The Embedded I/O Company

TEWS TECHNOLOGIES

TAMC900

AMC with 8 high Speed ADCs 105MSps, 14Bit

Version 2.0

User Manual

Issue 2.0.1

April 2010

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

TAMC900-10R

AMC with 8 high speed ADCs 105 MSps 14 bit, RoHS compliant. Requires Signal Conditioning Adapter

This document contains information, which is proprietary to TEWS TECHNOLOGIES GmbH. Any reproduction without written permission is forbidden.

TEWS TECHNOLOGIES GmbH has made any effort to ensure that this manual is accurate and complete. However TEWS TECHNOLOGIES GmbH reserves the right to change the product described in this document at any time without notice.

TEWS TECHNOLOGIES GmbH is not liable for any damage arising out of the application or use of the device described herein.

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

©2008 – 2010 by TEWS TECHNOLOGIES GmbH

All trademarks mentioned are property of their respective owners.

Issue	Description	Date
1.0	Initial Issue	April 2008
1.1	Minor changes in chapter 9 Added details to chapter 4.3	May 2008
1.2	Corrections in “Crosspoint Switch Control Register “ Different minor corrections and additions	August 2008
1.0.3	New User Manual Issue Notation Correction of “MMC JTAG Connector Pin Assignment”	January 2009
2.0.0	Update to TAMC900 V2.0 Added FPGA code description	November 2009
2.0.1	Firmware Upgrade to Version 2	April 2010

Table of Contents

List of Tables

List of Figures

1 Product Description

The TAMC900 is a high speed, high performance analog to digital converter AdvancedMC. In addition to the eight high speed ADCs, it provides excessive preprocessing power by a Virtex-5 FPGA and high speed on board memory for e.g. full bandwidth snapshots.

The up to x8 PCIe link of the TAMC900 is used to transmit the ADC data to the CPU.

To adapt the TAMC900 to different customer requirements, the TAMC900 is equipped with a Signal Conditioning Adapter (SiCA) which holds the connector for the analog inputs, the connectors for the clock and trigger inputs, and the analog signal conditioning.

The TAMC900 provides three clock inputs and three trigger inputs. The three external clock inputs and the PCIe reference clock are routed to a flexible clocking scheme that allows independent clocking of the ADCs in two groups. The trigger inputs are routed to the FPGA.

Eight LTC2254 ADCs provide up to 105 MSps and 14 bit resolution each. The minimum sample rate is 1 Msps.

4 MByte high speed on board SRAM enables snapshots of all ADCs at full speed and full resolution for 2ms.

According to AMC.0, the TAMC900 provides an IPMI compliant Module Management Controller (MMC) with temperature monitoring and hot-swap support.

For First-Time-Buyers the engineering documentation TAMC900-ED is recommended. The engineering documentation includes TAMC900-DOC, schematics and data sheets of TAMC900.

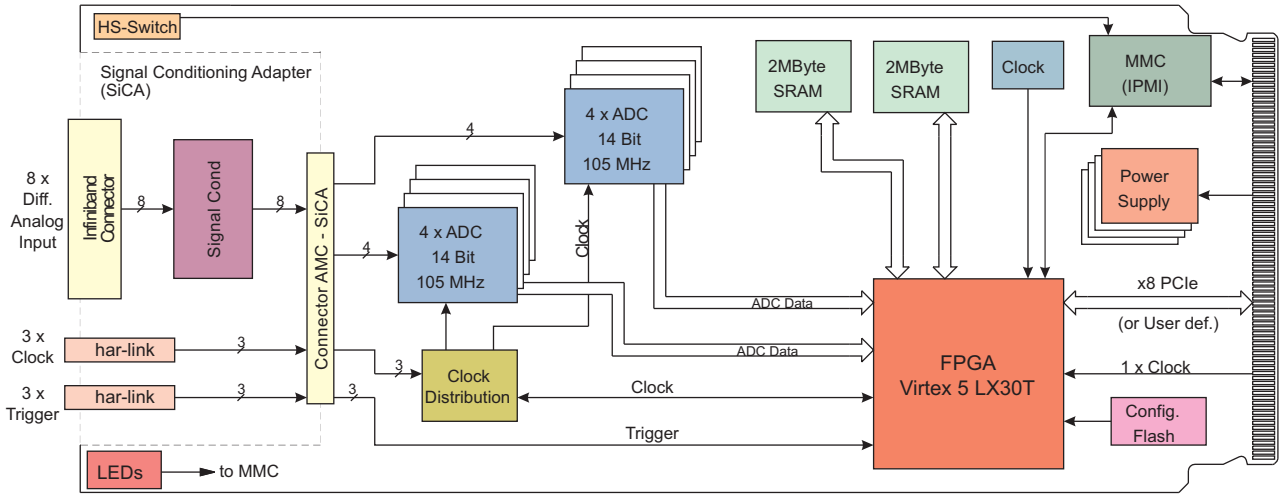

Figure 1-1 : Block Diagram TAMC900 with Signal Conditioning Adapter

2 Technical Specification

AMC Interface		
Mechanical Interface	Advanced Mezzanine Card (AMC) confirming to PICMG® AMC.0 R2.0, Single, Mid Size Module	
Electrical Interface	Virtex-5 FPGA connected to AMC Port 4-11 PICMG® AMC.1 R1.0 Type 4 (x4 PCI Express), or other interfaces defined by customer	
IPMI		
IPMI Version	1.5	
Front Panel LEDs	Blue Hot Swap LED Red Fail LED (LED1) Green User LED (LED2)	
On Board Devices		
Target Chip	Xilinx Virtex-5 with integrated PCI-Express Endpoint Block and two integrated Gigabit Ethernet MACs	
RAM	2 x 2 MByte QDR-II SRAM	
ADC	8 x LTC2254 (105 MSps, 14 bit)	
I/O Interface		
Number of Analog Channels	8 differential	
Analog Input Gain	depends on Signal Conditioning Adapter used	
Analog Input Voltage Range	depends on Signal Conditioning Adapter ADC analog input Voltage (differential) = $2V_{P-P}$ ADC input common mode Voltage = 1V to 1.9V	
ADC INL/DNL Error	±1 / ±0.5 LSB (typical)	
Number of Clock Inputs	3 differential (LVDS)	
Number of Trigger Inputs	3 differential (LVDS)	
I/O Connector	120 pin Connector to Signal conditioning adapter that holds the I/O Connectors for the analog inputs, clocks and trigger signals	
Physical Data		
Power Requirements	2A typical, 4A max. @ +12V DC (Payload Power) 50 mA typical @ +3.3V DC (Management Power) **The exact Power requirement of the TAMC900 depends on Signal Cond. Adapter used and FPGA utilization.**	
Temperature Range	Operating Storage	0 °C to +55 °C 0 °C to +70 °C
MTBF	391000 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.	
Humidity	5 – 95 % non-condensing	
Weight	100 g	

Table 2-1 : Technical Specification

3 Handling and Operating Instructions

Do not exceed the maximum input voltages of the TAMC900 I/Os.
The TAMC900 will be damaged if higher voltage levels are applied.

3.1 ESD Protection

The TAMC900 is sensitive to static electricity.
Packing, unpacking and all other handling of the TAMC900 has to be done in an ESD/EOS protected area.

3.2 Thermal Considerations

The TAMC900 requires forced air cooling during operation.
Without forced air cooling, damage to the device will occur.

3.3 AMC Module Insertion & Hot-Swap

3.3.1 Insertion

Handle	Blue LED	Description
Open (Full extracted)	OFF	Insert Module into slot
Open (Full extracted)	ON	Module is ready to attempt activation
Closed (Pushed all way in)	Long Blink	Hot-Swap Negotiation
Closed (Pushed all way in)	OFF	Module is ready & powered

Table 3-1 : AMC Module Insertion

When the blue LED does not go off but returns to the “ON” state, the module FRU information is invalid or the carrier cannot provide the necessary power.

3.3.2 Extraction

Handle	Blue LED	Description
Pulled out 1/2	OFF	Request Hot-Swap
Pulled out 1/2	Short Blink	Hot-Swap Negotiation
Pulled out 1/2	ON	Module is ready to be extracted
Open (Full extracted)	ON	Extract Module from slot

Table 3-2 : AMC Module Extraction

4 Functional Procedures

The following diagram illustrates the structure of the ADC data acquisition system implemented on the TAMC900. The structure is abstract in order to show the different functional units and the data flow. All physical connections are termed in a bold style. The remaining ones are internal connections.

The functional units are described in the subsequent chapters.

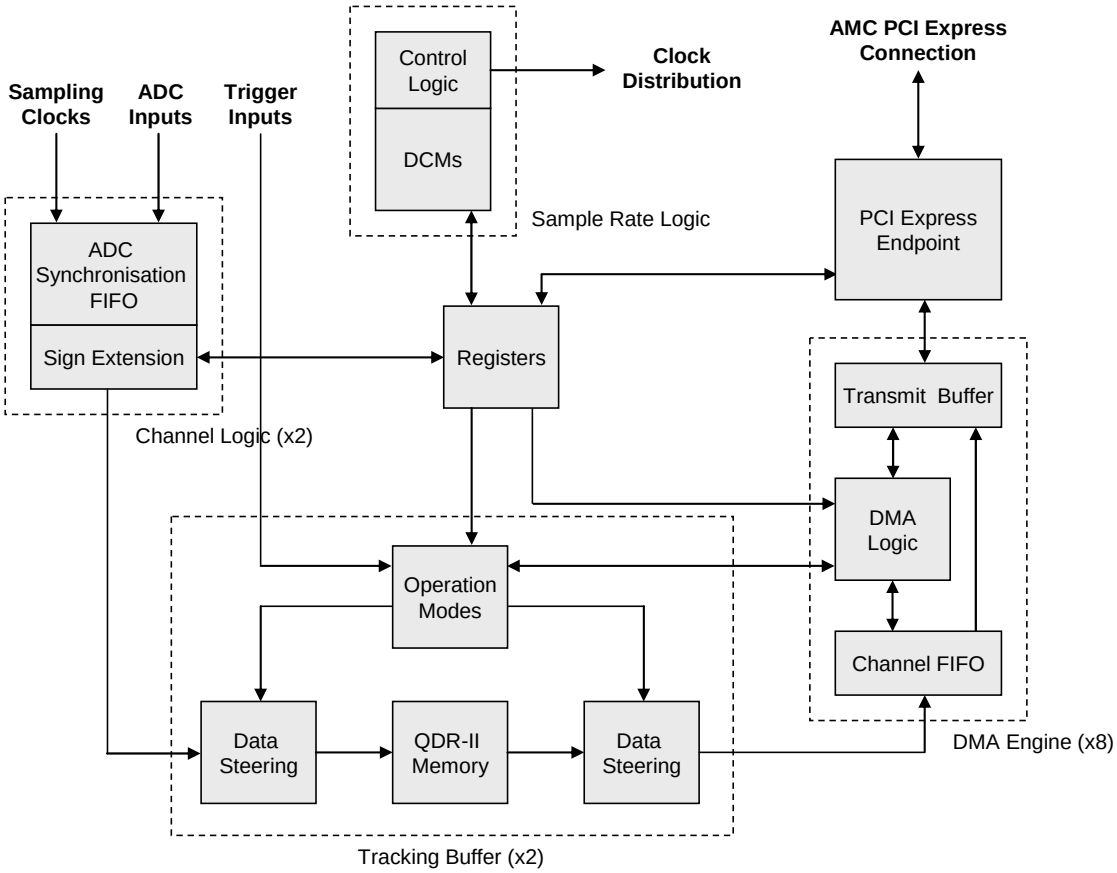

Figure 4-1: TAMC900 FPGA System Structure

The physical clocking structure of the TAMC900 results in two different (ADC) channel groups. The module contains eight separate 14 bit ADC channels that are structured in the way: two times four. The ADC-to-channel group assignment is shown in the subsequent table.

Channel Group	Corresponding ADC(s)
0	0-3
1	4-7

Table 4-1: ADC Channel Group Assignment

Each group has restrictions concerning a common sampling clock and a common trigger configuration.

4.1 Channel Logic

The channel logic is implemented for every channel group. It realizes the data synchronization between the internal processing clock and the ADC sample clock domain. This is necessary because these clocks can have an arbitrary ratio to each other. Only the limits of the ADCs have to be considered.

The ADCs have an operating frequency of 1 MHz up to 105 MHz.

Besides this the sign extension is performed in this unit. This has been implemented to simplify the data processing out of the target memory by mapping the ADC values into legal data types. The sign-extension method is described below.

- In 2-th complement the ADC sign bit (13) is mapped onto the additional bits 14 and 15.
- In binary offset format the ADC sign bit (13) is mapped onto bit 15 and the bits 13 and 14 are set to zero.

A_{IN}+ - A_{IN}- (2V Range)	OF	Data (Offset Bin.)	Data (2's Compl.)
>+1.000000V	1	0x9FFF	0x1FFF
+0.999878V	0	0x9FFF	0x1FFF
+0.999756V	0	0x9FFE	0x1FFE
...	...	...	...
+0.000122V	0	0x8001	0x0001
0.000000V	0	0x8000	0x0000
-0.000122V	0	0x1FFF	0xFFFF
-0.000244V	0	0x1FFE	0xFFFE
...	...	...	...
-0.999878V	0	0x0001	0xE001
-1.000000V	0	0x0000	0xE000
<-1.000000V	1	0x0000	0xE000

Table 4-2 : ADC Data Format with Sign Extension

4.2 Tracking Buffer

The tracking buffer is the central unit of the module. It is used to realize the different operation modes by using the on-board QDR-II memories. The modes are described in the following chapters.

Note that an operation mode is set for entire channel group and cannot be done for every channel in a different way.

The use of the QDR-II memory in the data path has another reason. In case of PCI Express transmission gaps the memory can buffer the data so that the data integrity is still held. Based on the internal organization structure there are 256k words for a single channel.

