struck innovative
systeme

SIS8300-L
µTCA FOR PHYSICS Digitizer

User Manual

SIS GmbH
Harksheider Str. 102A
22399 Hamburg
Germany

Phone: ++49 (0) 40 60 87 305 0
Fax: ++49 (0) 40 60 87 305 20

email: info@struck.de
http://www.struck.de

Version: SIS8300L-M-1402-1-V100.doc as of 13.02.2014

Revision Table:

| Revision	Date	Modification
0.01	11.02.2013	Based on SIS8300-M-1402-1-V104 Firmware: V1402
0.02	11.03.2013	TH: register update - Firmware: V1402 - RTM LVDS Test Input/Output Control register - DAC Control register
0.03	01.07.2013	AG: DAC clock scheme and MGT clock scheme added
0.04	20.08.2013	AG: Clock distribution scheme update
0.05	21.11.2013	CT: Added Firmware upgrade description AG: SIS8300-L picture update AG: AVR and Xilinx JTAG Connector Picture added
1.00	13.02.2014	First official release, bug fixes and DDR3 memory diagram

- **Table of contents**

1   Introduction

The SIS8300-L is a ten 10 channel 125 MS/s digitizer with 16-bit resolution according to the MTCA.4 standard.
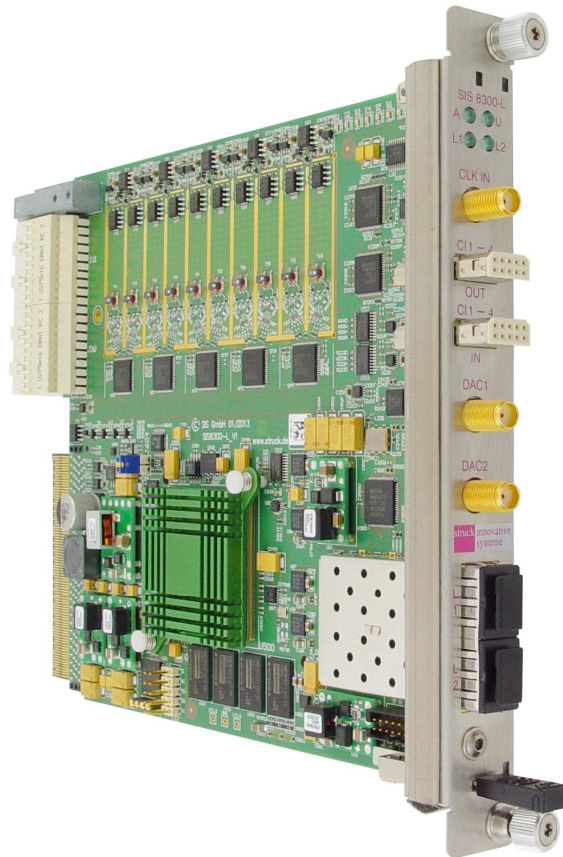
SIS8300-L

Note: While the SIS8300-L is Virtex 6 based you will find many Virtex 5 references in the firm- and software for historical reasons (i.e. remnants from the SIS8300 and SIS8300 V2 designs).

As we are aware, that no manual is perfect, we appreciate your feedback and will incorporate proposed changes and corrections as quickly as possible. The most recent version of this manual can be obtained by email from info@struck.de, the revision dates are online under http://www.struck.de/manuals.html .

1.1   Related documents

A list of available firmware designs can be retrieved from http://www.struck.de/SIS8300firm.html

ATTENTION
Observe Precautions
for Handling
Electrostatic Sensitive
Devices

2   Design

The central building block of the SIS8300-L card is a Xilinx Virtex 6 FPGA. It holds the 4 lane PCI Express interface and is in control of all active components.

2.1   Functionality

The key properties of the SIS8300-L card are listed below.

- AMC .4 µTCA for Physics Board
- 4 Lane PCI Express Interface
- Dual SFP Card Cage for optional Multi Gigabit Link
- Xilinx Virtex 6 FPGA
- DDR3 Memory Interface
- 4 x 4Gbit default DDR3 memory
- Atmega128 IPMI
- External Clock and Trigger Inputs
- Frontpanel digital I/O (4in/4 out) on Harlink Connectors
- RTM ADC Analog Inputs, I2C-Bus, DAC Analog Outputs
- 10 ADC Channels 125MS/s, 16-Bit
- 2 DAC Channels 250MS/s, 16-Bit
- Clock distribution with phase shifting
- 4 M-LVDS µTCA Ports
- 2 µTCA Clocks

