0	1	0	open	Low Side Switch: OFF
0	1	1	GND	Low Side Switch: ON
1	1	0	GND	Push/Pull: OFF
1	1	1	5V	Push/Pull: ON

### 4.7.8 FGENAB (DDFS Frequency Generator)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
8	0020	RW	wl	FGENAB	bbbbaaaa	00000000	FreqGenA/B Value 2x16 bit

#### Description

The DDFS frequency generator generates a square wave signal with a 50:50 duty cycle.

aaaa: parameter values of frequency generator A

bbbb: parameter values of frequency generator B

$\text{Frequency} = \frac{32 \text{ MHz}}{\text{Scaler}} \cdot \text{DDFS\_Value}$
--

Bit no. of aaaa and bbbb	Parameter
15, 14	Scaler
13 ... 0	DDFS_Value

Bit no.		Value of Scaler	Range of Frequency
15	14		
0	0	$2^{(15+12)}$	0 ... 3.906 kHz
0	1	$2^{(15+9)}$	0 ... 31.25 kHz
1	0	$2^{(15+6)}$	0 ... 250 kHz
1	1	$2^{(15+3)}$	0 ... 2 MHz

### 4.7.9 FGENCD (Clock Divider)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
9	0024	RW	wl	FGENCD	dddcccc	00000000	FreqGenC/D Value 2x16 bit

#### Description

The frequency generator generates a square wave signal with a 50:50 duty cycle.

cccc: *Scaler* of frequency generator C

ddd: *Scaler* of frequency generator D

$\text{Frequency} = \frac{16 \text{ MHz}}{\text{Scaler}}$
---

**Note:**

Do not use *Scaler* < 16!

### 4.7.10 SyncOut (External Trigger Output/Input)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
A	0028	RW	bwl	SyncOut	ddccbbaa	00000000	Trigger-Output-Select 4x 8 bits (RS-485)

#### Description

Selects the source for SyncOut0...3 .

Bytes of Register SyncOut	Trigger Output Select
aa	<i>TrigOut_SYNC0</i>
bb	<i>TrigOut_SYNC1</i>
cc	<i>TrigOut_SYNC2</i>
dd	<i>TrigOut_SYNC3</i>

Bits of <i>TrigOut_SYNCx</i>	Meaning
4 ... 0	source selector 0...31, selects TRP (Timing Routing Pool signal; see table at page 16)
5	bit = 1: Invert Output
6	bit = 1: Invert Input
7	bit = 1: Enable Output

### 4.7.11 DORout (Trigger Output/Input, DigOut-Route)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
B	002C	RW	bwl	DORout	rrrraaee	00000000	aa= Selector DigOut to TRP0, ee= TriggerOut-Select (LVDS Backplane) rrrr=reserved

**Description**

Multiplexes a discrete line to TRP0.

Bytes of Register DORout	Meaning
aa	<i>Selector_DigOut_to_TRP0</i>
ee	<i>TriggerOut-Select LVDS</i>
rrrr	<i>reserved</i>

Bits of Parameter <i>Selector_DigOut_to_TRP0</i>	Meaning
4 ... 0	Selector Digital Out 31...0 (Dout31...0) to TRP0
5	reserved, write 0
6	reserved, write 0
7	reserved, write 0

Parameter *TriggerOut-Select LVDS* (**DO NOT USE!**):

Bit 7...0: SyncOut for LVDS (backplane) transceiver.

Bits of <i>TriggerOut-Select LVDS</i>	Meaning
4 ... 0	source selector 31...0, selects TRP (Timing Routing Pool signal; see table at page 16)
5	bit = 1: Invert Output
6	bit = 1: Invert Input
7	bit = 1: Enable Output

**Attention:** The routing via backplane is untested!

### 4.7.12 DInInvert (Digital Input Inverted)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
C	0030	RW	bwl	DInInvert	v v v v v v v v	00000000	DigIn InversionMask 32 bit

#### Description

Bit no.	Digital Input
0	bit = 1: invert physical input Din0
1	bit = 1: invert physical input Din1
...	...
31	bit = 1: invert physical input Din31

### 4.7.13 DInFilter (Filter Digital Input)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
D	0034	RW	bwl	DInFilter	vvvvvvvv	00000000	DigIn FilterEnable 32 bit

#### Description

Bit no.	Digital Input
0	bit = 1: enable filter on physical input Din0
1	bit = 1: enable filter on physical input Din1
...	...
31	bit = 1: enable filter on physical input Din31

Value of *enable filter on physical input DinX* = 1 : filter enabled  
 (3 successive samples set/reset state)

### 4.7.14 ADCMode (Trigger Mode ADC7...0)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
E	0038	RW	bwl	ADCMode	hgfedcba	00000000	Trigger Mode for ADC7 ... ADC0

#### Description

Selects trigger condition for ADC7...0.

Nibbles of register ADCMode	Meaning
a	<i>TriggerMode ADC0</i>
b	<i>TriggerMode ADC1</i>
c	<i>TriggerMode ADC2</i>
d	<i>TriggerMode ADC3</i>
e	<i>TriggerMode ADC4</i>
f	<i>TriggerMode ADC5</i>
g	<i>TriggerMode ADC6</i>
h	<i>TriggerMode ADC7</i>

Format of *TriggerMode ADCx* see Register 'DIOMode' at page 26.

### 4.7.15 DIVMode (DMA-Mode, Timer Select, Interrupt Enable)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
F	003C	RW	bwl	DIVMode	missoxtt	00000000	m = DMA-Mode i = IrqEnable3...0 ss = TS-Counter-Select o = Master of DMA-Out x = unused tt = Output-Table Mode

#### Description

Content of Register DIVMode	Width	Meaning
m	4 bits	<i>DMA-Mode</i>
i	4 bits	<i>IrqEnable3...0</i>
ss	8 bits	<i>TS-Counter Select</i>
o	4 bits	<i>Master of DMA-Out</i>
x	4 bits	unused
tt	8 bits	<i>Output-Table Mode</i>

#### DMA-Mode

Bits 31...28 of register DIVMode are used for the selection of the *DMA-Mode*.

Value of parameter <i>DMA-Mode</i>	(Value of bits 31...28 of DIVMode)	Meaning
0	0000	DMA-disabled (clear read- and write-counter)
1	0001	DMA-enabled (FIFO-mode)
2	0010	Memory-Mode, one-shot
3	0011	Memory-Mode, continuous
4...15	0010...1111	reserved

#### IrqEnable3...0

Bits 27...24 of register DIVMode are used for *IrqEnable3...0*.

Bits of parameter <i>IrqEnable 0...3</i>	(Bits of DIVMode)	Meaning	Usage
0	24	bit = 1: Enable Irq0	FIFO-Irq
1	25	bit = 1: Enable Irq1	Compare-Irq
2	26	bit = 1: Enable Irq2	unused
3	27	bit = 1: Enable Irq3	unused

**TS-Counter Select** (*Timestamp Counter Select*)

Bits 18...16 of register DIVMode are used for *TS-Counter Select*.  
 Bits 23...19 of register DIVMode are reserved for future use.

Value of parameter <i>TS-Counter Select</i>	(Value of bits 18...16 of DIVMode)	Timestamp counter resolution
0	000	any write to DMA-FIFO (sequence counter)
1	001	0.25 $\mu$ s
2	010	0.5 $\mu$ s
3	011	1 $\mu$ s
4	100	2 $\mu$ s
5	101	4 $\mu$ s
6	110	8 $\mu$ s
7	111	128 $\mu$ s

**Master of DMA-Out**

Bits 14...12 of register DIVMode are used for *Master of DMA-Out*  
 Bit 15 of register DIVMode is reserved for future use.

Value of parameter <i>Master of DMA-Out</i>	(Value of bits 14...12 of DIVMode)	Selected DMA master
0	000	no DMA-Out
1	001	Dout 08...01 is master
2	010	Dout 16...09 is master
3	011	Dout 24...17 is master
4	100	Dout32...25 is master
5	101	DACA is master
6	110	DACB is master
7	111	DACB is master (reserved)

**Output-Table Mode**

Bits 5...0 of register DIVMode are used for *Output-Table Mode*

Bits 11...6 of register DIVMode are reserved for future use.