At a sample frequency of 105 MHz a period of approximate *2.4 m sec* can be buffered before the loss of data occurs.

The restriction on channel groups has impacts on the buffer technique. The throttle mode is shared by the concatenated channels. It is used to stop providing new data to the DMA Engines if one engine signalizes that its dedicated channel FIFO is running full.

One stalled channel may cause that all channels of the group get into error state due to buffer overflow.

4.2.1 Operation Modes (OM)

One operation mode is provided by the firmware (compare chapter “Trigger Configuration 0/1”) which allows selecting which acquired ADC data should be transmitted via DMA. The different settings are explained afterwards.

There is no absolute sample count that defines the end of transmission. This is performed until a descriptor signalizes the end of the link list.

Due to the initially mentioned restriction the channels of a group are started together and a channel of the group can be restarted first if all other channels of the group are in idle.

The *Post Trigger Data Gathering* is the standard case. The ADC data obtained after a trigger impulse is used for data transmission. Since the mode does not make use of previous data samples there are no restrictions concerning the number of transmitted samples. The content of register “Channel Pre-Trigger Data Size” must be zero (see below).

The *Pre Trigger Data Gathering* is the opposite case. If a trigger event has been detected, the data that has been monitored before its occurrence will be transmitted. For this, the content of the QDR-II memory is used. Consequently, the number of samples cannot exceed the QDR-II memory depth. The value of register “Channel Pre-Trigger Data Size” defines the transmission size. Moreover the linked list must match the requested sample count (see below).

The *Around Trigger Data Gathering* is a mixture of the previous two modes. It is used to obtain the data before and after the trigger event. Thus it makes also use of the QDR-II memory. Hence it is subdued the size limitation as the Pre-Trigger. The implementation is done in accordance to the above two methods. The value of register “Channel Pre-Trigger Data Size” defines the demanded tracked samples as starting point. Afterwards transmission is performed until the linked list has been finished by a descriptor.

Pre- and Around Trigger Data Gathering require some time between trigger events to monitor ADC channel data. This time depends on the adjusted sample rate. If the inter trigger event period is too short, it is possible that the transmitted sample data is invalid.

There are three external (physical) trigger sources that can be used. These are described in chapter “Trigger Configuration 0/1”. For adaptation purposes the interpretation of a trigger input signal change (edge) can be defined.

Currently only edge-evaluated trigger have been implemented. Level trigger are not considered.

Besides the external trigger inputs a software trigger input can be selected. This make use of an additionally register (see chapter “Global Reset and Software Trigger Input”) to initiate processing by writing at a certain bit position.

All trigger inputs are handled in the same manner. There is no difference.

Additionally the Post Trigger Data gathering can be used in conjunction with the software trigger to realize a free-running operation mode if necessary.

4.3 DMA Engine

Generating PCI Express Transaction-Layer-Packets (*TLPs*) from the channel data and transmitting these via the PCI Express Interface is the purpose of a DMA Engine. Every engine has an internal buffer structure build of a first-level FIFO and a second-level transmit buffer. The FIFO is used for collecting the channel specific data samples besides the QDR-II FIFO. The buffer is storage for the PCI Express TLPs used during burst writes and in case of a packet retry. The TLPs have a static defined packet size (see below).

Due to flexibility considerations a concept of DMA descriptors has been implemented. These descriptors built a *Linked List* (LL) which allows defining regions (windows) in host memory for placing channel ADC data. Moreover the LL contains additional processing information respectively settings (see chapter “DMA Descriptors”).

After channel activation or channel reset a DMA Engine uses its DMA (base) descriptor. This is set in the channel specific DMA (Base) Descriptor Address register.

If a channel is enabled, writing into the channels DMA (Base) Descriptor register will initiate the reload of the descriptor.

Starting at the obtained base address data packets are transmitted consecutively. The DMA memory is filled in blocks. If the defined length (sample count) of a window has been achieved, the subsequent descriptor will be loaded. Hence it will be switched to the next list element respectively memory address range. The data acquisition and its transmission are performed until the last descriptor of the chain has been processed.

To reduce PCI Express traffic, interrupts will only be generated if additional steering flags demand this.

After the last descriptor has been processed, its successor list element is loaded. Thus the last element must point to the first if the list should be used again.

If a DMA channel is stopped before its linked list could have been processed completely, it is not clear up to which address the data inside the window is valid. Such a case can be handled by controlling the fill-level counter. This monitors the current position inside a window. The information can be obtained through the DMA Buffer Fill-Level register. There is one dedicated register for every channel.

The channels are processed in a rotating mechanism that considers possible interrupted PCI Express transactions. The channels are selected for transmission in a consecutive manner, starting with channel zero. After the last channel has been processed, the first is taken again. If the current selected channel has no packet prepared e.g. since it is disabled or is offline caused by a transmission error, the subsequent channel is selected.

The engine does not make any restriction to the DMA window size. Odd and even sizes are legal. If a window has a remaining size less than the static packet size (32 DWORDS), a packet is generated that fills the remaining gap of the window.

Interrupt events have to be acknowledged in the corresponding Channel DMA Status register after occurrence. For additional information about interrupts refer chapter “Interrupts”.

4.4 DMA Descriptors

The DMA descriptors are used to define target memory regions (windows) that are used to store the sampled data of a channel inside the host memory. For this a descriptor contains on the one hand flags required for its processing and on the other hand information to build a *Linked List* (LL).

A DMA descriptor consists of three instructions words where every word comprises 32 bit. These have to be arranged in a consecutive manner in the module's dedicated memory (see chapter "DMA Descriptor Space"). Placing the descriptors into embedded Block RAM (inside the FPGA) will reduce the PCI Express traffic and hence accelerate the internal processing. Moreover it simplifies the access onto the descriptor instruction words.

An embedded Block RAM in Virtex-5 is 36 kb large and thus can contain up to 42 logical DMA regions for one physical ADC if all eight channels are used parallel. One single channel alone can have up to 341 different regions. The firmware occupies two memory blocks thus having twice the mentioned depth. The number of address bit is defined in accordance to that.

The structure of this block is shown below.

Bits							
31:28	27:24	23	22	21	20	19:11	10:0
0xC	Reserved	LF	Reserved	HI	CBA	Reserved	Subsequent Linked List Pointer (resident in on-board block RAM)
Pointer (Address) to DMA Memory Region inside the Host Memory							
Length of the DMA Memory Region inside the Host Memory in Samples (16 bit words)							

Table 4-3 : DMA Descriptor Structure

Providing steering information for every descriptor separately allows process-steering during runtime and offers a different handling of the descriptors. Notice that changing the descriptors after channel activation is not recommended. The flags are described afterwards.

The identifier 0xC marks the first instruction word of a set. It is used to detect if a descriptor is provided at the currently accessed memory address that can be used by a DMA Engine.

If the identifier is not set, the internal structure will not use the descriptor.

The Last identifier Flag *(LF)* marks the last descriptor of the linked list. This information is used to execute post-DMA-sequence tasks e.g. stop data acquisition and transmission.

The information that a certain descriptor has been processed can be sent via an interrupt to the host respective a software driver during processing by setting the Host Interrupt (HI) flag.

Processing all linked list DMA buffers in a cyclic manner, if required, is possible by setting the Continue at Base Address (CBA) descriptor flag. If the bit is set, the DMA (base) descriptor start address (compare chapter "Channel DMA (Base) Descriptor Addresses 0-7" is taken next regardless of a defined subsequent descriptor in the current descriptor.

The Subsequent Linked List Pointer is an address inside the module's descriptor address range (see above). Due to the 36 kbit Block RAM size the address length is 10 bit.

Setting the Linked List Pointer to 0x0 selects the first DMA descriptor memory address.

Besides this steering information the descriptor stores the information about the pointer (address) to the DMA memory region inside the host memory and the information about the length of this memory region.

Be aware that the length is defined in samples. Hence the number of required bytes is twice as large as the defined length.

4.5 Sample Rate Logic

The sample rate logic allows defining two different clocks that can be used as sample rate for the different ADC channel groups. There are two Digital Clock Managers (DCMs) that generate a clock based on a multiply-divide ratio. The internal control logic employs clock multiplexers for switching the clocks to a certain group. The multiplexers are glitch-free so that switching can be performed at every time.

Switching during runtime (processing) is not recommend.

In accordance with that, both groups can be set into a common clock mode (sourced from DCM0 or DCM1) or an independent clock mode.

4.6 Module Behavior

4.6.1 Power-Up/Reset

The module has reset-conditions after it has been activated. For this the PCI Express Endpoint's dedicated reset signal is used. This means

- all registers have their reset value
- DCMs have reset settings
- trigger signals are reset
- the QDR memories invalidate their data
- all DMA Engines lose their information
- DMA processing select the base Descriptors as defined through the register map
- interrupts are de-asserted
- CPLD has been reset

4.6.2 Pre-Initialization (Setup) Check

Before any operation on the TAMC900 can be performed it has to be checked that

- the GSTAT bit inside the Module Status and DCM 0/1 Status register indicates that the module is operational,
- all (unused) channels are disabled.

4.6.3 Channel Setup

The recommend steps to setup a channel respectively a channel group is described afterwards.

- Disable all channels in the Channel Group before manipulation
- Configure ADCs sample rate
    - DCM first if necessary (to have the correct clock at the ADCs' input)
    - Sample Clock Configuration afterwards
- Define corresponding Channel Group Trigger Mode
    - Pre-Trigger Data Size if necessary
- Provide DMA descriptors and assign a base address to the corresponding channel register
- Set Channel Configuration register
- Enable the channel in Global Channel Configuration register (this will start processing)

4.7 Channel Logic

4.7.1 Channel Reset

Beside the general reset, that has an effect on the complete module, every channel can be reset separately. Such a reset causes that

- a channel trigger event (if one has occurred) is reset
- channel DMA information (e.g. remaining window size) are obsolete
- the DMA base descriptor as defined through the register map is loaded
- the current processed DMA transmission is aborted
- the channel DMA status register is cleared

If a channel is reset while others are running, this channel cannot be started until the others of the corresponding channel group have finished.

4.7.2 Channel Activation

The activation is based on the Global Channel Configuration register and the Channel Configuration register. Both have to be configured appropriately.

The significant settings in the Channel Configuration register are the Channel Enable Bit (*CHEN*) and the Channel DMA Enable Bit (*DMAEN*).

Bit *CHEN* is used to activate (physically) the concatenated ADC. This refers to the LTC2254 Output Enable and the Shutdown steering input. Valid ADC samples can only be obtained if the sample frequency is set correctly and this bit is set. Stopping a channel should not be done with this steering bit since the DMA operation will not stop.

Bit *DMAEN* steers the data transport into the Tracking Buffer and to the associated DMA Engine. Moreover it controls the DMA transmission. Accordingly, if the bit is not set, no DMA transmission can be started respectively performed. This means that e.g. in the case that the DMA Engine has been started and the DMA enable bit is reset during operation the processing

- of the current packet is performed but no further packets are generated,
- all dynamic DMA information is held.

Stopping DMA transmission should only be done by using the DMAEN bit since all information especially the one up to with memory address valid data have been placed is visible.

The master enable is placed in the Global Channel Configuration register. If the channel is not activated there, the channel is held in reset. This allows configuring the channel completely before activation. Thus all changes occur concurrently and not consecutively.

The master activation of a channel causes (assuming that the channel has been configured) that

- ADC data can be read through the register interface (*)
- the channel's DMA (Base) Descriptor Addresses register content is used to load DMA descriptor information
- DMA information (e.g. Channel DMA Buffer Fill-Level) is set in the register map
- in dependence of the trigger mode, the QDR-II memory starts monitoring the ADC data

The asterisk marked aspect is already available after the Enable bit in the Channel Configuration register (see below) has been set.

4.7.3 Descriptor Change

As afore-mentioned the DMA operation makes use of DMA descriptors. There is one base address Descriptor address per channel which is used as entry point to the linked list.

Several processing flags can be set for a descriptor *before* the channel has been activated. Changes *after* activation may be destructive. Hence, only if a channel is *not* operating, a new base address can be set. This causes a reload of the selected address.

A change of an already loaded descriptor e.g. the DMA window length will not be detected. In such a case the base address must be assigned again or the channels need to be re-enabled.

Manipulating the content of a DMA (Base) Descriptor Addresses register while the accompany DMA Engine is active is not allowed. Moreover changing a Descriptor in memory during its processing is also not allowed.

4.7.4 Channel Start/Stop

There are some constraints on the configurability of a channel in context of its channel group initially mentioned in this chapter. Additionally, the channel start conditions for data processing are summarized afterwards.

The DMA operation can be performed if

- DMA operation is enabled
- a valid DMA descriptor is defined and loaded
- the associated QDR-II memory is ready for use
- the Channel Group Trigger is configured
- the channel itself (channel configuration) is enabled
- the channel's global control bits are set

If one condition is not met, the corresponding channel cannot be used.

Besides these conditions it has to be considered that the QDR-II memories have been filled with valid data as far as required for Pre-Trigger Data Gathering if used.

An active channel is stopped automatically if its associated Linked List has been processed. Stopping the processing while it is running (if necessary) should be done via the DMA Enable Bit (see chapter "Channel Activation").

Since the channels are handled concurrently the length of the channel DMA windows is defined separately. Consequently, one channel can still be active while the others the group have already finished.

4.8 Restrictions

4.8.1 Processing Limit

The PCI Express adaptation is currently limited to 1 GByte/sec. Due to header overhead a fraction of approximately 88.89% (888 MByte/sec) can be *continuously* transmitted via the module. Exceeding this limitation causes the DMA processing to stop after the internal buffer structure has collapsed.

For all channels at least 256 k samples can be obtained independently of the transmission rate caused by the integration of the QDR-II memory in the transmission path (refer chapter "Tracking Buffer").

The amount of data can be composed over all channels.

5 Address Map

5.1 PCI Express Configuration

The TAMC900 module will be present in the PCI Device Tree with the subsequent information.

PCI Information	Hex Value	Description
Vendor	0x1498	TEWS TECHNOLOGIES GmbH
Device ID	0x8384	TAMC900
Class Code	0x118000	Signal Processing Controller

Table 5-1 : TAMC900 PCI Device Information

The information about the local (on board) addressable data regions are summarized in the subsequent table.

Local Space	PCI Base Address (Offset in PCI Configuration Space)	PCI Space Mapping	Size (Byte)	Port Width (Bit)	Endian Mode	Description
0	0 (0x10)	MEM	1024	32	Little	Register Space
1	1 (0x14)	MEM	8192	32	Little	DMA Descriptor Space

Table 5-2 : TAMC900 Local Space Configuration

Accessing the module spaces have to be performed in single 32 bit transactions.

Error handling has been implemented in accordance with PCI Express Specification to ensure system stability. This means on the one hand unsupported TLPs (compare Table below “TLP Type Summary”) are dumped so that these may not affect the internal logic. On the other hand an error message (TLP) will be generated and transmitted to Root Complex.

TLP Type	Handling
Memory Read, 32 Bit Addressing	Permitted
Memory Write, 32 Bit Addressing	Permitted
Memory Read, 64 Bit Addressing	Prohibit, Error Message
Memory Write, 64 Bit Addressing	Prohibit, Error Message
I/O Read	Prohibit, Error Message
I/O Write	Prohibit, Error Message
Read Locked, 32 Bit Addressing	Prohibit, Error Message
Read Locked, 64 Bit Addressing	Prohibit, Error Message
Configuration Read, 0/1	Permitted
Configuration Write, 0/1	Permitted
Messages	Dumped
Completion	Prohibit, Error Message
Completion, Locked	Prohibit, Error Message

Table 5-3 : TLP Type Summary

5.2 Register Space

The register space has been defined to be 32 bit. Thus all registers can be accessed with the same data width. The size column present in the subsequent table shows the *effective width* of the implemented registers. Registers with a smaller size than 32 bit will align their data to bit zero.

Accessing registers with a smaller size than their effective one may result in invalid data.

Offset to PCI Base Address 0	Register Name	Access	Size (Bit)
0x0000	Module Status and DCM 0/1 Status	R	32
0x0004	DCM Multiply/Divide 0	R/W	16
0x0008	DCM Multiply/Divide 1	R/W	16
0x000C	Global Channel Configuration	R/W	16
0x0010	Global Reset and Software Trigger Input	R/W[1]	16
0x0014	Sample Clock Configuration 0	R/W	8
0x0018	Sample Clock Configuration 1	R/W	8
0x001C	Trigger Configuration 0	R/W	8
0x0020	Trigger Configuration 1	R/W	8
0x0024	Channel Configuration 0	R/W	8
0x0028	Channel Configuration 1	R/W	8
0x002C	Channel Configuration 2	R/W	8
0x0030	Channel Configuration 3	R/W	8
0x0034	Channel Configuration 4	R/W	8
0x0038	Channel Configuration 5	R/W	8
0x003C	Channel Configuration 6	R/W	8
0x0040	Channel Configuration 7	R/W	8
0x0044	Channel DMA (Base) Descriptor Addresses 0	R/W	32
0x0048	Channel DMA (Base) Descriptor Addresses 1	R/W	32
0x004C	Channel DMA (Base) Descriptor Addresses 2	R/W	32
0x0050	Channel DMA (Base) Descriptor Addresses 3	R/W	32
0x0054	Channel DMA (Base) Descriptor Addresses 4	R/W	32
0x0058	Channel DMA (Base) Descriptor Addresses 5	R/W	32
0x005C	Channel DMA (Base) Descriptor Addresses 6	R/W	32
0x0060	Channel DMA (Base) Descriptor Addresses 7	R/W	32
0x0064	Channel Pre-Trigger Data Size 0	R/W	24
0x0068	Channel Pre-Trigger Data Size 1	R/W	24
0x006C	Channel Pre-Trigger Data Size 2	R/W	24
0x0070	Channel Pre-Trigger Data Size 3	R/W	24

[1] The read value will always be zero.

Offset to PCI Base Address 0	Register Name	Access	Size (Bit)
0x0074	Channel Pre-Trigger Data Size 4	R/W	24
0x0078	Channel Pre-Trigger Data Size 5	R/W	24
0x007C	Channel Pre-Trigger Data Size 6	R/W	24
0x0080	Channel Pre-Trigger Data Size 7	R/W	24
0x0084	AD Channel Data 0	R	16
0x0088	AD Channel Data 1	R	16
0x008C	AD Channel Data 2	R	16
0x0090	AD Channel Data 3	R	16
0x0094	AD Channel Data 4	R	16
0x0098	AD Channel Data 5	R	16
0x009C	AD Channel Data 6	R	16
0x00A0	AD Channel Data 7	R	16
0x00A4	General DMA Status	R	8
0x00A8	Channel DMA Status 0	R/W	16
0x00AC	Channel DMA Status 1	R/W	16
0x00B0	Channel DMA Status 2	R/W	16
0x00B4	Channel DMA Status 3	R/W	16
0x00B8	Channel DMA Status 4	R/W	16
0x00BC	Channel DMA Status 5	R/W	16
0x00C0	Channel DMA Status 6	R/W	16
0x00C4	Channel DMA Status 7	R/W	16
0x00C8	Channel DMA Buffer Fill-Level 0	R	32
0x00CC	Channel DMA Buffer Fill-Level 1	R	32
0x00D0	Channel DMA Buffer Fill-Level 2	R	32
0x00D4	Channel DMA Buffer Fill-Level 3	R	32
0x00D8	Channel DMA Buffer Fill-Level 4	R	32
0x00DC	Channel DMA Buffer Fill-Level 5	R	32
0x00E0	Channel DMA Buffer Fill-Level 6	R	32
0x00E4	Channel DMA Buffer Fill-Level 7	R	32
0x00E8	Revision Control Register	R	32

Table 5-4 : Register Map

5.3 DMA Descriptor Space

Offset to PCI Base Address 1	Register Name	Access	Size (Bit)
0x0 : 0x1FFF	Channel DMA Descriptor Memory	R/W	32

Table 5-5 : DMA Descriptor Space

6 Register Description

6.1 Module Status and DCM 0/1 Status Register

There are several components integrated in the module that need to be configured after power on respectively reset. This register summarizes accompany information and has to be checked before the module can be used.

Bit	Symbol	Description	Access	Reset Value
31:17	-	Reserved	R	0
16	STAT3	Core PLL Locked. The Core PLL generates the clocks for the DCMs and the QDR-II Memory Controllers. 1 = PLL is locked 0 = PLL is not locked	R	0
15:13	-	Reserved	R	0
12	STAT2	This bit indicates if the DCM 1 has been locked and operates correctly. 1 = DCM 1 locked 0 = DCM 1 not locked	R	0
11:9	-	Reserved	R	0
8	STAT1	This bit indicates if the DCM 0 has been locked and operates correctly 1 = DCM 0 locked 0 = DCM 0 not locked	R	0
7:5	-	Reserved	R	0
4	STAT0	QDR-II Memory Controllers' Ready Flag	R	0
3:1	-	Reserved	R	0
0	GSTAT	Whether the module is in working order is shown by this bit. 1 = Module is operating 0 = Module is not operating	R	0

Table 6-1 : Module Status Register (Address 0x0)

The module should not be used for DMA transmissions if the GSTAT bit is not asserted.

6.2 DCM Multiply/Divide 0/1

It is possible to define sample rates for the on board ADCs that are generated by the FPGA. For this the module provides two configurable digital clock managers (DCMs).

The sample frequency is adjusted in the way of defining a multiply and division value. The resulting fraction is applied on a source clock which has a frequency of 50 MHz.

The values for multiplication and division are grouped into a single data word. This is due to the physical interface of a Virtex-5 DCM.

Bit	Symbol	Description	Access	Reset Value
15:8	MULT	The selected multiplier is the assigned value plus one. Valid range is 1 up to 31	R/W	24
7:0	DIV	The selected divider is the assigned value plus one. Valid range is 0 up to 31	R/W	24

Table 6-2 : DCM Multiply/Divide Register (Address 0x4+ 0x4*DCM Number)

The legal value range reflect the Virtex-5 DCM specification. Do not use values other than the allowed ones. The minimum frequency that can be set is 32 MHz. The reset value adjusts a sample frequency of 50 MHz.

6.3 Global Channel Configuration

This register controls global (high-level) operation settings. This includes top-level activation, deactivation and trigger activation of certain channels.

The trigger activation bits have been implemented due to safety consideration. This prevents spurious interrupt-processing while requiring arming a channel before using.

Bit	Symbol	Description	Access	Reset Value
15	CATE7	To activation a channel for processing a subsequent trigger event the corresponding bit has to be set. The bit is reset after an event has processed. 0 = channel **not** armed 1 = channel armed	R/W	0
14	CATE6			
13	CATE5			
12	CATE4			
11	CATE3			
10	CATE2			
9	CATE1			
8	CATE0			
7	CHEN7	A channel is enabled (CHENx) for internal processing if the accompany bit is set. The configuration is done in the way 0 = disable channel 1 = enable channel The channel numbers 0-7 matches the ADC numbers 0-7.	R/W	0
6	CHEN6			
5	CHEN5			
4	CHEN4			
3	CHEN3			
2	CHEN2			
1	CHEN1			
0	CHEN0			

Table 6-3 : Global Channel Configuration Register (Address 0xC)

The channel trigger activation bits (CATx) are reset immediately after processing has started. It can be re-enabled first if corresponding channel processing has been finished.

Activation of a channel via the register above should be done after the channel has been configured correctly. Internal processing holds the corresponding channel processing logic in reset state if the channel enable bit (CHENx) is not set.

6.4 Global Reset and Software Trigger Input

Simulating external trigger signals can be performed by the subsequent register. Trigger events initiated by using this register are only processed if software trigger input is selected (per channel group).

Moreover the register can be used to reset the internal structure that is channel specific.

Bit	Symbol	Description	Access[2]	Reset Value
15	CRST7	Resetting a channel (x) is performed by writing 0x1 at the channel corresponding bit position (CRSTx).	R/W	0
14	CRST6		R/W	0
13	CRST5		R/W	0
12	CRST4		R/W	0
11	CRST3		R/W	0
10	CRST2		R/W	0
9	CRST1		R/W	0
8	CRST0		R/W	0
7:1	-	Reserved	R	0
0	SWT	Writing a value 0x1 at this bit position causes a single-cycle trigger input.	R/W	0

Table 6-4 : Global Reset and Software Trigger Input Register (Address 0x10)

The software generated trigger inputs are equivalent to external trigger signals.

[2] Read Value is always zero

6.5 Sample Clock Configuration 0/1

The register defines the source for the operating respectively sampling clock. This setting is applied due to physical restrictions (see chapter “Clock Distribution”) on a group of ADCs (0-3 and 4-7) where the below structure is the same for both groups.

Choosing a clock source must be done considering the ADCs specification to avoid a physical damage.

<table>
<tr><th>Bit</th><th>Symbol</th><th>Description</th><th>Access</th><th>Reset Value</th></tr>
<tr><td>7:4</td><td>SRSC</td><td>Sample Rate Source Configuration
<table>
<tr><th>Selection</th><th>Sample Clock</th></tr>
<tr><td>000x</td><td>AMC Reference Clock</td></tr>
<tr><td>001x</td><td>External Clock 2</td></tr>
<tr><td>010x</td><td>External Clock 1</td></tr>
<tr><td>011x</td><td>External Clock 0</td></tr>
<tr><td>1xx1</td><td>FPGA DCM 1</td></tr>
<tr><td>1xx0</td><td>FPGA DCM 0</td></tr>
<tr><td>Others</td><td>Reserved</td></tr>
</table></td><td>R/W</td><td>0</td></tr>
<tr><td>3:1</td><td>-</td><td>Reserved</td><td>R</td><td>0</td></tr>
<tr><td>0</td><td>SCEN</td><td>Channel Group Sample Clock Enable</td><td>R/W</td><td>0</td></tr>
</table>

Table 6-5 : Sample Clock Configuration Register (Address 0x14+ 0x4*Channel Group)

The ‘x’ inside the embedded table represents a do not care condition.

6.6 Trigger Configuration 0/1

Due to the internal processing all channels in one group (refer beginning of chapter “Functional Procedures”) are restricted to have the same trigger configuration. In accordance with that there are two trigger configuration registers that allows defining the corresponding settings.

<table>
<tr><th>Bit</th><th>Symbol</th><th>Description</th><th>Access</th><th>Reset Value</th></tr>
<tr><td>7</td><td>-</td><td>Reserved</td><td>R</td><td>0</td></tr>
<tr><td>6:4</td><td>TMSEL</td><td>Selection of the Trigger operation mode.

<table><tr><th>TMSEL</th><th>Input Selection</th></tr><tr><td>000</td><td>Disabled</td></tr><tr><td>001</td><td>Around Trigger Data Gathering</td></tr><tr><td>Others</td><td>Reserved</td></tr></table></td><td>R/W</td><td>0</td></tr>
<tr><td>3:0</td><td>TISEL</td><td>Trigger input (source) and polarity selection. There are three external trigger inputs provided by the SiCA.

<table><tr><th>TISEL</th><th>Input Selection</th><th>Polarity</th></tr><tr><td>0000</td><td>Disabled</td><td>-</td></tr><tr><td>0010</td><td rowspan="2">Trigger input 0 (TRIG_0)</td><td>Rising</td></tr><tr><td>0011</td><td>Falling</td></tr><tr><td>0100</td><td rowspan="2">Trigger input 1 (TRIG_1)</td><td>Rising</td></tr><tr><td>0101</td><td>Falling</td></tr><tr><td>0110</td><td rowspan="2">Trigger input 2 (TRIG_2)</td><td>Rising</td></tr><tr><td>0111</td><td>Falling</td></tr><tr><td>1000</td><td>Software Trigger</td><td>-</td></tr><tr><td>Others</td><td>Reserved</td><td>-</td></tr></table></td><td>R/W</td><td>0</td></tr>
</table>

Table 6-6 : Trigger Configuration Register (Address 0x1C+ 0x4*Channel Group)

6.7 Channel Configuration 0-7

ADC respectively channel specific configuration and steering settings are defined by these registers. The settings are made in addition to the general channel activation for internal processing (see chapter “Global Channel Configuration”).