Bits of parameter <i>Output-Table Mode</i>	(Bits of DIVMode)	Assignment of the Outputs
0	0	Dout 08...01
1	1	Dout 16...09
2	2	Dout 24...17
3	3	Dout32...25
4	4	DACA
5	5	DACB

Value of <i>Output-Table Mode</i> Bits	Meaning
0	use Output-FIFO as FIFO
1	output always the last 256 entries ('Table-Mode')

### 4.7.16 Auxreg (Trigger Mode DACA/B)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
10	0040	RW	bwl	AuxReg	rrrrdcbba	00000000	ba= Trigger Mode DACA/B dc= Enable bits Out-FIFO-Irq 5...0

#### Description

Bytes of Register AuxReg	Meaning
ba	<i>Trigger Mode DACA/B</i>
dc	<i>En_Out-FIFO-Irq</i>
rrrr	reserved

Bits of Parameter <i>TriggerMode DACA/B</i>	Meaning
3 ... 0	<i>Trigger Mode_DACA</i>
7 ... 4	<i>Trigger Mode_DACB</i>

Format of *Mode\_DACA/B* see register 'DIOMode' on page 26.

Bits of Parameter <i>En_Out-FIFO-Irq</i>	(Bits of AuxReg)	Meaning
0	8	bit = 1: Enable Irq on FIFO Dout 7...0 *) <sup>1</sup>
1	9	bit = 1: Enable Irq on FIFO Dout 15...8 *) <sup>1</sup>
2	10	bit = 1: Enable Irq on FIFO Dout 23...16 *) <sup>1</sup>
3	11	bit = 1: Enable Irq on FIFO Dout 31...24 *) <sup>1</sup>
4	12	bit = 1: Enable Irq on FIFO DACA *) <sup>1</sup>
5	13	bit = 1: Enable Irq on FIFO DACB *) <sup>1</sup>

<sup>\*)1</sup> active in FIFO status = '00' (empty/almost empty) see register 'Status' at page 22.  
These interrupts are routed to IRQ0.

### 4.7.17 QDac (Direct Read Back DAC A/B)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
11	0044	RO	n/a	QDac	bbbbaaaa	00000000	DirectDacValue Corrected DAC_B/A 2x16 bit

#### Description

QDac returns the physical DAC-value after gain/offset correction (Out\_Value).

Words of Register QDac	Meaning
aaaa	<i>DirectDacValue DACA (16 bit)</i>
bbbb	<i>DirectDacValue DACB (16 bit)</i>

$V_{out} = 10.24V \cdot \frac{\text{Value}}{0x8000}$
--

### 4.7.18 DACCorr (Gain/Offset Correction DAC B/A)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
12	0048	RW	bwl	DACCorr	ddccbbaa	00000000	Gain/Offset-Correction DAC_B/A

**Note:** The analog outputs are factory calibrated. The calibration data is loaded to this FPGA-register at every start-up.

#### Description

The user can change the factory calibration data at the register DACCorr. The changed contents of DACCorr are lost at reset or power-off. For permanent storage the contents of DACCorr can be stored in a EEPROM. Restoration of the factory calibration is possible as well. For details please refer chapter '4.7.29 Command Register' from page 63 on.

Bytes of Register DACCorr	Meaning
aa	<i>Offset DAC_A</i>
bb	<i>Gain DAC_A</i>
cc	<i>Offset DAC_B</i>
dd	<i>Gain DAC_B</i>

Correction formula:

$$\text{Out\_Value} = (\text{In\_Value} + \text{Offset}) \cdot \left(1 + \frac{\text{Gain}}{2^{14}}\right)$$

Correction coefficients are signed.

Out\_Value is limited to 0x7FFF / 0x8000 on overflow.

### 4.7.19 EVENT

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
13	004C	RW	bwl	EVENT	rrrrrvvvv	00000000	PCI-Write: Set SW-Event 31...0 SPI-Write: Ack SPI-Irq All_Read: SPI-Irq status 15...0

#### Description

The function of the register EVENT depends on the write or read access type.

#### PCI-Write

Bit no.	Meaning
0	bit = 1: generate SW-Event 0
1	bit = 1: generate SW-Event 1
...	...
16	bit = 1: generate SW-Event 16
...	...
31	bit = 1: generate SW-Event 31

#### SPI-Write (internal use only)

Bit no.	Meaning
0	bit = 1: Ack SPI-Irq0
1	bit = 1: Ack SPI-Irq1
...	...
15	bit = 1: Ack SPI-Irq15
16	bit = 1: SW-Event16
...	...
31	bit = 1: SW-Event31

#### Read

Bit no.	Meaning
15 ... 0	SPI-Irq 15...0
31 ... 16	reserved

**Event Assignment**

Event	Source
SW-Event 0	Source for TRP(10)
SW-Event 1	Source for TRP(11)
SW-Event 2...7	reserved
SW-Event 8	SPI-Irq8 (Command-Irq, see page 63)
...	...
SW-Event 15	SPI-Irq15 (future use)
SW-Event 16...29	reserved
SW-Event 30	Source for TRP(12)
SW-Event 31	Source for TRP(13)

SW-Event 0, 1, 30, 31 are generating a 250 ns pulse on the trigger lines.

### 4.7.20 DMAMode (DMA enable/disable)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
14	0050	RW	bwl	DMAMode	ddddeeee	00000000	dddd = DMA-Write Disable eeee = DMA-TimeEnable 16-Bit

#### Description

Content of Register DMAMode	Width	Meaning
dddd	16 bits	<i>DMA-Write Disable</i>
eeee	16 bits	<i>DMA-Time Enable 16-Bit</i>

#### Combined evaluation of *DMA-Write Disable* (dddd) and *DMA-Time Enable 16-Bit* (eeee);

bits 'd' (of <i>DMA-Write Disable</i> )	bits 'e' (of <i>DMA-Time Enable 16-Bit</i> )	Write to DMA FIFO	Update xxxAct	Cycles
0	0	on valid Event	always	7
0	1	always	always	4
1	0	never	never	0
1	1	never	always	4

For more details about the combined evaluation of *DMA-Write Disable* (dddd) and *DMA-Time Enable 16-Bit* (eeee) see page 65.

Each of the 16 bits of the parameters is assigned to one DMA channel (see following table).

**DMA-Write Disable** (dddd), **DMA-Time Enable 16-Bit** (eeee)

The 16 bits of the parameters are assigned to the DMA channels:

Bits of parameter <i>DMA-Write Disable</i> 'dddd'	Bits of parameter <i>DMA-Time Enable 16-Bit</i> 'eeee'	Assigned to DMA channel
0	0	ADC0
1	1	ADC1
2	2	ADC2
3	3	ADC3
4	4	ADC4
5	5	ADC5
6	6	ADC6
7	7	ADC7
8	8	Din 15...0
9	9	Din 31...16
10	10	Dout 15...0
11	11	Dout 31...16
12	12	DACA
13	13	DACB
14	14	TimeStamp
15	15	unused

### 4.7.21 Compare

#### Timestamp-Compare-Irq

The lower 16 bits of the Timestamp Counter (*Current Timestamp Counter*) are compared with a Compare Value (*Current Compare Value*). If they match, the local Irq #1 is set.

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
15	0054	R	-	Compare Value	ttttcccc	00000000	tttt = Current TS Counter cccc = Current Compare Value
		W	bwl	Next	rrrrnnnn	00000000	nnnn = unsigned Delta Compare Value

#### Description

Content of Register Compare	Width	Meaning
tttt	16 bits	<i>Current Timestamp Counter</i>
cccc	16 bits	<i>Current Compare Value</i>
rrrr	16 bits	<i>reserved</i>
nnnn	16 bits	<i>Delta Compare Value</i>

**On Read:** *Current Timestamp Counter*

Bits 31...16 of register 'Compare' return the *Current Timestamp Counter*. These bits contain the current lower 16 bits of the Timestamp Counter.

**Current Compare Value**

Bits 15...0 of register 'Compare' return the *Current Compare Value*.

**On Write:** Acknowledge the pending compare interrupt.

$$Current\_Compare\_Value = Current\_Compare\_Value + Delta\_Compare\_Value$$

### 4.7.22 ADC1Corr ... ADC8Corr (ADC Gain and Offset)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
30	00C0	RW	wl	ADC0Corr	uuuuggoo	0000	ADC0: Gain/Offset-Correction gg=GainCorr, oo=OffsCorr
31	00C4	RW	wl	ADC1Corr	uuuuggoo	0000	ADC1: see above
..	..	..	..	..	..	..	..
37	00DC	RW	wl	ADC7Corr	uuuuggoo	0000	ADC7: see above

**Note:** The analog inputs are factory calibrated. The calibration data is loaded to these FPGA-registers at every start-up.