Bit	Symbol	Description	Access	Reset Value
7	-	Reserved	R	0
6	EFSEL	Endian Format Selection steers the endianess of the DMA data in host memory. 0 = little endian 1 = big endian	R/W	0
5	OF	ADC Output Format 0 = offset binary output format 1 = 2’s complement output format	R/W	0
4	CDCS	ADC Clock Duty Cycle Stabilizer 0 = disabled 1 = enabled	R/W	0
3	OFSEN	Overflow Signalize Enable steers whether an ADC overflow signal is taken into the Channel DMA status register or not. 0 = suppress ADC overflow 1 = consider ADC overflow Activation causes DMA events if an overflow is signalized and interrupt-generation is enabled.	R/W	0
2	DMAEN	The bit steers the DMA channel data transmission. 0 = disable DMA operation 1 = enable DMA operation	R/W	0
1	INTEN	Using this flag allows to enable or disable the interrupt generation. 0 = disable channel Interrupts 1 = enable channel Interrupts	R/W	0
0	CHEN	Channel enable 0 = disable channel 1 = enable channel	R/W	0

Table 6-7 : Channel Configuration Register (Address 0x24+ 0x4*Channel)

If a channel is not enabled (CHEN), the corresponding Channel Data register will always be zero (0x0).

6.8 Channel DMA (Base) Descriptor Addresses 0-7

This register has two functions. On the one hand it holds an address into the embedded Block RAM (DMA Descriptor Space). This defines the first descriptor to load out of the memory for a channel. Consequently, it selects the first out of the three DMA descriptor instruction words. On the other hand it shows the Block RAM address of the current processed descriptor.

Bit	Symbol	Description	Access	Reset Value
31:26	-	Reserved	R	0
25:16	CADDR	The pointer reflects the memory address of the current loaded descriptor.	R	0
15:10	-	Reserved	R	0
9:0	BADDR	This address points into the memory where the channel base address descriptor is placed.	R/W	0

Table 6-8 : Channel DMA (Base) Descriptor Address Register (Address 0x44+ 0x4*Channel)

6.9 Channel Pre-Trigger Data Size

In addition to the trigger mode that allows transmitting data before and after the trigger event this register is used. The number of samples that are transmitted before the trigger event (pre-trigger data) is defined by the register value.

Bit	**Symbol**	**Description**	**Access**	**Reset Value**
23:18	-	Reserved	R	0
17:0	NUM	Number of samples that is transmitted via DMA as pre-trigger data	R/W	0

Table 6-9 : Channel Pre-Trigger Data Register (Address 0x64+ 0x4*Channel)

Setting a value different then one requires that the number of samples has already been acquired by the module. If this condition is violated, the obtained data will be invalid.

6.10 Channel Data 0–7

Reading the current ADC sample of a channel (0-7) is possible via these registers. For this, the channel must be enabled in the corresponding Channel Configuration register (CHEN), since otherwise the read value will always be zero (0x0).

The intention of this register interface is more a static evaluation of the input data, e.g. channel calibration.

The channel data is subdued a sign extension for both ADC operating modes (2's complement and offset binary output format).

Bit	Symbol	Description	Access	Reset Value
15:13	SEXT	Sign Extension	R	0
12:0	DAT	In dependency of the selected ADC operation mode the converted data is readable through this register	R	0

Table 6-10: Channel Data Register (Address 0x84+ 0x4*Channel)

The value read from such a register cannot be the last that has been sampled due to the AD Converter 5-stage internal pipeline.

6.11 General DMA Status

In order to bundle the DMA status over all channels this register has been implemented. This simplifies the detection of the (DMA) event-generating channel (interrupt source).

Bit	Symbol	Description	Access	Reset Value
7	ECH7	The bits indicate an event (ECHx) on channel (x). 1 = event pending 0 = no event on channel	R	0
6	ECH6		R	0
5	ECH5		R	0
4	ECH4		R	0
3	ECH3		R	0
2	ECH2		R	0
1	ECH1		R	0
0	ECH0		R	0

Table 6-11: Global DMA Status Register (Address 0xA4)

The channel event information is obtained from the single channel DMA status registers. Thus it must not be cleared here.

6.12 Channel DMA Status 0-7

The different events that may occur during processing are shown afterwards. There are eight such registers – one for every channel – which source the General DMA Status register.

Bit	Symbol	Description	Access	Reset Value
15:13	-	Reserved	R	0
12	OFERR	This bit indicates that an ADC overflow has occurred. If this information is desired, it can be enabled in the corresponding Channel Configuration register.	R/W	0
11:10	-	Reserved	R	0
9	SEQLE	The Sequence Last Event (SEQLE) indicates that a Descriptor has been processed that is the last of a Linked List.	R/W	0
8	SEQHI	Sequence Host Interrupt (SEQHI) is asserted if a Descriptor of a Linked List has been processed with the Host Interrupt flag set.	R/W	0
7:2	-	Reserved	R	0
4	DSINF	Data sampling has stopped due to the load of an invalid descriptor	R/W	0
3:1	-	Reserved	R	0
0	OVERR	Showing that the ADC channel data got corrupt (invalid) due to an internal buffer overrun is the purpose of this bit. DMA transmission is stopped in such a case. (*)	R/W	0

Table 6-12: Channel DMA Status Register (Address 0xA8+ 0x4*Channel)

The status flags have to be cleared for acknowledgement. The errors marked with an asterisk (*) require a channel reset to get into a defined state.
All flags are cleared automatically after processing start (valid trigger input).

6.13 Channel DMA Buffer Fill-Level 0-7

These registers offer the possibility to monitor the current DMA window write address of a channel.

Bit	Symbol	Description	Access	Reset Value
31:0	CADDR	Current DMA Window Write Position (Address)	R	0

Table 6-13: Channel DMA Buffer Fill-Level Register (Address 0xC8+ 0x4*Channel)

The value after channel setup and activation is the one that has been read from the defined DMA (base) descriptor.

6.14 Revision Control Register

The register content reflects the firmware revision currently implemented on the module.

Bit	**Symbol**	**Description**	**Access**	**Reset Value**
31:0	CREV	Code Revision of the FPGA Firmware	R	-[3]

Table 6-14: Revision Control Register (Address 0xE8)

[3] The version of the Firmware is not fixed and depends besides others on the hardware version.

7 Interrupts

Legacy interrupt messages and message signaled interrupts (MSIs) are supported by the firmware. In dependency of the chosen mode - defined by the PCI header and system (driver) software – one of the two methods is used.

In legacy interrupt mode INTA interrupts are asserted in case of an interrupt event.

The MSI mode uses a single vector for interrupt signalize. A larger vector would simplify the detection of an interrupt source but increase the implementation and system resource efforts. Moreover it is not safe to obtain the resources for a multi-vector MSI from the system.

7.1 Interrupt Sources

IRQ	Description
IRQCH0	Every channel can generate an interrupt event if it is enabled appropriately. The General DMA Status register can be used to detect which channel caused the interrupt. The corresponding Channel DMA Status register must be read afterwards to obtain further information about the interrupt.
IRQCH1	
IRQCH2	
IRQCH3	
IRQCH4	
IRQCH5	
IRQCH6	
IRQCH7	

Table 7-1 : Interrupt Sources

7.2 Interrupt Handling

IRQ	Description	IRQ Enable	IRQ Ack.
IRQCH0	Channel 0 Interrupt	Channel Configuration Register	Corresponding Channel Status Register
IRQCH1	Channel 1 Interrupt		
IRQCH2	Channel 2 Interrupt		
IRQCH3	Channel 3 Interrupt		
IRQCH4	Channel 4 Interrupt		
IRQCH5	Channel 5 Interrupt		
IRQCH6	Channel 6 Interrupt		
IRQCH7	Channel 7 Interrupt		

Table 7-2 : Interrupt Handling

For all interrupts should be noted that every DMA channel is treated as one source. This causes that an interrupt that occurs while a pending interrupt is present will not generate a new interrupt signal. Consequently interrupts have to be acknowledged after their occurrence.

The use of Message Signaled Interrupts may introduce spurious interrupts as described in the PCI Specification.

8 FPGA

The TAMC900 provides a Virtex-5 FPGA in FFG665 for customer programming and board control.

Information about the FPGA pin assignment is part of the engineering documentation.

8.1 Configuration

The FPGA is configured from a Xilinx Platform Flash. The flash has the potential to store different FPGA code revisions.

The Module Management Controller (MMC) can be programmed to select the code revision that is loaded into the FPGA. By default, Code Revision 0 is loaded into the FPGA.

On the TAMC900, the Xilinx Platform Flash is configured to store multiple code revisions. The FPGA is configuration Master and loads Revision 0 by default.
Pay attention to this while generating PROM files.

The Platform Flash or the FPGA are programmed using the Payload JTAG interface. The Payload JTAG interface is accessible via the “Payload JTAG connector” (J4).

The on board CPLD is part of the “Payload JTAG Cain”. TEWS recommends setting the CPLD in Bypass Mode during FPGA or Platform Flash JTAG operations.

8.2 MMC Interface

The FPGA has the following signals which are connected to the MMC:

Interface	Description
FUNC_LED2	This signal can be used to flash the USER LED in the front panel of the TAMC900. A rising or falling edge of FUNC_LED2 triggers the MMC to turn the USER LED off for app. 100ms.
EKEY[4:1]	These signals can be used to transmit connectivity data from the MMC to the FPGA. The implementation in MMC and FPGA has to be done by the customer. Please refer to the chapter “Module Management Controller (MMC)” for more information.
I2C	This I2C Interface can be used by the FPGA to read connectivity data from the MMC. The implementation in MMC and FPGA has to be done by the customer. Please refer to chapter “Module Management Controller (MMC)” for more information.
RXD0 / TXD0	These signals can be used to implement a serial communication between FPGA and MMC. By default, this is used as debug-output of the MMC. Any other implementation in MMC and FPGA has to be done by the customer.

Table 8-1 : FPGA Signals connected to the MMC

8.3 AMC Interface

The Multi-Gigabit Transceivers (MTGs) of the FPGA are connected to Port 4 – 11 of the AMC Interface.

Signal Name	Virtex-5 Pin	AMC Conn. Pin (TAMC900-10)
TX0+	B2	44
TX0-	C2	45
RX0+	C1	47
RX0-	D1	48
TX1+	G2	50
TX1-	F2	51
RX1+	F1	53
RX1-	E1	54
TX2+	H2	59
TX2-	J2	60
RX2+	J1	62
RX2-	K1	63
TX3+	N2	65
TX3-	M2	66
RX3+	M1	68
RX3-	L1	69
TX4+	P2	91
TX4-	R2	90
RX4+	R1	88
RX4-	T1	87
TX5+	W2	97
TX5-	V2	96
RX5+	V1	94
RX5-	U1	93
TX6+	Y2	103
TX6-	AA2	102
RX6+	AA1	100
RX6-	AB1	99
TX7+	AE2	109
TX7-	AD2	108
RX7+	AD1	106
RX7-	AC1	105

Table 8-2 : AMC Fabric Interface connections to the FPGA

FCLKA of the AMC Interface is multiplied by the factor of 2.5 and then connected to a MGT Clock Input:

Signal Name	Virtex-5 Pin	AMC Conn. Pin
AMC_REFCLK+	T3	80
AMC_REFCLK-	V1	81

Table 8-3 : AMC FCLKA connection to the FPGA

It is free to the customer what kind of interfaces to implement using the FPGA logic recourses.

If any other interfaces than PCI-Express are implemented, it is necessary to adapt the Connectivity Records of the MMC. Otherwise, proper operation of the TAMC900 is not possible.

8.4 RAM Interface

The RAM interface to access the QDR-II SRAM of the TAMC900 has to be implemented in the FPGA. TEWS recommends using the Xilinx Memory Interface Generator (MIG) to build the RAM interface logic.

Please refer to the QDR-II SRAM Data Sheet and the Xilinx documentation for more details.

8.5 ADC Interface

The ADC data lines and the corresponding sample clocks are routed to the FPGA.

For accurate and precise sampling of the ADC data, TEWS recommends using the ILOGIC FlipFlops of the FPGA. Please refer to the sample application for more details.

Configuration of the ADCs is done via the on board CPLD. The on board CPLD is accessible via the FPGA. Please refer to the chapter “On Board CPLD” for more details.

9 ADCs

The TAMC900 provides eight high speed ADCs (LTC2254 from Linear Technologies). The LTC2254 provides 14 bit resolution. Configuration of the ADC (Output Enable, Shutdown, Output Format, …) is done by the on board CPLD. It can be controlled via the FPGA.

The LTC2254 differential inputs are routed to the Signal Conditioning Adapter (SiCA) connector. For the pin assignment, please refer to chapter “I/O Connector”.

Any signal conditioning of the analog inputs is not done on the TAMC900. This is done by the SiCA. The SiCA also carries the I/O connectors accessible through the face plate.

9.1 AC / DC Characteristics

9.1.1 Min / Max Sample Rate

The minimum sample rate of the LTC2254 is 1 Msps.

The maximum sample rate is 105 Msps.

9.1.2 Input Voltage Range

The differential input voltage range of the LTC2254 is 2V, and can be set to 1V based on the application.

The 2V input range will provide the best signal-to-noise performance while maintaining excellent SFDR.

The 1V input range will have better SFDR performance, but the SNR will degrade by 5.7dB.

By default, the TAMC900 sets the input voltage range of all ADCs to 2V. The Common Mode Voltage of the ADC Differential Inputs is 1.5V.

9.1.3 Input Frequency Range

The LTC2254 provides a full power bandwidth from DC to 640 MHz.

9.2 Operational Modes

Each ADC may be placed in shutdown or nap mode to conserve power.

In sleep mode, which powers down all circuitry including the reference, the ADC typically dissipates 1mW. When exiting sleep mode it will take milliseconds for the output data to become valid because the reference capacitors have to recharge and stabilize.

In nap mode, the ADC typically dissipates 15mW. In nap mode, the on-chip reference circuit is kept on, so that recovery from nap mode is faster than that from sleep mode, typically taking 100 clock cycles. In both sleep and nap modes, all digital outputs are disabled and enter the Hi-Z state.

If the sample clock is stopped during normal ADC operation, the output data becomes invalid. After the sample clock is turned on, it takes app. 4000 clock cycles until the output data becomes valid.

9.2.1 Data Format

The ADCs parallel digital output can be selected for offset binary or 2’s complement format. The following table shows the relationship between the analog input voltage, the digital ADC output data, and the overflow bit (OF).

A_{IN}+ - A_{IN}- (2V Range)	OF	Data (Offset Bin.)	Data (2's Compl.)
>+1.000000V	1	0x3FFF	0x1FFF
+0.999878V	0	0x3FFF	0x1FFF
+0.999756V	0	0x3FFE	0x1FFE
...	...	...	...
+0.000122V	0	0x2001	0x0001
0.000000V	0	0x2000	0x0000
-0.000122V	0	0x1FFF	0x3FFF
-0.000244V	0	0x1FFE	0x3FFE
...	...	...	...
-0.999878V	0	0x0001	0x2001
-1.000000V	0	0x0000	0x2000
<-1.000000V	1	0x0000	0x2000

Table 9-1 : ADC Output Data Format

The FPGA Logic performs a sign extension of the ADC-Data. Refer to chapter "Channel Logic" for more information.

9.2.2 Clock Duty Cycle Stabilizer

An optional clock duty cycle stabilizer circuit ensures high performance even if the input clock has a non 50% duty cycle. Using the clock duty cycle stabilizer is recommended for most applications.

This circuit uses the rising edge of the CLK pin to sample the analog input. The falling edge of CLK is ignored and the internal falling edge is generated by a phase-locked loop. The input clock duty cycle can vary from 40% to 60% and the clock duty cycle stabilizer will maintain a constant 50% internal duty cycle.

If the clock is turned off for a long period of time, the duty cycle stabilizer circuit will require a hundred clock cycles for the PLL to lock onto the input clock.

For applications where the sample rate needs to be changed quickly, the clock duty cycle stabilizer can be disabled. If the duty cycle stabilizer is disabled, care should be taken to make the sampling clock have a 50% (±5%) duty cycle.

9.3 Shielding

The TAMC900 allows adding a cover to shield its ADCs. See picture below for cover mounting area dimensions. The cover must ensure a minimum free height of 3 mm and sufficient cooling of the ADCs.

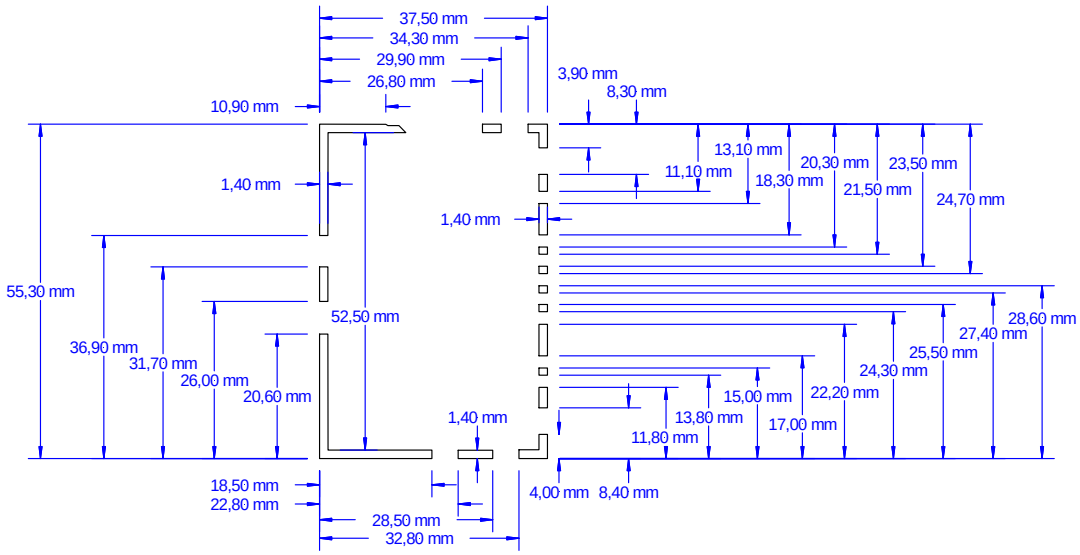

Figure 9-1 : Shielding cover dimensions

10 Memory

The TAMC900 provides 4 MByte QDR-II SRAM.

2 RAM devices with 18 bit wide data bus each are used to implement the 4 MByte RAM of the TAMC900.

The two RAMs have fully independent interfaces to the FPGA:

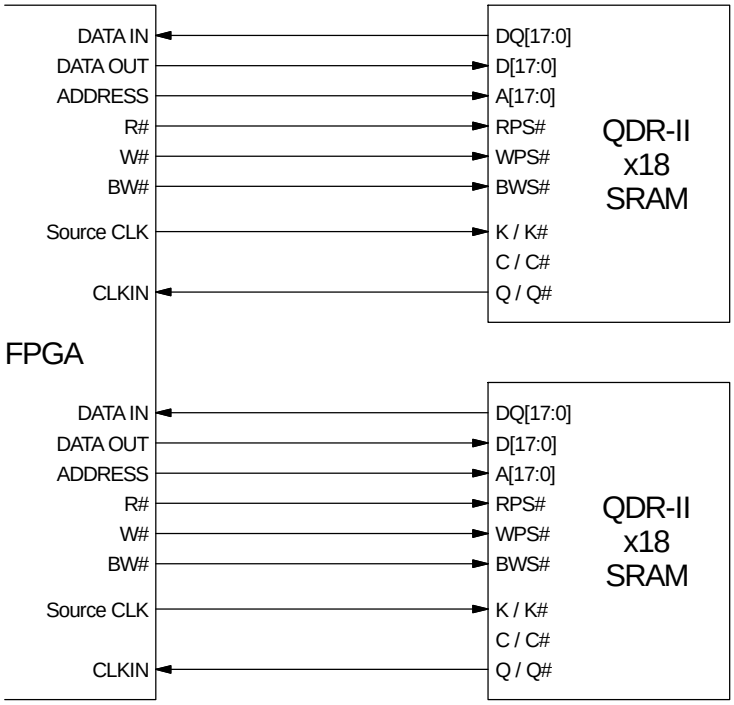

Figure 10-1: QDR-II SRAM Interface

The TAMC900 uses 4 Burst SRAM to lower address-bus switching speed and simultaneously achieve read and write accesses to independent addresses of the SRAM without any wait cycles.

The RAM can be clocked with up to 250 MHz. Effective maximum access speed depends on FPGA speed and available routing resources.

For timing details regarding the QDR-II SRAM interface, please refer to the QDR-II SRAM data sheet.

11 Clock Distribution

The TAMC900 has several Clock sources:

- 3 Clock Inputs from the SiCA
- 1 Clock Input from AMC interface connector
- 2 local Clocks
- 2 Clock outputs of the FPGA

The Clocks are distributed to the ADCs and the FPGA. Configuration of the Clock Distribution is done via the on board CPLD. Refer to the CPLD registers description for programming details.

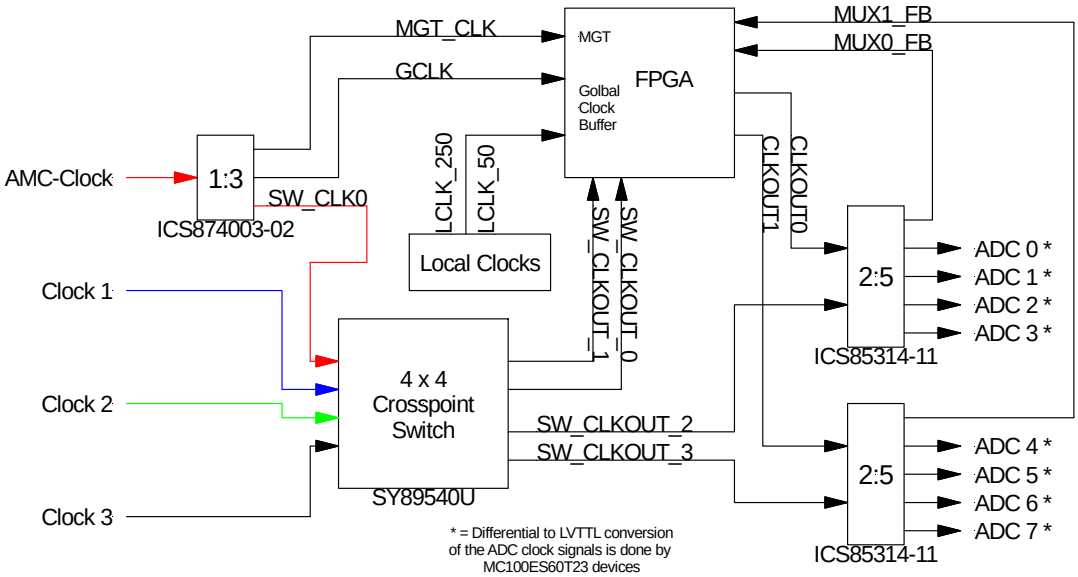

Figure 11-1: Clock Distribution Block Diagram

The external clock inputs “Clock 1”, “Clock 2” and “Clock 3” are LVDS compatible.

“AMC Clock” is the FCLKA signal from the AMC connector, routed through a jitter attenuator (ICS874003-02) to achieve a max. Jitter below 40ps even if the original FCLKA has much more jitter. In most systems, FCLKA is a 100 MHz PCI-Express reference clock that may be a SSC (Spread Spectrum Clock)

The first local Clock (LCLK_250) is an on board 250 MHz oscillator with LVDS signaling. It is possible to turn the oscillator off via the on board CPLD. The second local clock is a 50 MHz (single ended) clock. The following table provides the pin assignment of the FPGA local clock inputs:

Signal Name	Virtex-5 Pin
LCLK_250+	D15
LCLK_250-	E15
LCLK_50	AC14

Table 11-1: FPGA Local Clock (LCLK_250) Inputs

The two FPGA Clock outputs (CLKOUT0 and CLKOUT1) provide the ability to run the ADCs with user specified clocks that are generated in the FPGA. The following table provides the pin assignment of the clock outputs:

Signal Name	Virtex-5 Pin	to ADC
V5_CLKOUT0+	V6	0 – 3
V5_CLKOUT 0-	V7	
V5_CLKOUT 1+	J5	4 – 7
V5_CLKOUT 1-	J6	

Table 11-2: FPGA Clock Outputs

The TAMC900 provide two feedback clocks (MUX0_FB and MUX1_FB) that return the ADC clocks back to the FPGA. The following table provides the pin assignment of the feedback clock inputs:

Signal Name	Virtex-5 Pin	from ADC
MUX0_FB+	E13	0 – 3
MUX0_FB-	E12	
MUX1_FB+	E18	4 – 7
MUX1_FB-	F19	

Table 11-3: ADC Feedback Clocks

12 LVDS Link

The TAMC900 provides a differential interface from the FPGA to the SiCA connector. 10 differential pairs out of one FPGA I/O bank are routed to the SiCA connector. For the Virtex-5 and SiCA connector pin assignment, see table below.

The use of these lines is free to the customer. The FPGA programming provided by TEWS does not use these lines. Care must be taken to avoid damaging the FPGA.

Signal Name	Virtex-5 Pin	SiCA Pin
LVDS_0+	K6	18
LVDS_0-	K7	20
LVDS_1+	K8	24
LVDS_1-	L7	26
LVDS_2+	M7	30
LVDS_2-	L8	32
LVDS_3+	R6	36
LVDS_3-	T7	38
LVDS_4+	P6	44
LVDS_4-	N6	46
LVDS_5+	M6	50
LVDS_5-	N7	52
LVDS_6+	N8	56
LVDS_6-	P8	58
LVDS_7+	R8	62
LVDS_7-	R7	64
LVDS_8+	Y6	68
LVDS_8-	Y5	70
LVDS_9+	G6	74
LVDS_9-	H6	76

Table 12-1: LVDS Link Pin Assignment

The maximum input voltage of the Virtex-5 I/Os is 2.5 Volt.
The TAMC900 will be damaged if higher voltage levels are used.

13 On Board CPLD

The On Board CPLD of the TAMC900 is used for static board configuration and provides the following functions:

- QDR-II SRAM DLL enable / disable
- ADC Shutdown
- ADC Output Enable
- ADC Mode selection (all 4 modes available)
- Control of 16 GP I/O pins to the SiCA.
- Enabling / disabling of the local FPGA clock
- Controlling of the jitter attenuator
- Control of the local clock distribution

Please refer to the following subsections for more information.

13.1 Interface to FPGA

All functions of the CPLD are accessible by the User FPGA via an easy to use interface. The clock is driven by the CPLD. Per default this is set to 25 MHz. It can be set to 12.5 MHz if necessary.