#### Description

The user can change the factory calibration data at the registers ADC0Corr...ADC7Corr. The changed contents of the ADCxCorr registers are lost at reset or power-off. For permanent storage the contents of the ADCxCorr registers can be stored in a EEPROM. Restoration of the factory calibration is possible as well. For details please refer chapter '4.7.29 Command Register' from page 63 on.

Bytes of Register ADCxCorr	Meaning
oo	<i>Offset</i>
gg	<i>Gain</i>
uuuu	unused

Correction formula:

$$\text{Out\_Value} = (\text{In\_Value} + \text{Offset}) \cdot \left(1 + \frac{\text{Gain}}{2^{14}}\right)$$

Correction coefficients are signed.

Out\_Value is limited to 0x7FFF / 0x8000 on overflow.

### 4.7.23 DOWriteX (Digital Out Mask)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description	
38	00E0	RW	I	DOWrite32	vvvvvvvvv	00000000	Write: 32 bit DigitalOut	Read: 32 bit DDO
39	00E4	RW	wl	DOWriteA	mmvvmvmmv	00000000	Write: Mask/Value Dout31...24/Dout23...16	Read: 32 bit DDO
3A	00E8	RW	wl	DOWriteB	mmvvmvmmv	00000000	Write: Mask/Value Dout15...08/Dout07...00	Read: 32 bit DDO

#### Description

The setting of the digital outputs of a channel group is always done via the according FIFOs:

Group	Outputs
1	Dout07...Dout00
2	Dout15...Dout08
3	Dout23...Dout16
4	Dout31...Dout24

Within the group the eight according mask/value data are stored. Triggered by the according timing signal (DIOMode) the physical outputs are set.

DOWrite32 sets all 32 outputs at once.

DOWriteA/B expects mask and value in two 16-bit data words.

For setting an output bit, the related mask bit must be '1'.

If a mask bit = '0', the output bit stays unchanged.

### 4.7.24 DACWriteBA (Write DAC)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
3C	00F0	RW	wl	DACWriteBA	bbbbaaaa	00000000	Write: Value DACB/DACA Read: 2x16 bit DACA/BAct

#### Description

Words of Register DACWriteBA	Meaning
aaaa	<i>Value DACA</i>
bbbb	<i>Value DACB</i>

The values for DACA/DACB have to be written in the according FIFO.

Triggered by the according timing signal (AuxReg) the data correction is executed and the physical output of the DAC is set.

<i>Value</i>
$V_{out} = 10.24V \cdot \frac{\text{-----}}{0x8000}$

### 4.7.25 ChanWrite

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
3F	00FC	WO	I	ChanWrite	rrrrrcvvvv	uuuuuuuuu (undefined)	write: Value(Channel)

#### Description

An alternative method for writing output data (DMA-write).

Content of Register ChanWrite	Meaning
c	<i>Channel</i>
vvvv	<i>Value</i>
rrrr	reserved

Value of Channel [Hex]	Meaning
0	Group0 (Dout07...Dout00)
1	Group1 (Dout15...Dout08)
2	Group2 (Dout23...Dout16)
3	Group3 (Dout31...Dout24)
4	DACA
5	DACB
F...6	do not use

Value of Value [Hex]	Meaning
0000...FFFF	<i>Output Value Group 0 (Dout07...Dout00)</i>
0000...FFFF	<i>Output Value Group 1 (Dout15...Dout08)</i>
0000...FFFF	<i>Output Value Group 2 (Dout23...Dout16)</i>
0000...FFFF	<i>Output Value Group 3 (Dout31...Dout24)</i>
0000...FFFF	<i>Output Value DACA</i>
0000...FFFF	<i>Output Value DACB</i>

## 4.7.26 Current Register Values

The current register value can be read directly at any time, but attention should be paid to the duration of a read cycle via PCIe, which can last up to 2...3  $\mu$ s !  
Therefore the operation via DMA is highly recommended to achieve higher data rates.

Write access is possible. After the following trigger event the new values will be written.

**Note:**

The data of the registers described in this chapter will be written in the DMA-FIFO, if DMA mode is configured.

### 4.7.26.1 ADCxAct (Current Analog Input Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
40	0100	R(W)	wl	ADC0Act	ttttvvvv	xxxxxxxx	ADC0 current value
41	0104	R(W)	wl	ADC1Act	ttttvvvv	xxxxxxxx	ADC1 current value
...	...	..	..	..	..	..	..
47	011C	R(W)	wl	ADC7Act	ttttvvvv	xxxxxxxx	ADC7 current value

#### Description

Content of Register ADCxAct	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (current ADC value)

$$V_{in} = 10.24V \cdot \frac{Value}{0x8000}$$

### 4.7.26.2 DInAAct, DInBAct (Current Digital Input Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
48	0120	R(W)	wl	DInAAct	ttttvvvv	xxxxxxxxxx	Din15...0 current value
49	0124	R(W)	wl	DInBAct	ttttvvvv	xxxxxxxxxx	Din31...16 current value

#### Description

Content of Register DInA/B Act	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (current digital input value)

### 4.7.26.3 DOutAAct, DOutBAct (Current Digital Output Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
4A	0128	R(W)	wl	DOutAAct	ttttvvvv	xxxxxxxxxx	Dout15..0 current value
4B	012C	R(W)	wl	DOutBAct	ttttvvvv	xxxxxxxxxx	Dout31..16 current value

#### Description

Content of Register DOutA/B Act	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (current digital output value)

#### 4.7.26.4 DACAAct, DACBAct (Current Analog Output Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
4C	0130	R(W)	wl	DACAAct	ttttvvvv	xxxxxxxx	DACA current value
4D	0134	R(W)	wl	DACBAct	ttttvvvv	xxxxxxxx	DACB current value

#### Description

Content of Register DAC/A/B Act	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (current analog output value)

#### 4.7.26.5 TSAct (Current Timestamp Counter Value)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
4E	0138	R(W)	wl	TSAct	ttttvvvv	xxxxxxxx	TimeStamp current value

#### Description

Content of Register TSAct	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (upper 16 bits of the 27-bit TS-counter)

## 4.7.27 Last DMA Values

### 4.7.27.1 ADC1Last... ADC8Last (Last Analog Input DMA Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
50	0140	R(W)	wl	ADC0Last	ttttvvvv	xxxxxxxx	ADC0 LastDMAValue
51	0144	R(W)	wl	ADC1Last	ttttvvvv	xxxxxxxx	ADC1 LastDMAValue
...	...	..	..	..	..	..	..
57	015C	R(W)	wl	ADC7Last	ttttvvvv	xxxxxxxx	ADC7 LastDMAValue

#### Description

Last value written to DMA-In FIFO.

Content of Register ADCxLast	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (ADC last DMA value)

### 4.7.27.2 DInALast, DInBLast (Last Digital Input DMA Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
58	0160	R(W)	wl	DInALast	ttttvvvv	xxxxxxxx	Din15...0 LastDMAValue
59	0164	R(W)	wl	DInBLast	ttttvvvv	xxxxxxxx	Din31...16 LastDMAValue

#### Description

Last value written to DMA-In FIFO.

Content of Register DInA/B Last	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (digital input last DMA value)

### 4.7.27.3 DOutALast, DOutBLast (Last Digital Output DMA Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
5A	0168	R(W)	wl	DOutALast	ttttvvvv	xxxxxxxx	Dout15...0 LastDMAValue
5B	016C	R(W)	wl	DOutBLast	ttttvvvv	xxxxxxxx	Dout31...16 LastDMAValue

#### Description

Last value written to DMA-In FIFO.

Content of Register DOutA/B Last	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (digital output last DMA value)

#### 4.7.27.4 DACALast, DACALast (Last Analog Output DMA Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
5C	0170	R(W)	wl	DACALast	ttttvvvv	xxxxxxxx	DACA LastDMAValue
5D	0174	R(W)	wl	DACBLast	ttttvvvv	xxxxxxxx	DACB LastDMAValue

#### Description

Last value written to DMA-In FIFO.

Content of Register DACA/B Last	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (analog output last DMA value)

#### 4.7.27.5 TSLast (Last TS-Counter DMA Values)

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
5E	0178	R(W)	wl	TSLast	ttttvvvv	xxxxxxxx	TimeStamp Last DMA Value

#### Description

Last value written to DMA-In FIFO.