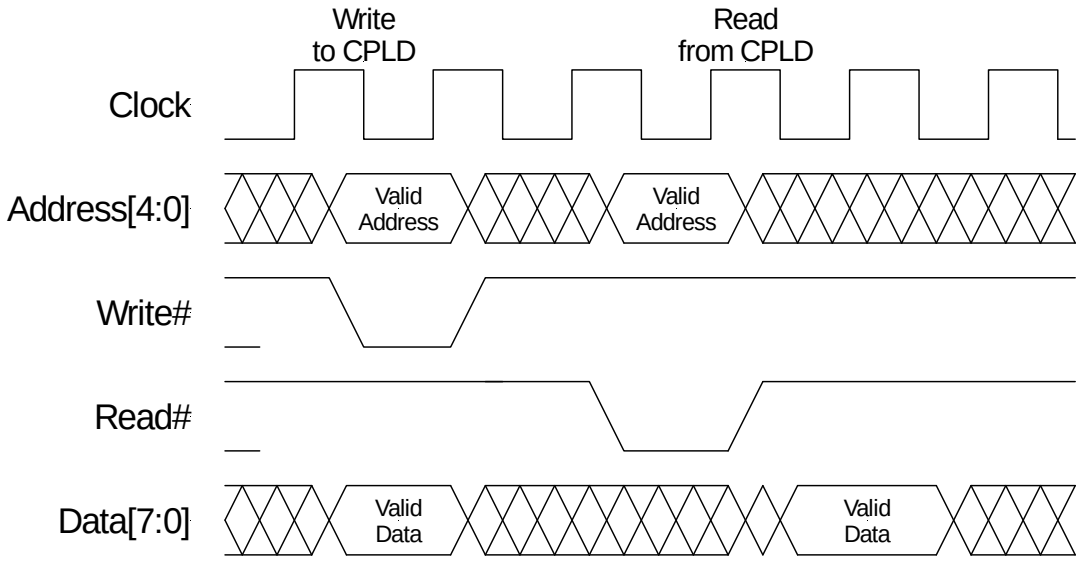

Figure 13-1: Timing of FPGA-CPLD Interface

13.2 CPLD Register Description

The following registers are implemented in the on board CPLD of the TAMC900. They are used for static board configuration. For user access, the CPLD provides an interface to the FPGA.

CPLD-Address (hex)	Description	Register access	**Value after Reset**
0x00	ADC 0 Control Register	R/W	0x02
0x01	ADC 1 Control Register	R/W	0x02
0x02	ADC 2 Control Register	R/W	0x02
0x03	ADC 3 Control Register	R/W	0x02
0x04	ADC 4 Control Register	R/W	0x02
0x05	ADC 5 Control Register	R/W	0x02
0x06	ADC 6 Control Register	R/W	0x02
0x07	ADC 7 Control Register	R/W	0x02
0x08	SiCA Input Register 1	R	depends on SiCA
0x09	SiCA Input Register 2	R	depends on SiCA
0x0A	SiCA Output Register 1	R/W	0x00
0x0B	SiCA Output Register 2	R/W	0x00
0x0C	SiCA Output Enable Register 1	R/W	0x00
0x0D	SiCA Output Enable Register 2	R/W	0x00
0x0E – 0x0F	reserved	R	0x00
0x10	Crosspoint Switch Control Register	R/W	0x00
0x11	Clock Mux Control Register	R/W	0x00
0x12	Jitter Attenuator Control Register	R/W	0x10
0x13 – 0x17	reserved	R	0x00
0x18	General board Control Register	R/W	0x00
0x19 – 0x1E	reserved	R	0x00
0x1F	Revision Control Register	R	Code Revision of the CPLD

Table 13-1: CPLD Register Overview

13.2.1 ADC x Control Register (Address 0x00 to 0x07)

Bit	Symbol	Description	Access	Reset Value
7	-	reserved for future use	R	0
6	-	reserved for future use	R	0
5	OF	Output Format 0 = offset binary output format 1 = 2's complement output format	R/W	0
4	CDCS	Clock Duty Cycle Stabilizer 0 = OFF 1 = ON	R/W	0
3	-	reserved for future use	R	0
2	-	reserved for future use	R	0
1	SHDN	ADC Shutdown 0 = normal operation 1 = shutdown corresponding ADC	R/W	1
0	OE	ADC output Enable (enables the digital outputs of the ADC) 0 = output disable 1 = output enable	R/W	0

Table 13-2: ADC x Control Register (Address 0x00 to 0x07)

13.2.2 SiCA Input Register 1 (Address 0x08)

Bit	Symbol	Description	Access	Reset Value
7	I_SiCA15	Displays the value of the corresponding SiCA general purpose pin. Value after reset depends on the SiCA.	R	-
6	I_SiCA14		R	-
5	I_SiCA13		R	-
4	I_SiCA12		R	-
3	I_SiCA11		R	-
2	I_SiCA10		R	-
1	I_SiCA9		R	-
0	I_SiCA8		R	-

Table 13-3: SiCA Input Register 1 (Address 0x08)

13.2.3 SiCA Input Register 2 (Address 0x09)

Bit	Symbol	Description	Access	Reset Value
7	I_SiCA7	Displays the value of the corresponding SiCA general purpose pin. Value after reset depends on the SiCA.	R/W	-
6	I_SiCA6		R/W	-
5	I_SiCA5		R/W	-
4	I_SiCA4		R/W	-
3	I_SiCA3		R/W	-
2	I_SiCA2		R/W	-
1	I_SiCA1		R/W	-
0	I_SiCA0		R/W	-

Table 13-4: SiCA Input Register 2 (Address 0x09)

13.2.4 SiCA Output Register 1 (Address 0x0A)

Bit	Symbol	Description	Access	Reset Value
7	O_SiCA15	The value written to this register is displayed to the SiCA on the corresponding SiCA general purpose pins if the pin is defined as output in the corresponding SiCA Output Enable Register.	R/W	0
6	O_SiCA14		R/W	0
5	O_SiCA13		R/W	0
4	O_SiCA12		R/W	0
3	O_SiCA11		R/W	0
2	O_SiCA10		R/W	0
1	O_SiCA9		R/W	0
0	O_SiCA8		R/W	0

Table 13-5: SiCA Output Register 1 (Address 0x0A)

13.2.5 SiCA Output Register 2 (Address 0x0B)

Bit	Symbol	Description	Access	Reset Value
7	O_SiCA7	The value written to this register is displayed to the SiCA on the corresponding SiCA general purpose pins if the pin is defined as output in the corresponding SiCA Output Enable Register.	R/W	0
6	O_SiCA6		R/W	0
5	O_SiCA5		R/W	0
4	O_SiCA4		R/W	0
3	O_SiCA3		R/W	0
2	O_SiCA2		R/W	0
1	O_SiCA1		R/W	0
0	O_SiCA0		R/W	0

Table 13-6: SiCA Output Register 2 (Address 0x0B)

13.2.6 SiCA Output Enable Register 1 (Address 0x0C)

Bit	Symbol	Description	Access	Reset Value
7	EN_SiCA15	Sets the corresponding SiCA general purpose pins as input or output. 0 = CPLD input 1 = CPLD output	R/W	0
6	EN_SiCA14		R/W	0
5	EN_SiCA13		R/W	0
4	EN_SiCA12		R/W	0
3	EN_SiCA11		R/W	0
2	EN_SiCA10		R/W	0
1	EN_SiCA9		R/W	0
0	EN_SiCA8		R/W	0

Table 13-7: SiCA Output Enable Register 1 (Address 0x0C)

13.2.7 SiCA Output Enable Register 2 (Address 0x0D)

Bit	Symbol	Description	Access	Reset Value
7	EN_SiCA7	Sets the corresponding SiCA general purpose pins as input or output. 0 = CPLD input 1 = CPLD output	R/W	0
6	EN_SiCA6		R/W	0
5	EN_SiCA5		R/W	0
4	EN_SiCA4		R/W	0
3	EN_SiCA3		R/W	0
2	EN_SiCA2		R/W	0
1	EN_SiCA1		R/W	0
0	EN_SiCA0		R/W	0

Table 13-8: SiCA Output Enable Register 2 (Address 0x0D)

13.2.8 Crosspoint Switch Control Register (Address 0x10)

Bit	Symbol	Description	Access	Reset Value
7	OUT3_ SEL1	Clock Select for ADC 4 to 7: 00 = Clock input 0 : SW_CLK0 (100MHz PCIe Ref-Clock) 01 = Clock input 1 : EXT_CLK2 10 = Clock input 2 : EXT_CLK1 11 = Clock input 3 : EXT_CLK0	R/W	0
6	OUT3_ SEL0		R/W	0
5	OUT2_ SEL1	Clock Select for ADC 0 to 3: 00 = Clock input 0 : SW_CLK0 (100MHz PCIe Ref-Clock) 01 = Clock input 1 : EXT_CLK2 10 = Clock input 2 : EXT_CLK1 11 = Clock input 3 : EXT_CLK0	R/W	0
4	OUT2_ SEL0		R/W	0
3	OUT1_ SEL1	Clock Select for FPGA_CLK_IN1: 00 = Clock input 0 : SW_CLK0 (100MHz PCIe Ref-Clock) 01 = Clock input 1 : EXT_CLK2 10 = Clock input 2 : EXT_CLK1 11 = Clock input 3 : EXT_CLK0	R/W	0
2	OUT1_ SEL0		R/W	0
1	OUT0_ SEL1	Clock Select for FPGA_CLK_IN0: 00 = Clock input 0 : SW_CLK0 (100MHz PCIe Ref-Clock) 01 = Clock input 1 : EXT_CLK2 10 = Clock input 2 : EXT_CLK1 11 = Clock input 3 : EXT_CLK0	R/W	0
0	OUT0_ SEL0		R/W	0

Table 13-9: Crosspoint Switch Control Register (Address 0x10)

13.2.9 Clock Mux Control Register (Address 0x11)

Bit	Symbol	Description	Access	Reset Value
7	-	reserved for future use	R	0
6	-	reserved for future use	R	0
5	SEL_47	Select Clock for ADC 4 to 7 0 = route clock from crosspoint switch to ADCs 1 = route Clock from FPGA to ADCs	R/W	0
4	EN_47	Enable Clock outputs to ADC 4 to 7. 0 = disable 1 = enable	R/W	0
3	-	reserved for future use	R	0
2	-		R	0
1	SEL_03	Select Clock for ADC 0 to 3 0 = route clock from crosspoint switch to ADCs 1 = route Clock from FPGA to ADCs	R/W	0
0	EN_03	Enable Clock outputs to ADC 0 to 3. 0 = disable 1 = enable	R/W	0

Table 13-10: Clock Mux Control Register (Address 0x11)

13.2.10 Jitter Attenuator Control Register (Address 0x12)

<table>
<tr><th>Bit</th><th>Symbol</th><th>Description</th><th>Access</th><th>Reset Value</th></tr>
<tr><td>7</td><td>-</td><td>reserved for future use</td><td>R</td><td>0</td></tr>
<tr><td>6</td><td>SEL2</td><td rowspan="3">Frequency select pins for QAx and QB0 outputs:

<table><tr><th colspan="3">Inputs</th><th colspan="2">Outputs</th></tr><tr><th>SEL2</th><th>SEL1</th><th>SEL0</th><th>QA0, QA1</th><th>QB0</th></tr><tr><td>0</td><td>0</td><td>0</td><td>2.5x f_{IN}</td><td>2.5x f_{IN}</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1x f_{IN}</td><td>2.5x f_{IN}</td></tr><tr><td>0</td><td>1</td><td>0</td><td>1.25x f_{IN}</td><td>2.5x f_{IN}</td></tr><tr><td>1</td><td>1</td><td>0</td><td>2.5x f_{IN}</td><td>1.25x f_{IN}</td></tr><tr><td>0</td><td>0</td><td>1</td><td>2.5x f_{IN}</td><td>1x f_{IN}</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1x f_{IN}</td><td>1.25x f_{IN}</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1.25x f_{IN}</td><td>1x f_{IN}</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1.25x f_{IN}</td><td>1.25x f_{IN}</td></tr></table>Input frequency (f_{IN}) should be 100 MHz (AMC FCLKA).</td><td>R/W</td><td>0</td></tr>
<tr><td>5</td><td>SEL1</td><td>R/W</td><td>0</td></tr>
<tr><td>4</td><td>SEL0</td><td>R/W</td><td>1</td></tr>
<tr><td>3</td><td>-</td><td>reserved for future use</td><td>R</td><td>0</td></tr>
<tr><td>2</td><td>MR</td><td>Reset
0 = normal operation
1 = reset device and disable outputs</td><td>R/W</td><td>0</td></tr>
<tr><td>1</td><td>OEB</td><td>Output disable for the QB0 clock
0 = enable
1 = disable</td><td>R/W</td><td>0</td></tr>
<tr><td>0</td><td>OEA</td><td>Output disable for the QA0 and QA1 clocks
0 = enable
1 = disable</td><td>R/W</td><td>0</td></tr>
</table>

Table 13-11: Jitter Attenuator Control Register (Address 0x12)

Do not change the frequency of QA0/QA1.

13.2.11 General Board Control Register (Address 0x18)

Bit	Symbol	Description	Access	Reset Value
7	-	reserved for future use	R	0
6	-	reserved for future use	R	0
5	RAMDLL1	Disable DLL of RAM 1 0 = enable 1 = disable	R/W	0
4	RAMDLL0	Disable DLL of RAM 0 0 = enable 1 = disable	R/W	0
3	-	reserved for future use	R	0
2	-	reserved for future use	R	0
1	-	reserved for future use	R	0
0	LOC_CLK	Disable local 250 MHz clock 0 = enable 1 = disable	R/W	0

Table 13-12: General Board Control Register (Address 0x18)

14 Signal Conditioning Adapter (SiCA)

The SiCA holds the connector for the analog inputs, the connectors for the clock and trigger inputs and the analog signal conditioning for the TAMC900. Depending on customer needs, the form and function of the SiCA can vary. For more information please refer to the corresponding SiCA User Manual.

The SiCA is powered from the TAMC900 with +6 Volts from a switched power supply. Any other voltages needed must be generated on the SiCA.

A 120 pin high speed connector from Samtec (QSE-060-01-L-D-A) is used on the TAMC900 to interface with the SiCA. The pin assignment of this connector can be found in the chapter “I/O Connector”.

14.1 Connector and Standoff Positions

Five mounting positions for Standoffs are provided by the TAMC900 to secure the SiCA on the TAMC900. Depending on the size of the SiCA, all or only a subset of these can be used to secure the SiCA on the TAMC900. See figure below for the exact position.

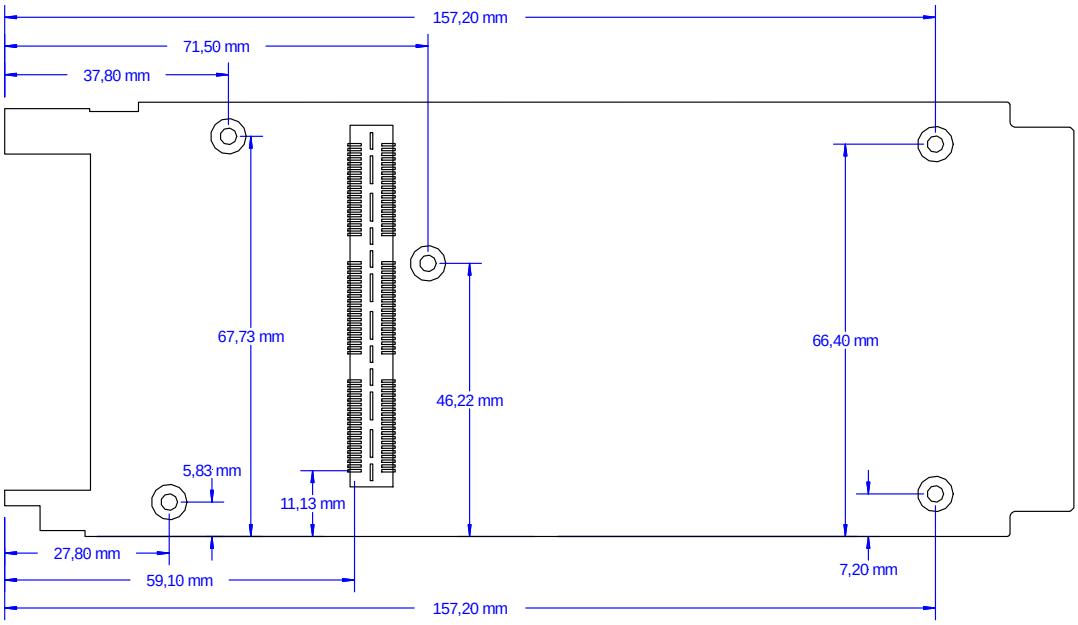

Figure 14-1: TAMC900 Connector and Standoff Positions

14.2 Mating Connector

The mating connector connects the SiCA with the TAMC900. The Samtec QSE/QTE Series is used, and the height of the connector mounted on the SiCA dictates the stacking height of the SiCA above the TAMC900. Possible connectors are (e.g):

Pin-Count	120 pins
Connector Type	Q Strip (High Speed SMT)
Source & Order Info	QTE-060-02-L-D-A (8 mm stacking height)

Pin-Count	120 pins
Connector Type	Q Strip (High Speed SMT)
Source & Order Info	QTE-060-03-L-D-A (11 mm stacking height)

Table 14-1: I/O mating connectors

14.3 Board and Component Height

The height of the SiCA above the AMC depends on the mated stacking height of the connector used.

When selecting a stacking height, care must be taken to ensure enough space for the I/O connectors of the SiCA.

Take care not to violate the maximum component height according to AMC.0.

The maximum component height on the SiCA depends on the component height of the AMC below the SiCA, and the stacking height. See figure below for more details (Shaded areas are “no placement” areas).

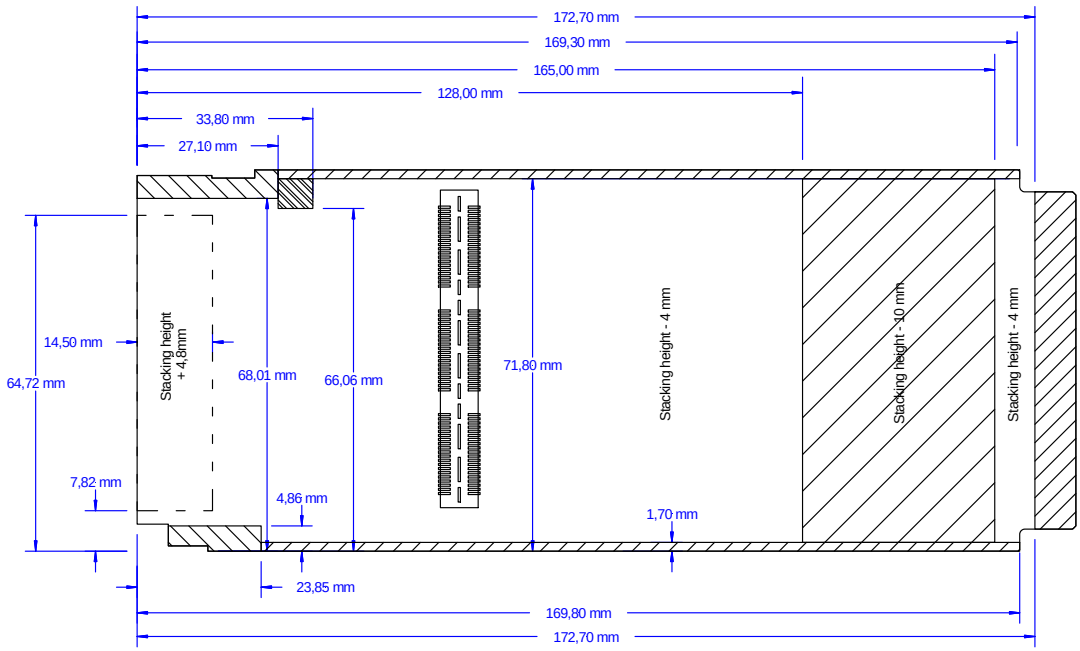

Figure 14-2: Maximum Component height for the SiCA

15 Module Management Controller (MMC)

The Module Management Controller of the TAMC900 handles the IPMI protocol, controls front panel LEDs and the payload power.

Four on board temperature sensors and the power supplies are observed by the MMC to ensure correct and secure operation of the TAMC900.

15.1 Indicators

For a quick visual inspection the TAMC900 offers 3 LEDs in the front panel and seven on board LEDs. For a detailed description of the on board LEDs, please refer to chapter “On Board Indicators”.

15.1.1 Front Panel LEDs

LED	Color	State	Description
HS	Blue	Off	No Power or module is powered
		Short Blink	Hot-Swap negotiation (extraction)
		Long Blink	Hot-Swap negotiation (insertion)
		On	Module is ready to be powered or module is ready to be unpowered
FAIL	Red	Off	No fault
		On	Failure or out of service status
USER	Green	Off	Board is unpowered
		On	Board is powered and OK
		Blink	controlled by on board FPGA

Table 15-1: Front Panel LEDs

15.2 Temperature and Voltage Sensors

The TAMC900 provides access to four temperature sensors via IPMI:

TEMP_ADC1	Temperature Sensor near ADC 0-3
TEMP_ADC2	Temperature Sensor near ADC 4-7
TEMP_RAM	Temperature Sensor at the QDR-II SRAM devices
TEMP_V5-CORE	On-DIE Temperature Sensor of the Virtex-5

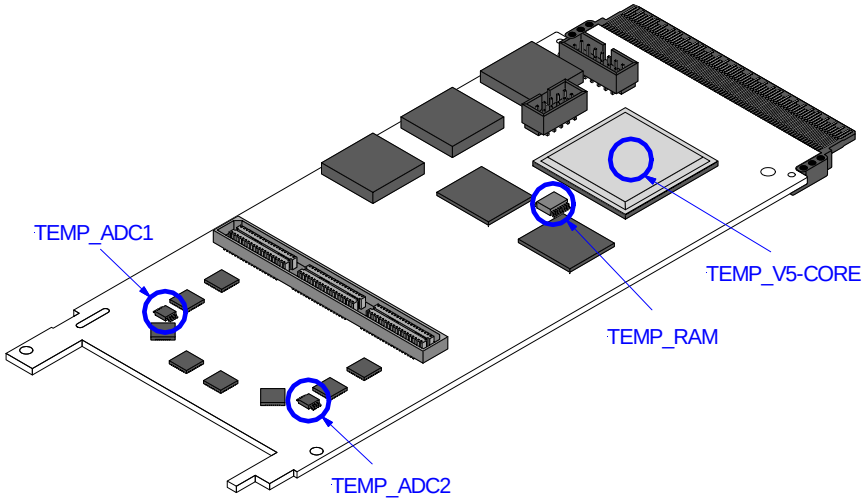

Figure 15-1: Temperature Sensor Locations

The TAMC900 provides access to two voltage sensors via IPMI:

VOLT_PAYLOAD	monitors the +12V Payload Power Supply
VOLT_SiCA	monitors the +6V Power Supply for the Signal Conditioning Adapter

15.3 Connectivity

The on board FPGA of the TAMC900 is connected to AMC Port 4 to 11.

AMC FCLKA (CLK3) is connected to the FPGA via a Jitter-Attenuator that scales the clock from 100 MHz up to 250 MHz and reduces the Clock Jitter.

15.4 Interfaces to Payload

The MMC has the following interfaces that allow interaction between the on board FPGA and the MMC.

Interface	Description
Payload Reset	The MMC controls the Payload RESET# signal. The RESET# signal stays low for app. 200ms after Payload Power is turned on and FPGA is configured.
FUNC_LED2	FUNC_LED2 is an Input to the MMC. A rising or falling edge of FUNC_LED2 triggers the MMC to flash the USER LED in the front panel (turn off for app. 100ms).
EKEY[4:1]	These signals can be used to transmit connectivity data from the MMC to the FPGA. This may be necessary in applications that implement different kinds of connections from the FPGA to the AMC interface. The implementation in MMC and FPGA has to be done by the customer.
I2C	This I2C Interface can be used by the FPGA to read connectivity data from the MMC. This may be necessary in applications that implement different kinds of connections from the FPGA to the AMC interface. The implementation in MMC and FPGA has to be done by the customer.
RXD0 / TXD0	These signals can be used to implement a serial communication between FPGA and MMC. By default, this is used as debug-output of the MMC. Any other implementation in MMC and FPGA has to be done by the customer.

16 On Board Indicators

For a quick visual inspection the TAMC900 offers 3 LEDs in the front panel. Please refer to chapter “Front Panel LEDs” for more information.

The TAMC900 provides additional on board LEDs that indicate Power Good of the local power supplies and successful FPGA configuration.

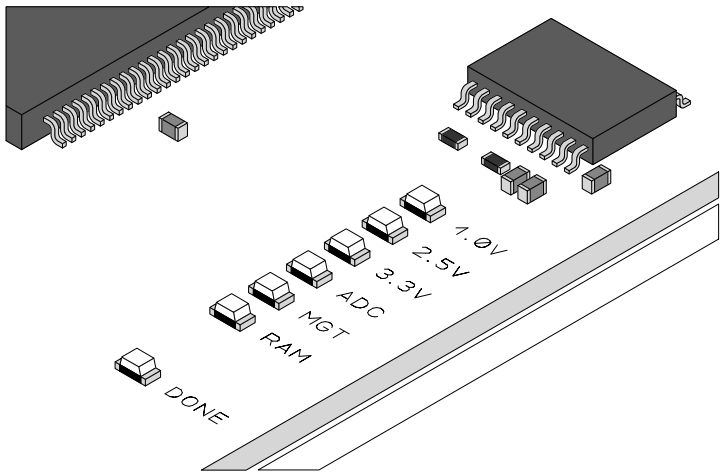

Figure 16-1: On Board Indicators

16.1 DONE LED

The green “DONE” LED is located on the bottom side of the PCB. It indicates that the FPGA is successful configured. If the LED is off, the FPGA is not configured.

16.2 Power Good LEDs

There are six green “Power Good” LEDs on the bottom side of the TAMC900. If one of the LEDs is off, this indicates a power failure of the corresponding power supply.

17 Pin Assignment

17.1 Overview

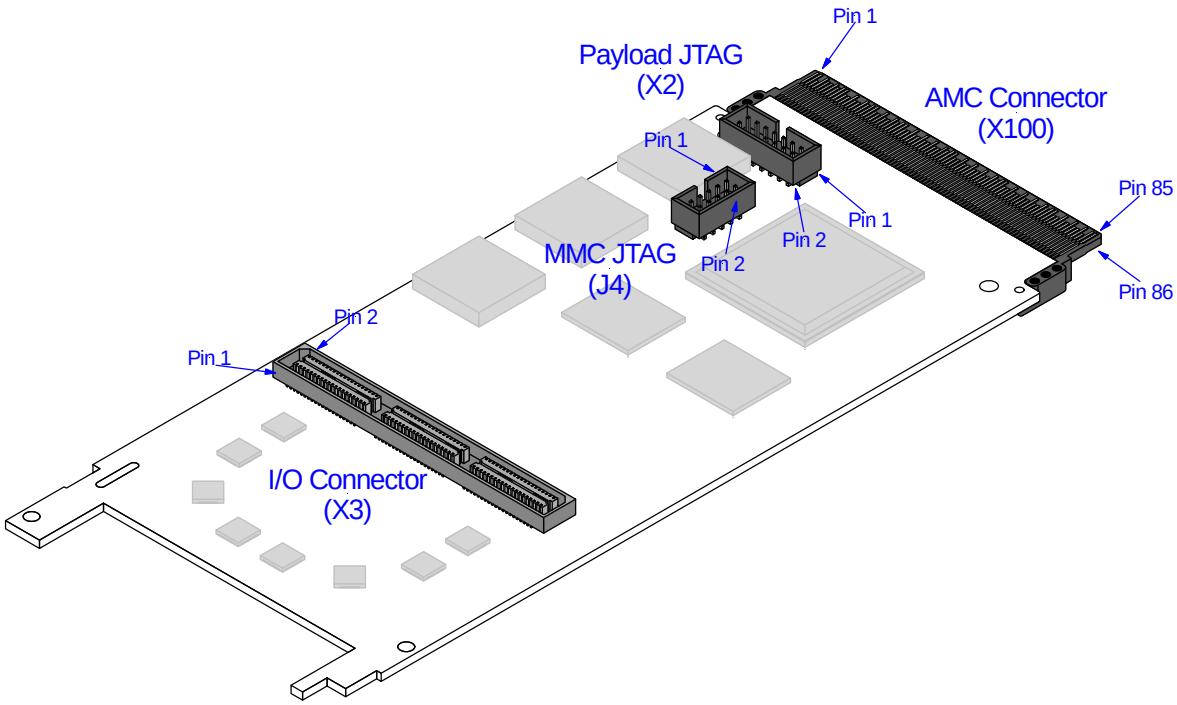

Figure 17-1: Connector Overview

The TAMC900 interfaces to a Signal Conditioning Adapter that carries the I/O connectors accessible through the front panel.