Content of Register TSLast	Meaning
tttt	<i>TS_counter</i> (lower 16 bits of the 27-bit TS-counter)
vvvv	<i>Value</i> (upper 16 bits of the 27-bit TS-counter)

## 4.7.28 Difference Values for Trigger Conditions

### 4.7.28.1 ADC1Delta ... ADC8Delta

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
60	0180	RW	wl	ADC0Delta	ddddvvvv	00000000	ADC0 MinDelta for DMA
61	0184	RW	wl	ADC1Delta	ddddvvvv	00000000	ADC1 MinDelta for DMA
...	...	..	..	..	..	..	..
67	019C	RW	wl	ADC7Delta	ddddvvvv	00000000	ADC7 MinDelta for DMA

#### Description

Content of Register ADCxDelta	Meaning
dddd	do not care (always write '0')
vvvv	Value (minimum difference value of analog input)

In the registers xxxxDelta the trigger conditions for writing the data in the DMA-FIFO are defined:

$$\text{Trigger} = \text{ABS}(\text{ADCxAct} - \text{ADCxLast}) > \text{ADCxDelta}$$

For more information see DMA-Mode from page 69 on.

### 4.7.28.2 DInA/BMask

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
68	01A0	RW	wl	DInAMask	ddddvvvv	00000000	Din15..0 Mask for DMA
69	01A4	RW	wl	DInBMask	ddddvvvv	00000000	Din31..16 Mask for DMA

#### Description

Content of Register DInA/B Mask	Meaning
dddd	do not care (always write '0')
vvvv	<i>Value</i> (mask values for digital inputs)

In the registers DInA/BMask the trigger conditions for writing the data in the DMA-FIFO are defined:

$$\text{Trigger} = (\text{DInxAct} \text{ EXOR } \text{DInxLast}) \text{ AND NOT}(\text{DInxMask}) \neq 0\text{x}0000$$

For more information see DMA-Mode from page 69 on.

### 4.7.28.3 DOutA/BMask

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
6A	01A8	RW	wl	DOutAMask	ddddvvvv	00000000	Dout15...0 Mask for DMA
6B	01AC	RW	wl	DOutBMask	ddddvvvv	00000000	Dout31...16 Mask for DMA

#### Description

Content of Register DOutA/B Mask	Meaning
dddd	do not care (always write '0')
vvvv	<i>Value</i> (mask values for digital outputs)

In the registers DOutA/BMask the trigger conditions for writing the data in the DMA-FIFO are defined:

$\text{Trigger} = (\text{DOutxAct} \text{ EXOR } \text{DOutxLast}) \text{ AND NOT}(\text{DOutxMask}) \neq 0\text{x}0000$
--

For more information see DMA-Mode from page 69 on.

#### 4.7.28.4 DACA/BDelta

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
6C	01B0	RW	wl	DACADelta	dddvvvv	00000000	DACA MinDelta for DMA
6D	01B4	RW	wl	DACBDelta	dddvvvv	00000000	DACB MinDelta for DMA

#### Description

Content of Register DACA/B Delta	Meaning
ddd	do not care (always write '0')
vvv	<i>Value</i> (minimum difference value of analog output)

In the registers DACA/BDelta the trigger conditions for writing the data in the DMA-FIFO are defined:

$$\text{Trigger} = \text{ABS}(\text{DACxAct} - \text{DACxLast}) > \text{DACxDelta}$$

For more information see DMA-Mode from page 69 on.

### 4.7.28.5 TSDelta

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
6E	01B8	RW	wl	TSDelta	dddvvvv	00000000	TS MinDelta for DMA

#### Description

Content of Register TSDelta	Meaning
ddd	do not care (always write '0')
vvv	<i>Value</i> (minimum difference value of analog output)

In the register TSDelta the trigger condition for writing the data in the DMA-FIFO are defined:

$\text{Trigger} = \text{ABS}(\text{TSAct} - \text{TSLast}) > \text{TSDelta}$
--

For more information see DMA-Mode from page 69 on.

### 4.7.29 Command Register

Reg No [Hex]	Addr. [Hex]	Attr.	Write Acc.	Name	Format	Default	Description
78	01E0	RW	wl	Command	cccccccc	00000000	Command
79	01E4	RW	wl	Parameter	pppppppp	00000000	Parameter
7A	01E8	RW	wl	reserved		rrrrrrrr	
...	...	..	..	..	..	..	..
7E	01F8	RW	wl	reserved		rrrrrrrr	
7F	01FC	RO	n/a	Result	vvvvvvvv	00000000	Return code

#### Description

Setting Command Register [Hex]	Defines (examples)	Parameter required?	Command
0	AMC_ADIO24CMD_NOP	no	no operation, returns OK
1	AMC_ADIO24CMD_FRU_RESTORE	no	restore factory FRU information
2	AMC_ADIO24CMD_FRU_CLEAR	no	delete FRU information
3	AMC_ADIO24CMD_CAL_READ	no	read calibration data of analog inputs and outputs from EEPROM and write to FPGA register
4	AMC_ADIO24CMD_CAL_SAVE	no	read calibration data of analog inputs and outputs from FPGA registers and write to EEPROM
5	AMC_ADIO24CMD_CAL_RESTORE	no	restore factory defaults of calibration data and write them to EEPROM and FPGA register
6	AMC_ADIO24CMD_ALL_RESTORE	no	restore factory default state of FPGA-image, FRU and calibration
0x07...0x40	unused	-	-
0x41...0x4F	reserved	reserved	reserved for factory service purposes
0x50...0xFFFFFFFF	unused	-	-

Return Codes of Result Register [Hex]	Defines (examples)	Status message
0	AMC_ADIO24CMD_RESULT_OK	OK
FFFFFFFF	AMC_ADIO24CMD_RESULT_INVALIDCMD	invalid command
FFFFFFFE	AMC_ADIO24CMD_RESULT_INVALIDPARA	invalid parameter
FFFFFFFD	AMC_ADIO24CMD_RESULT_NOCALIBDATA	no valid calibration data found

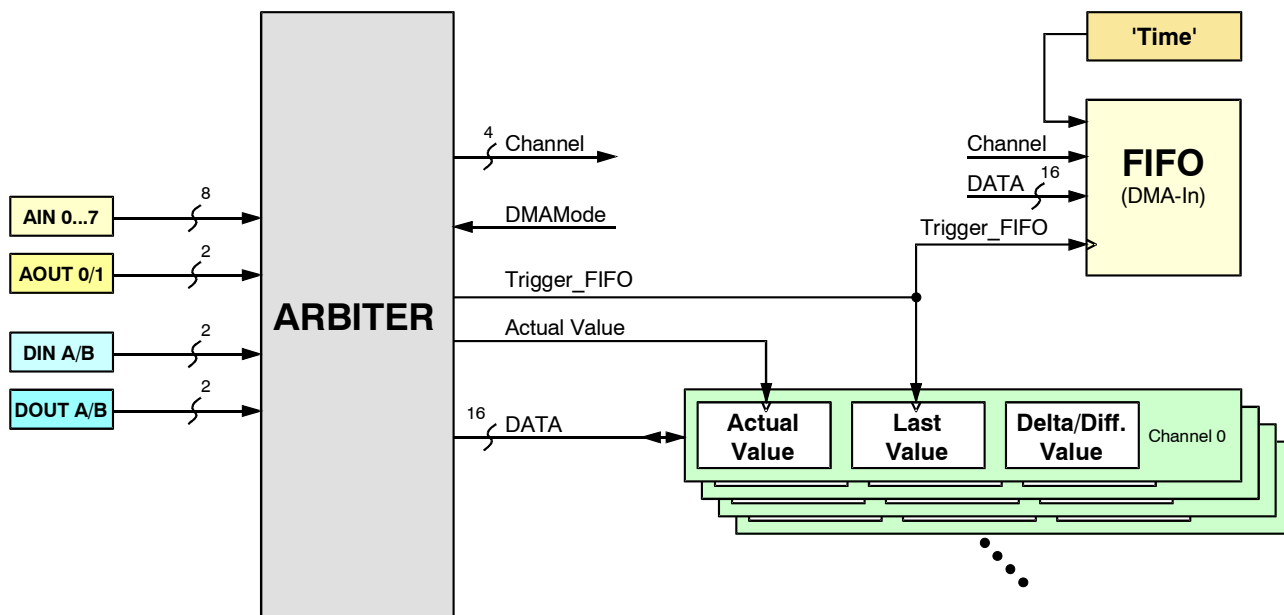
### Command Setting Procedure:

The execution of a command is controlled by the firmware. In order to check the success of the command's execution the register EVENT has to be polled:

1. Write command value to register 'Command' (register no. 0x78)
2. Set bit 8 in register 'EVENT' (register no. 0x13) (e.g. by writing value 0x100)
3. Poll register 'EVENT' until value of bit 8 is returned to '0' (i.e. firmware is ready)
4. Evaluate result code from register 'Result' (register no. 0x7F)

### 4.8 I/O Data Processing

Each I/O group generates a Ready signal after a trigger event occurs. These Ready signals are routed to an arbiter, which carries out the data processing according to signal priority (e.g. support of 'xxxACT', resp. check of valid DMA-In conditions).