The TAMC900 has a 120 pin connector (Samtec QSE-060-01-L-D-A) that interfaces to the Signal Conditioning Adapter. See the figure below for pin locating of this connector.

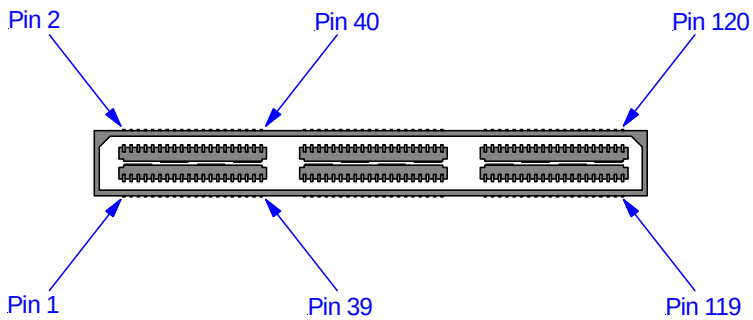

Figure 17-2: I/O Connector Pin Positions

17.2 I/O Connector

The I/O interface of the TAMC900 is the connector between SiCA and the TAMC900.

The Samtec QTE / QSE Series are used as I/O connection between TAMC900 and the SiCA. The TAMC900 carries a QSE connector, and the mating QTE connector is populated on the SiCA. The stacking height is defined by the QTE connector on the SiCA.

The Samtec QSE-060-01-L-D-A or compatible is used on the TAMC900.

Pin	Signal	Level
1	GP_IO_0	2.5 Volt CMOS
3	GP_IO_1	2.5 Volt CMOS
5	GP_IO_2	2.5 Volt CMOS
7	GP_IO_3	2.5 Volt CMOS
9	GP_IO_4	2.5 Volt CMOS
11	GP_IO_5	2.5 Volt CMOS
13	GP_IO_6	2.5 Volt CMOS
15	GP_IO_7	2.5 Volt CMOS
17	GP_IO_8	2.5 Volt CMOS
19	GP_IO_9	2.5 Volt CMOS
21	GND	logic Ground
23		
25		
27		
29		
31	GND	logic Ground
33	GND	logic Ground
35	AIN_0-	0 – 2.9 Volt
37	AIN_0+	
39	GND	logic Ground
41	GND	logic Ground
43	AIN_1-	0 – 2.9 Volt
45	AIN_1+	
47	GND	logic Ground
49	GND	logic Ground
51	AIN_2-	0 – 2.9 Volt
53	AIN_2+	
55	GND	logic Ground
57	GND	logic Ground

Pin	Signal	Level
2	SiCA_PWR	+6 Volt
4	SiCA_PWR	+6 Volt
6	SiCA_PWR	+6 Volt
8	SiCA_PWR	+6 Volt
10	SiCA_PWR	+6 Volt
12	SiCA_PWR	+6 Volt
14	GND	logic Ground
16	GND	logic Ground
18	LVDS_0+	LVDS
20	LVDS_0-	LVDS
22	GND	logic Ground
24	LVDS_1+	LVDS
26	LVDS_1-	LVDS
28	GND	logic Ground
30	LVDS_2+	LVDS
32	LVDS_2-	LVDS
34	GND	logic Ground
36	LVDS_3+	LVDS
38	LVDS_3-	LVDS
40	GND	logic Ground
42	GND	logic Ground
44	LVDS_4+	LVDS
46	LVDS_4-	LVDS
48	GND	logic Ground
50	LVDS_5+	LVDS
52	LVDS_5-	LVDS
54	GND	logic Ground
56	LVDS_6+	LVDS
58	LVDS_6-	LVDS

Pin	Signal	Level
59	AIN_3-	0 – 2.9 Volt
61	AIN_3+	
63	GND	logic Ground
65	GND	logic Ground
67	AIN_4-	0 – 2.9 Volt
69	AIN_4+	
71	GND	logic Ground
73	GND	logic Ground
75	AIN_5-	0 – 2.9 Volt
77	AIN_5+	
79	GND	logic Ground
81	GND	logic Ground
83	AIN_6-	0 – 2.9 Volt
85	AIN_6+	
87	GND	logic Ground
89	GND	logic Ground
91	AIN_7-	0 – 2.9 Volt
93	AIN_7+	
95	GND	logic Ground
97	GND	logic Ground
99		
101		
103		
105		
107	GND	logic Ground
109	GP_IO_10	2.5 Volt CMOS
111	GP_IO_11	2.5 Volt CMOS
113	GP_IO_12	2.5 Volt CMOS
115	GP_IO_13	2.5 Volt CMOS
117	GP_IO_14	2.5 Volt CMOS
119	GP_IO_15	2.5 Volt CMOS

Pin	Signal	Level
60	GND	logic Ground
62	LVDS_7+	LVDS
64	LVDS_7-	LVDS
66	GND	logic Ground
68	LVDS_8+	LVDS
70	LVDS_8-	LVDS
72	GND	logic Ground
74	LVDS_9+	LVDS
76	LVDS_9-	LVDS
78	GND	logic Ground
80	GND	logic Ground
82	GND	logic Ground
84	TRIG_0+	LVDS
86	TRIG_0-	LVDS
88	GND	logic Ground
90	TRIG_1+	LVDS
92	TRIG_1-	LVDS
94	GND	logic Ground
96	TRIG_2+	LVDS
98	TRIG_2-	LVDS
100	GND	logic Ground
102	GND	logic Ground
104	CLKIN_0+	LVDS
106	CLKIN_0-	LVDS
108	GND	logic Ground
110	CLKIN_1+	LVDS
112	CLKIN_1-	LVDS
114	GND	logic Ground
116	CLKIN_2+	LVDS
118	CLKIN_2-	LVDS
120	GND	logic Ground

Table 17-1: Pin Assignment I/O Connector

17.3 MMC JTAG Connector (Factory Use Only)

Pin	Signal	Level
1	TCK	3.3V CMOS / TTL
3	TDO	3.3V CMOS / TTL
5	TMS	3.3V CMOS / TTL
7	MP (out)	+3.3V
9	TDI	3.3V CMOS / TTL

Pin	Signal	Level
2	GND	Logic Ground
4	MP (out)	+3.3V
6	RST#	3.3V CMOS / TTL
8	Do Not Connect	-
10	GND	Logic Ground

Table 17-2: Pin Assignment MMC JTAG Connector (Factory use only)

17.4 Payload JTAG Connector

Pin	Signal	Level
1	GND	Logic Ground
3	GND	Logic Ground
5	GND	Logic Ground
7	GND	Logic Ground
9	GND	Logic Ground
11	GND	Logic Ground
13	GND	Logic Ground

Pin	Signal	Level
2	+3.3 V (out)	+3.3V
4	TMS	3.3V CMOS / TTL
6	TCK	3.3V CMOS / TTL
8	TDO	3.3V CMOS / TTL
10	TDI	3.3V CMOS / TTL
12	Do Not Connect	-
14	Do Not Connect	-

Table 17-3: Pin Assignment Payload JTAG Connector

17.5 AMC Connector

Signals written in *Italic* are not connected on the TAMC900

Pin	Signal	Function
85	GND	Logic Ground
84	PWR	+12V Payload Power
83	PS0#	Present detect
82	GND	Logic Ground
81	FCLKA-	Differential Signaling
80	FCLKA+	
79	GND	Logic Ground
78	*TCLKB-*	*Differential Signaling*
77	*TCLKB+*	
76	GND	Logic Ground
75	*TCLKA-*	*Differential Signaling*

Pin	Signal	Function
86	GND	Logic Ground
87	Rx8-	Differential Signaling
88	Rx8+	
89	GND	Logic Ground
90	Tx8-	Differential Signaling
91	Tx8+	
92	GND	Logic Ground
93	Rx9-	Differential Signaling
94	Rx9+	
95	GND	Logic Ground
96	Tx9-	Differential Signaling

Pin	Signal	Function
74	*TCLKA+*	
73	GND	Logic Ground
72	PWR	+12V Payload Power
71	SDA_L	IPMB-L Data
70	GND	Logic Ground
69	Rx7-	Differential Signaling
68	Rx7+	
67	GND	Logic Ground
66	Tx7-	Differential Signaling
65	Tx7+	
64	GND	Logic Ground
63	Rx6-	Differential Signaling
62	Rx6+	
61	GND	Logic Ground
60	Tx6-	Differential Signaling
59	Tx6+	
58	GND	Logic Ground
57	PWR	+12V Payload Power
56	SCL_L	IPMB-L Clock
55	GND	Logic Ground
54	Rx5-	Differential Signaling
53	Rx5+	
52	GND	Logic Ground
51	Tx5-	Differential Signaling
50	Tx5+	
49	GND	Logic Ground
48	Rx4-	Differential Signaling
47	Rx4+	
46	GND	Logic Ground
45	Tx4-	Differential Signaling
44	Tx4+	
43	GND	Logic Ground
42	PWR	+12V Payload Power
41	ENABLE#	AMC Enable Input
40	GND	Logic Ground
39	*Rx3-*	*Differential Signaling*
38	*Rx3+*	
37	GND	Logic Ground
36	*Tx3-*	*Differential Signaling*

Pin	Signal	Function
97	Tx9+	
98	GND	Logic Ground
99	Rx10-	Differential Signaling
100	Rx10+	
101	GND	Logic Ground
102	Tx10-	Differential Signaling
103	Tx10+	
104	GND	Logic Ground
105	Rx11-	Differential Signaling
106	Rx11+	
107	GND	Logic Ground
108	Tx11-	Differential Signaling
109	Tx11+	
110	GND	Logic Ground
111	*Rx12-*	*Differential Signaling*
112	*Rx12+*	
113	GND	Logic Ground
114	*Tx12-*	*Differential Signaling*
115	*Tx12+*	
116	GND	Logic Ground
117	*Rx13-*	*Differential Signaling*
118	*Rx13+*	
119	GND	Logic Ground
120	*Tx13-*	*Differential Signaling*
121	*Tx13+*	
122	GND	Logic Ground
123	*Rx14-*	*Differential Signaling*
124	*Rx14+*	
125	GND	Logic Ground
126	*Tx14-*	*Differential Signaling*
127	*Tx14+*	
128	GND	Logic Ground
129	*Rx15-*	*Differential Signaling*
130	*Rx15+*	
131	GND	Logic Ground
132	*Tx15-*	*Differential Signaling*
133	*Tx15+*	
134	GND	Logic Ground
135	*TCLKC-*	*Differential Signaling*

Pin	Signal	Function
35	*Tx3+*	
34	GND	Logic Ground
33	*Rx2-*	*Differential Signaling*
32	*Rx2+*	
31	GND	Logic Ground
30	*Tx2-*	*Differential Signaling*
29	*Tx2+*	
28	GND	Logic Ground
27	PWR	+12V Payload Power
26	GA2	Geogr. Address Input
25	GND	Logic Ground
24	*Rx1-*	*Differential Signaling*
23	*Rx1+*	
22	GND	Logic Ground
21	*Tx1-*	*Differential Signaling*
20	*Tx1+*	
19	GND	Logic Ground
18	PWR	+12V Payload Power
17	GA1	Geogr. Address Input
16	GND	Logic Ground
15	*Rx0-*	*Differential Signaling*
14	*Rx0+*	
13	GND	Logic Ground
12	*Tx0-*	*Differential Signaling*
11	*Tx0+*	
10	GND	Logic Ground
9	PWR	12V Payload Power
8	*RSRVD8*	-
7	GND	Logic Ground
6	*RSRVD6*	-
5	GA0	Geogr. Address Input
4	MP	+3.3V
3	PS1#	Present detect
2	PWR	+12V Payload Power
1	GND	Logic Ground

Pin	Signal	Function
136	*TCLKC+*	
137	GND	Logic Ground
138	*TCLKD-*	*Differential Signaling*
139	*TCLKD+*	
140	GND	Logic Ground
141	*Rx17-*	*Differential Signaling*
142	*Rx17+*	
143	GND	Logic Ground
144	*Tx17-*	*Differential Signaling*
145	*Tx17+*	
146	GND	Logic Ground
147	*Rx18-*	*Differential Signaling*
148	*Rx18+*	
149	GND	Logic Ground
150	*Tx18-*	*Differential Signaling*
151	*Tx18+*	
152	GND	Logic Ground
153	*Rx19-*	*Differential Signaling*
154	*Rx19+*	
155	GND	Logic Ground
156	*Tx19-*	*Differential Signaling*
157	*Tx19+*	
158	GND	Logic Ground
159	*Rx20-*	*Differential Signaling*
160	*Rx20+*	
161	GND	Logic Ground
162	*Tx20-*	*Differential Signaling*
163	*Tx20+*	
164	GND	Logic Ground
165	TCK	JTAG Clock Input
166	TMS	JTAG TMS Input
167	*TRST#*	*JTAG Reset Input*
168	TDO	JTAG TDO Output
169	TDI	JTAG TDI Input
170	GND	Logic Ground

Table 17-4: Pin Assignment AMC Connector