The following maximum occurring band width is required:

- ADC: 8 x 200 KSPS = 1.2 MSPS
- DAC: 2 x 600 KSPS = 1.2 MSPS
- DIN: 2 x 1 MSPS = 2.0 MSPS
- DOUT: 2 x 1 MSPS = 2.0 MSPS

This results in an average data rate of 6.4 MSPS, resp. 25.6 MB/s.

A complete processing of all input requests (check of DMA-In conditions) without loss can be guaranteed up to an average data rate of approx. 4 MSPS.

Therefore the following data processing can be configured (via bits 'e' and 'd' in register 'DMAMode'; channel assignment as described in chapter '4.7.20 DMAMode (DMA enable/disable)' from page 44 on).

bits 'd' (of DMA-Write Disable)	bits 'e' (of DMA-Time Enable 16-Bit)	Write to DMA FIFO	Update xxxAct	Cycles
0	0	on valid Event	always	7
0	1	always	always	4
1	0	never	never	0
1	1	never	always	4

To guarantee a loss-free processing, the following condition must be valid:

$$\text{Sum of all channels (Sample\_rate(ch) \cdot \text{Cycles(ch,mode)})} \ll 32 \text{ MHz}$$

## 4.9 DMA Access

### 4.9.1 DMA Input Data

#### 4.9.1.1 Read Data FIFO

If a valid DMA-FIFO trigger condition occurs ('Trigger\_FIFO'), the data is stored in combination with time stamp and channel number in a 32-bit wide, 256 entries deep, FIFO. From the FIFO the DMA unit of the PCI bridge can transfer the entries to the CPU memory.

If at least 16 entries are stored in the FIFO (FPGA to PCI) the signal 'DREQ0' at the PCI bridge becomes active and remains active as long as more than 4 entries are stored in the FIFO.

Memory Mode: Read of address returns data of write pointer. No incrementation of read pointer (for test purpose only).

FIFO Mode: A read of any FIFO address returns FIFO data and increments the read pointer if the FIFO is not empty.

#### Read Data FIFO Memory Area

Addr. [Hex]	Attr.	Acc.	Name	Format	Default	Description	
0400	RO	n/a	DMARead0	tttcaaaa	xxxxxxxxxx	Read FIFO	ttt= TimeStamp c= Channel aaaa= Value
0404	RO	n/a	DMARead1	tttcaaaa	xxxxxxxxxx	see above	
...	..	..	..	..	..	..	
07FC	RO	n/a	DMARead...	tttcaaaa	xxxxxxxxxx	see above	

#### Description

Content of Register DMAReadx	Meaning
ttt	<i>Timestamp</i> (11+1 bits)
c	<i>DMA channel no</i> (4 bits)
aaaa	<i>Value</i> (16 bits)

### Timestamp Generation

To enable timestamps > 12 bits the following logic is implemented:

The local timestamp counter (TS counter) is realized as a combination of a 11-bit counter (lower count) and a 16-bit counter (upper count).

Within the 12-bit area of the process data timestamp the 11-bit count value plus a 'pending' flag is transmitted. This 'pending' flag is set, if the local 11-bit counter overruns and is reset, if the 16-bit counter is written in the FIFO.

If trigger conditions apply simultaneously, the 16-bit counter (DMA-channel = 0xE) gets the worst priority.

To determine timestamp values > 11 bit, the application just has to execute the following operation:

$$\text{TimeStampLong} = \text{LocalUpperCount} \ll 11 + \text{DMAData} \gg 20$$

If the DMA-channel = 0xE, the LocalUpperCount is taken.

### DMA Channel No

DMA channel no [Hex]	I/O Group
0	ADC0
1	ADC1
2	ADC2
3	ADC3
4	ADC4
5	ADC5
6	ADC6
7	ADC7
8	Din15..0
9	Din31..16
A	Dout15..0
B	Dout31..16
C	DACA
D	DACB
E	TimeStamp
F	FIFO-empty

### 4.9.1.2 Setting DMA Operation

The DMA operation mode is configured via the register `DIVMode` (see page 36):

#### 0. DMA disabled

read-address = write-address = 0, DMA disabled

#### 1. DMA-enabled (FIFO-mode)

A valid trigger condition causes an entry in the circular buffer and the write address (WA) is incremented as long as  $WA-RA < 255$  (RA = read address).

Each long word read access in the range of `0x0400 ... 0x07FC` returns the content of RA. RA is incremented as long as  $WA-RA > 0$ . If the FIFO is empty (WA=RA) the read access returns `0xFFFFwwrr` (ww=WA, rr=RA).

#### 2. Memory-Mode, one-shot

Writing data as in FIFO-mode (1.), but the read address is determined from the access address. The write address can not exceed 255 (One-Shot-Mode).

#### 3. Memory-Mode, continuous

As 2., but the write address wraps from 255 to 0.

For more information on the DMA at PCIe read the manual of the PCI bridge PEX 8311 (<http://www.plxtech.com/products/expresslane/pex8311>).

## 4.9.2 DMA Output Data

Beside by CPU writes output data can be set by the DMA controller of the PCI bridge (e.g if digital-out patten or analog wave forms are required).

For this the second DMA channel ('DREQ1') of the PCI bridge can be used. The FIFO state of one of the output groups can be evaluated as the DMA requester (see parameter *Master of DMA-Out* in register DIVMode (reg. no. 0xF)).

DREQ1 is set according to the following conditions:

**DREQ1 is active,** if the FIFO state is '00' (< 64 entries in FIFO)

**DREQ1 stays active,** as long as the FIFO state is '01' (< 244 entries in the FIFO)

The according register 'DOWriteA/B', resp. 'DACWriteA/B' can be used as write addresses. If several output groups are controlled BY THE SAME TRIGGER LINE, they can be supported by the same DMA. Therefore one of the output channels has to be select as 'master'. Note, that the data of the 'slave' channels are included in the DMA stream, as well. The register 'Chanwrite' (reg. no. 0x3F) has to be used as write address.

For more information on the DMA at PCIe read the manual of the PCI bridge PEX 8311 (<http://www.plxtech.com/products/expresslane/pex8311>).

## 5. Technical Data

### 5.1 General Technical Data

Power supply voltage	nominal voltage: 3.3 V ( $I_{3.3VMPMAX} = 60 \text{ mA}$ ), 12 V ( $I_{12VTYPICAL} = 0.6 \text{ A}$ , $I_{12VMAX} = 1.0 \text{ A}$ )
Connectors	<p>AOUT0, AOUT1 (10-pin har-link® connector, P1) - 2x analog output</p> <p>AIN0 ... AIN3 (10-pin har-link® connector, P2) - 4x analog input</p> <p>AIN4 ... AIN7 (10-pin har-link® connector, P3) - 4x analog input</p> <p>DIO0 ... DIO7 (10-pin har-link® connector, P4) - 8x digital I/O</p> <p>DIO8 ... DIO15 (10-pin har-link® connector, P5) - 8x digital I/O</p> <p>DIO16 ... DIO23 (10-pin har-link® connector, P6) - 8x digital I/O</p> <p>SYNC0 ... SYNC3 (10-pin har-link® connector, P7) - 4x RS-485, trigger/sync</p> <p>AMC plug connector (170-pin AMC plug connector, J1) - AMC B/B+ compatible (MicroTCA™)</p> <p>Only for test- and programming purposes: X600 programming, debugging X1850 JTAG interface</p>
Temperature range	0...70 °C ambient temperature (free convection)
Humidity	max. 90%, non-condensing
Dimensions	mid-height, single-width (73.5 mm x 180 mm) AMC
Weight	140 g

**Table 1:** General data of the module

## 5.2 MicroTCATM /AMC Standards

μTCA	PICMG® MTCA.0 R1.0, PICMG® AMC.0 R2.0
IPMI	IPMI V1.5
Updates	PICMG® HPM.1 R1.0
PCIe	PCISIG® PCIe spec. R.1.0a, only lane 4 is used
Connector	AMC plug connector according to PICMG® 3.0 Rev. 3.0 AdvancedTCA® Base Specification and PICMG® AMC.1 R2.0 PCI Express on AdvancedMC™

**Table 2:** MicroTCA standards

## 5.3 Analog Outputs

Number of outputs	2 outputs, available as single-ended and differential (AOUT0, AOUT1)
Output voltage	$U_{OUT} = \pm 10\text{ V}$
Resolution	16 bit
Sampling rate	up to 600 KSPS
Output current	$I_{OUTMAX} = 10\text{ mA}$
Output state if board is switched off	high-resistance
I/O control	FPGA Spartan II
Protection circuits	transient protection $U_{BREAK} \approx \pm 26\text{ V}$
Connector	10-pin har-link female connector (P1)

**Table 3:** Data of analog outputs

## 5.4 Analog Inputs

Number of inputs	8 inputs, (AIN0 ... AIN7)
Input voltage	$U_{IN} = \pm 10 \text{ V}$ absolute max. input voltage with respect to Analog_GND: $U_{INMAX} = \pm 13 \text{ V}$
Resolution	16 bit
Sampling rate	up to 200 KSPS
Input filter	Low pass filter (Bessel filter 3 <sup>rd</sup> order), $f_G = 50 \text{ kHz}$
Input resistance	$R_{IN} = 100 \text{ k}\Omega$
I/O control	FPGA Spartan II
Protection circuits	transient protection $U_{BREAK} \approx \pm 26 \text{ V}$
Connector	10-pin har-link female connectors (P2, P3)

**Table 4:** Data of analog inputs

## 5.5 Digital Inputs/Outputs

Digital I/Os	24 digital I/Os (DIO0 ... DIO23), TTL-level, each channel separately programmable as <ul style="list-style-type: none"> <li>• input only</li> <li>• output sink</li> <li>• output source</li> <li>• output sink/source</li> </ul>
Maximum output current/channel	64 mA sink, 32 mA source
Input circuit	comparator with hysteresis, threshold '1' → '0' : $U_{OFF} = 0.8\text{ V}$ threshold '0' → '1' : $U_{ON} = 2.4\text{ V}$
Input resistance	Status = '0' : $R_{in0} \approx 4\text{ k}\Omega$ to GND Status = '1' : $R_{in1} \approx 4\text{ k}\Omega$ to +5V
I/O control	FPGA Spartan II
Input sampling rate	up to 1 MHz
Output update rate	up to 1 MHz
I/O protection	short circuit protection $U_{OUT} = f(I_{OUT})$ transient protection circuit activated at negative voltages below: $U_{LOW} < -0.7\text{ V}$ activated at positive voltages from: $U_{HIGH} > +7.0\text{ V}$
Connector	10-pin har-link female connectors (P4, P5, P6)

**Table 5:** Data of digital I/Os

## 5.6 Trigger Ports

Number	4x RS-485, trigger/sync (SYNC0 ... SYNC3)
Data rate	max. 1 MHz
Connector	10-pin har-link female connectors (P7)

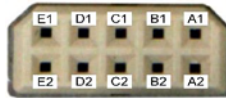
**Table 6:** Data of the trigger ports

## 6. Interfaces and Connector Assignments

### 6.1 Analog Out

Device connector: 10-pin har-link female connector, P1,  
 For the position of the connector in the front panel see page 10.

**Pin Position:**



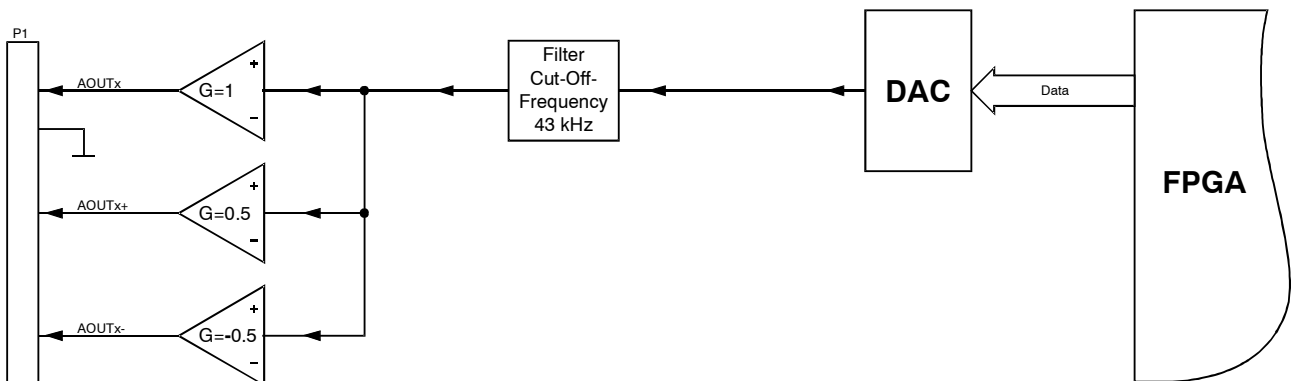
**Pin Assignment:**

AOUT0, AOUT1 (P1)					
<b>Pin</b>	<b>E1</b>	<b>D1</b>	<b>C1</b>	<b>B1</b>	<b>A1</b>
<b>Signal</b>	AOUT1+	AOUT0+	-	AOUT1	AOUT0
<b>Pin</b>	<b>E2</b>	<b>D2</b>	<b>C2</b>	<b>B2</b>	<b>A2</b>
<b>Signal</b>	AOUT1-	AOUT0-	ANALOG_GND	AOUTGND1	AOUTGND0

**Signal Description:**

AOUT<sub>x</sub>+, AOUT<sub>x</sub>-                      analog output signal lines of differential output circuit of channel x  
 (x = 0, 1)  
 AOUT<sub>x</sub>, AOUT<sub>x</sub>GND                  analog output signal lines of single ended output circuit of channel x  
 (x = 0, 1)  
 ANALOG\_GND                          reference potential of analog I/Os

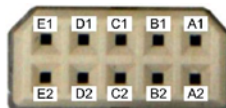
**Block Diagram Output Circuit:**



## 6.2 Analog In

Device connector: 10-pin har-link female connector, P2, P3  
 For the position of the connectors in the front panel see page 10.

### Pin Position:



### Pin Assignment:

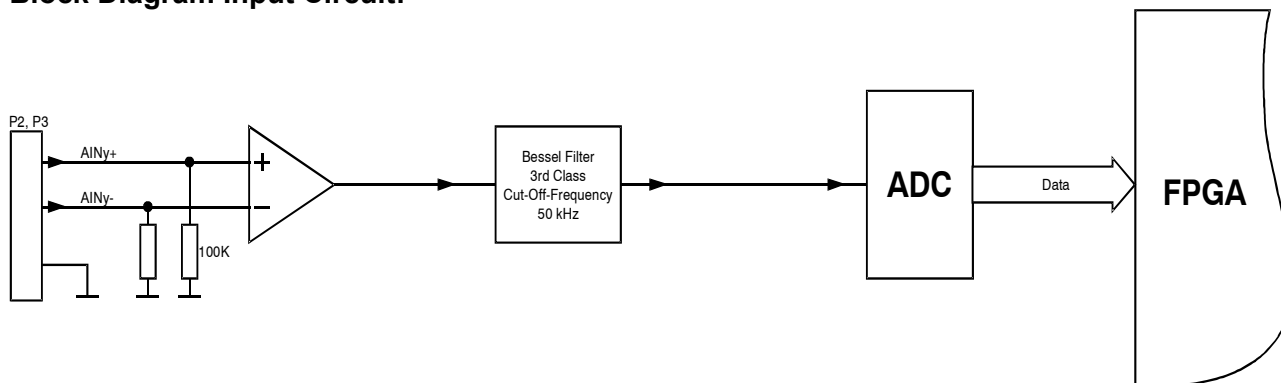
AIN0 ... AIN3 (P2)					
<b>Pin</b>	<b>E1</b>	<b>D1</b>	<b>C1</b>	<b>B1</b>	<b>A1</b>
<b>Signal</b>	AIN3+	AIN2+	-	AIN1+	AIN0+
<b>Pin</b>	<b>E2</b>	<b>D2</b>	<b>C2</b>	<b>B2</b>	<b>A2</b>
<b>Signal</b>	AIN3-	AIN2-	ANALOG_GND	AIN1-	AIN0-

AIN4 ... AIN7 (P3)					
<b>Pin</b>	<b>E1</b>	<b>D1</b>	<b>C1</b>	<b>B1</b>	<b>A1</b>
<b>Signal</b>	AIN7+	AIN6+	-	AIN5+	AIN4+
<b>Pin</b>	<b>E2</b>	<b>D2</b>	<b>C2</b>	<b>B2</b>	<b>A2</b>
<b>Signal</b>	AIN7-	AIN6-	ANALOG_GND	AIN5-	AIN4-

### Signal Description:

AIN+, AINx-                      analog input signal lines of channel x (x = 0 ...7)  
 ANALOG\_GND                    reference potential of analog I/Os

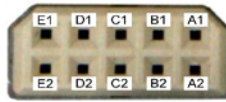
### Block Diagram Input Circuit:



### 6.3 Digital I/O

Device connector: 10-pin har-link female connector, P4, P5, P6  
 For the position of the connectors in the front panel see page 10.

**Pin Position:**



**Pin Assignment:**

DIO0 ... DIO7 (P4)					
Pin	E1	D1	C1	B1	A1
Signal	DIO6	DIO4	-	DIO2	DIO0
Pin	E2	D2	C2	B2	A2
Signal	DIO7	DIO5	GND	DIO3	DIO1

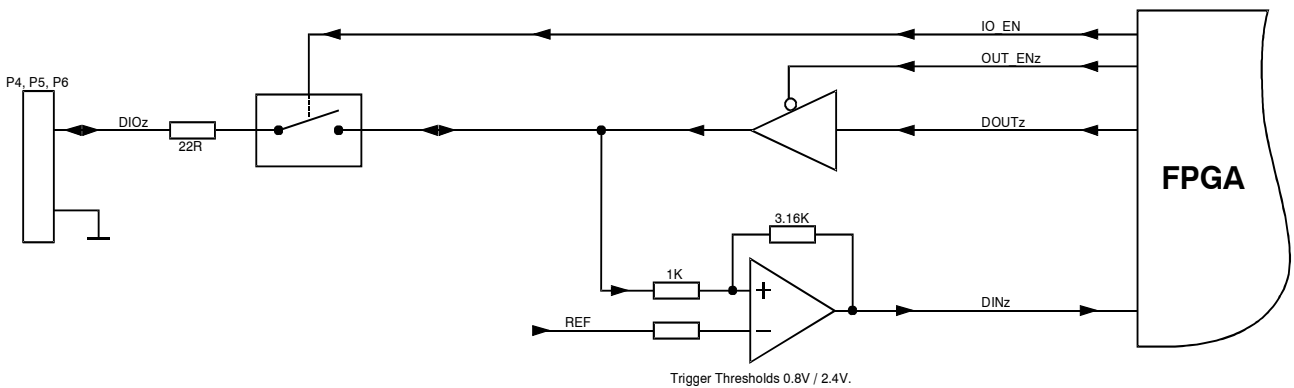
DIO8 ... DIO15 (P5)					
Pin	E1	D1	C1	B1	A1
Signal	DIO14	DIO12	-	DIO10	DIO8
Pin	E2	D2	C2	B2	A2
Signal	DIO15	DIO13	GND	DIO11	DIO9

DIO16 ... DIO23 (P6)					
Pin	E1	D1	C1	B1	A1
Signal	DIO22	DIO20	-	DIO18	DIO16
Pin	E2	D2	C2	B2	A2
Signal	DIO23	DIO21	GND	DIO19	DIO17

**Signal Description:**

DIOx digital input/output signal lines of channel x (x = 0 ...23)  
 GND reference potential of digital I/Os

**Block Diagram I/O-Circuit:**



## 6.4 Trigger Ports

Device connector: 10-pin har-link female connector, P7  
 For the position of the connector in the front panel see page 10.

### Pin Position:



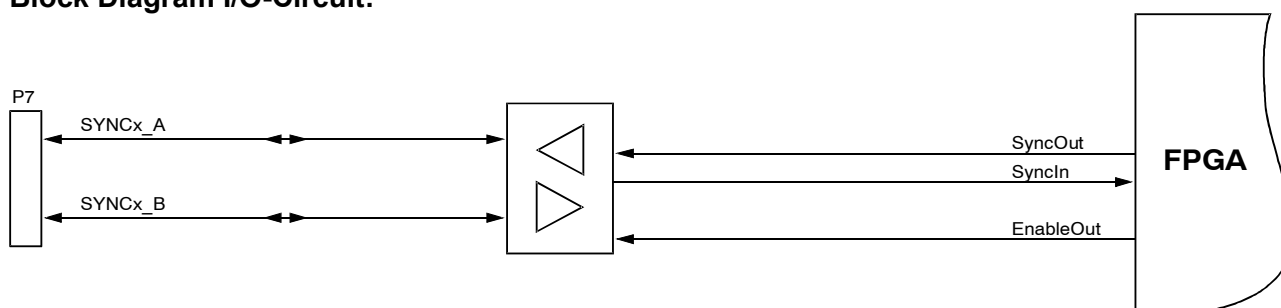
### Pin Assignment:

SYNC0 ... SYNC3 (P7)					
Pin	E1	D1	C1	B1	A1
Signal	SYNC3_A	SYNC2_A	-	SYNC1_A	SYNC0_A
Pin	E2	D2	C2	B2	A2
Signal	SYNC3_B	SYNC2_B	GND	SYNC1_B	SYNC0_B

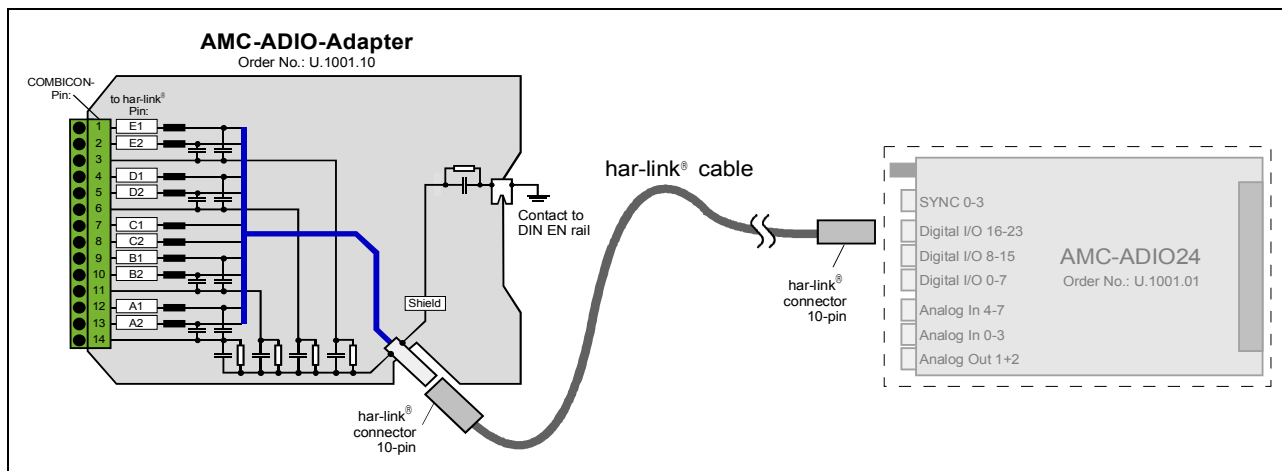
### Signal Description:

SYNC<sub>x</sub>\_A,  
 SYNC<sub>x</sub>\_B     signal lines of the RS-485 ports (trigger/sync) of channel x (x = 0 ... 3)  
 GND            reference potential of trigger/sync I/Os

### Block Diagram I/O-Circuit:



## 7. AMC-ADIO-Adapter (Order no. U1001.10)



**Figure 5:** AMC-ADIO- Adapter

The AMC-ADIO-Adapter is designed to support the wiring of the AMC-ADIO24. The adapter can be connected to each I/O-type of the AMC-ADIO24: digital I/O, analog inputs, analog outputs or SYNC-ports.

Eight signal lines plus Ground and Shield are routed via the har-link® cable to the Mini COMBICON connector at the AMC-ADIO-Adapter. One Shield clamp is available for a signal pair. This ensures optimum shielding of the cable assembly.

The shield potential is automatically connected with the DIN-EN rail via a spring cage contact when the AMC-ADIO-Adapter is mounted on the rail.

If connected to one of the analog input connectors at the AMC-ADIO24, four differential analog I/O signals plus Shield and Ground are available. Connected to the analog output connector two analog outputs each as differential and single ended plus Shield are available.

If connected to one of the digital I/O connectors eight digital I/O-signals plus Shield and Ground are available. Connected to the SYNC-connector 4 SYNC ports plus Shield are available.

HF ferrite in the signal lines improve the EMC properties of the AMC-ADIO-Adapter.

The Mini COMBICON connector is pluggable to allow easy removing of the cable assemblies.

## 7.1 AMC-ADIO-Adapter View



**Figure 6:** AMC-ADIO-Adapter view with COMBICON connector

## 7.2 Technical Data AMC-ADIO-Adapter

Connectors	AMC side: (10-pin har-link® connector, female) - routed signals Process side: (14-pin COMBICON style connector with spring-cage connection)
Temperature range	-40 °C ... +85 °C ambient temperature
Humidity	max. 90%, non-condensing
Dimensions	22.5 mm x 112 mm x 113 mm
IP Class	IP20
UL-Flame Rating	UL94 V-0
Weight	tbd.
Connectors	AMC side: 10-pin har-link® female connector Process side: 14-pin COMBICON style connector FK-MCP 1,5/14-STF-3,81

**Table 7:** General data of the adapter

### 7.3 Connector Assignment

Eight signal lines plus ground and shield are routed via the har-link® cable to the Mini COMBICON connector at the AMC-ADIO-Adapter.

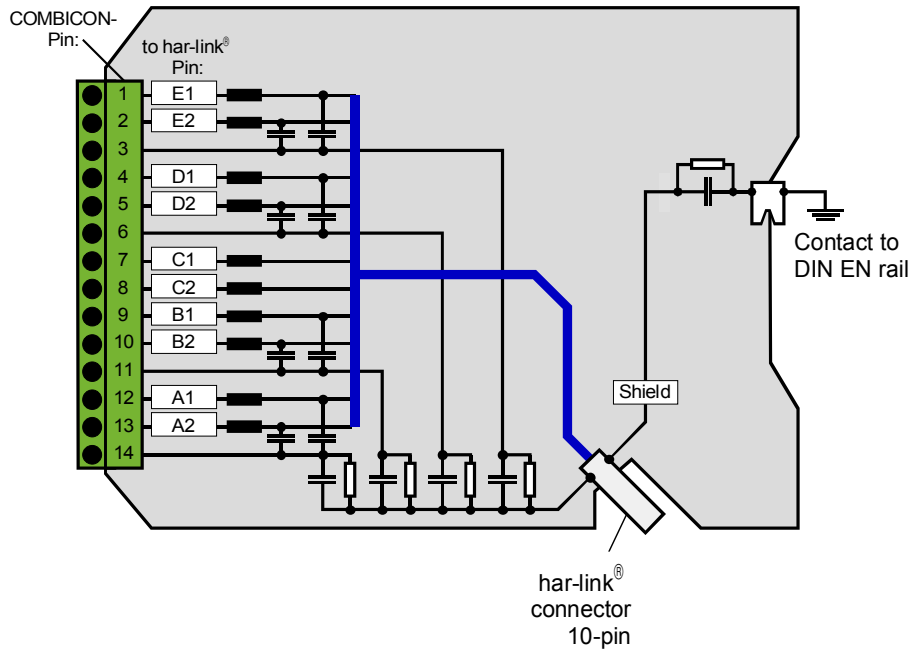


Figure 7: Signal assignment of the 14-pin COMBICON connector

Pin	Signal routed	Signal
1	E1	XSIG0+
2	E2	XSIG0-
3	Shield_E	Shield
4	D1	XSIG1+
5	D2	XSIG1-
6	Shield_D	Shield
7	C1	Reserved
8	C2	XAGND0
9	B1	XSIG2+
10	B2	XSIG2-
11	Shield_B	Shield
12	A1	XSIG3+
13	A2	XSIG3-
14	Shield_A	Shield

#### Signal Description

XSIGy+, XSIGy- Signal lines of the connected interface routed via the har-link® cable to the Mini-COMBICON connector at the AMC-ADIO-Adapter, channel y (y= 0, 1, 2, 3) The assignment of the pins depends on the assignment of the AMC-ADIO24 har-link connector connected (see page 82).

Pin	Signal routed	Pin assignment depending on the AMC-ADIO24 connector connected to the AMC-ADIO-Adapter						
		P1 Analog Out	P2 Analog In	P3 Analog In	P4 Digital IO	P5 Digital IO	P6 Digital IO	P7 Trigger Ports
1	E1	AOUT1+	AIN3+	AIN7+	DIO6	DIO14	DIO22	SYNC3_A
2	E2	AOUT1-	AIN3-	AIN7-	DIO7	DIO15	DIO23	SYNC3_B
3	Shield	Shield	Shield	Shield	Shield	Shield	Shield	Shield
4	D1	AOUT0+	AIN2+	AIN6+	DIO4	DIO12	DIO20	SYNC2_A
5	D2	AOUT0-	AIN2-	AIN6-	DIO5	DIO13	DIO21	SYNC2_B
6	Shield	Shield	Shield	Shield	Shield	Shield	Shield	Shield
7	C1	-	-	-	-	-	-	-
8	C2	ANALOG_GND	ANALOG_GND	ANALOG_GND	GND	GND	GND	GND
9	B1	AOUT1	AIN1+	AIN5+	DIO2	DIO10	DIO18	SYNC1_A
10	B2	AOUT_GND1	AIN1-	AIN5-	DIO3	DIO11	DIO19	SYNC1_B
11	Shield	Shield	Shield	Shield	Shield	Shield	Shield	Shield
12	A1	AOUT0	AIN0+	AIN4+	DIO0	DIO8	DIO16	SYNC0_A
13	A2	AOUT_GND0	AIN0-	AIN4-	DIO1	DIO9	DIO17	SYNC0_B
14	Shield	Shield	Shield	Shield	Shield	Shield	Shield	Shield

**Table 8:** Pin assignment of the COMBICON connector

## 8. Order Information

Type	Properties	Order No.
AMC-ADIO24	8x analog input, 2x analog output, 24x digital I/O, 4x trigger	U.1001.01
<b>Accessories:</b>		
AMC-ADIO-Adapter	DIN EN rail module for conversion from 10-pin har-link® interface female connector to a 14-pin MC 1,5 COMBICON female connector	U.1001.10
har-link® cable	adapter cable, 10-pin har-link® male connector to 10-pin har-link® male connector,	U.1001.11
	length 0.5 m	U.1001.12
	length 1.0 m	U.1001.13
	length 2.0 m	
<b>Manuals:</b>		
AMC-ADIO24-ME	Manual in English	U.1001.21
AMC-ADIO24-ENG	Engineering manual in English <sup>1)</sup> Content: Circuit diagrams, PCB top overlay drawing, data sheets of significant components	U.1001.25

<sup>1)</sup> This manual is liable for costs, please contact our support.

**Table 9:** Order information

## 9. Abbreviations

---

0x1234	hexadecimal value 1234
acc.	access
ADC	analog/digital converter
addr.	address
attr.	attribute
bwl	byte + word + long access possible
DAC	digital/analog converter
DDFS	direct digital frequency synthesizer
DMA	direct memory access
DREQ	DMA request
freq.	frequency
Hex	hexadecimal
I/O	input/output
in	input
Irq	interrupt
KSPS	kilo samples per second
l	long
MSPS	mega samples per second
n/a	not applicable
out	output
R	read access
RA	read address
reg. no.	register number
RO	read only access
SPI	serial peripheral interface (bus)
sync	synchronisation
TRP	timing routing pool (see page 16)
TS	timestamp
W	write access
WA	write address
wl	word + long access possible
WO	write only access

---