2.2   Block Diagram

A simplified block diagram of the SIS8300-L is shown below.
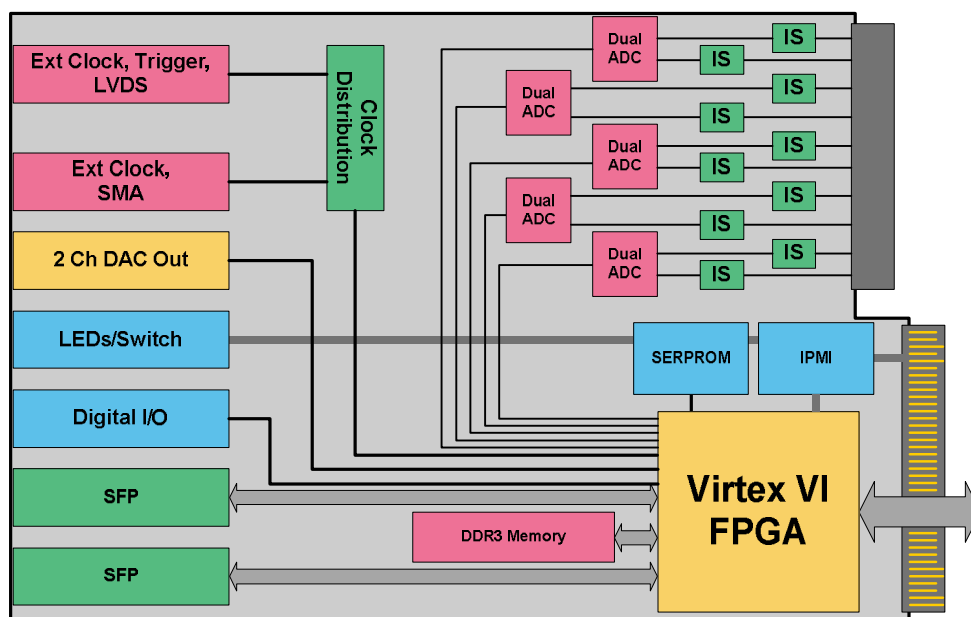
2.3   Platform Management

The management code of the SIS8300-L is implemented in an Atmel Atmega1281-16MU microcontroller and can be upgrade in field over connector J32 (see section 3.3).

2.4   Clock Distribution

The clock distribution scheme of the SIS8300-L is illustrated below.
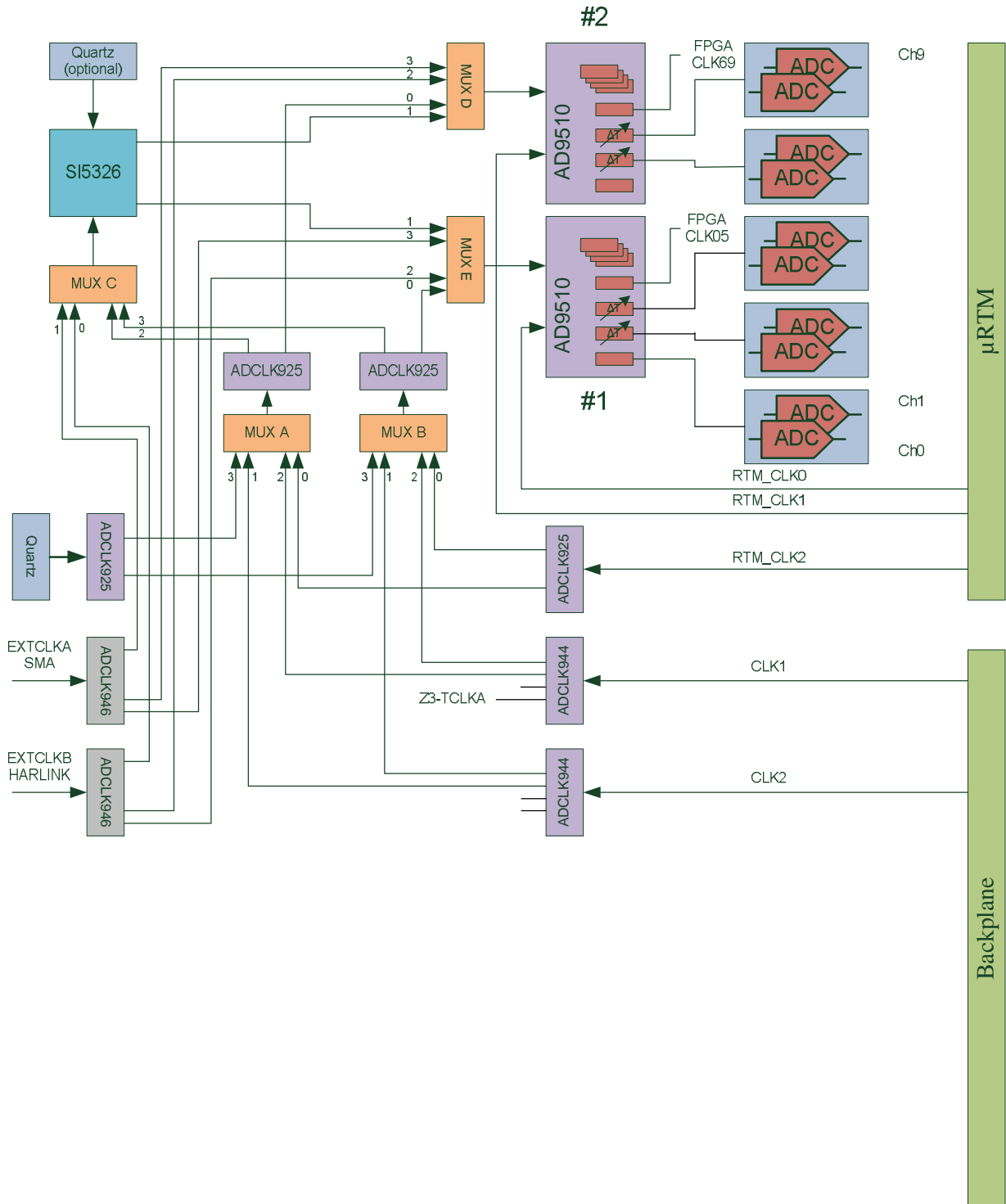
2.5 DAC Clock

The DAC clock scheme of the SIS8300-L is illustrated below.
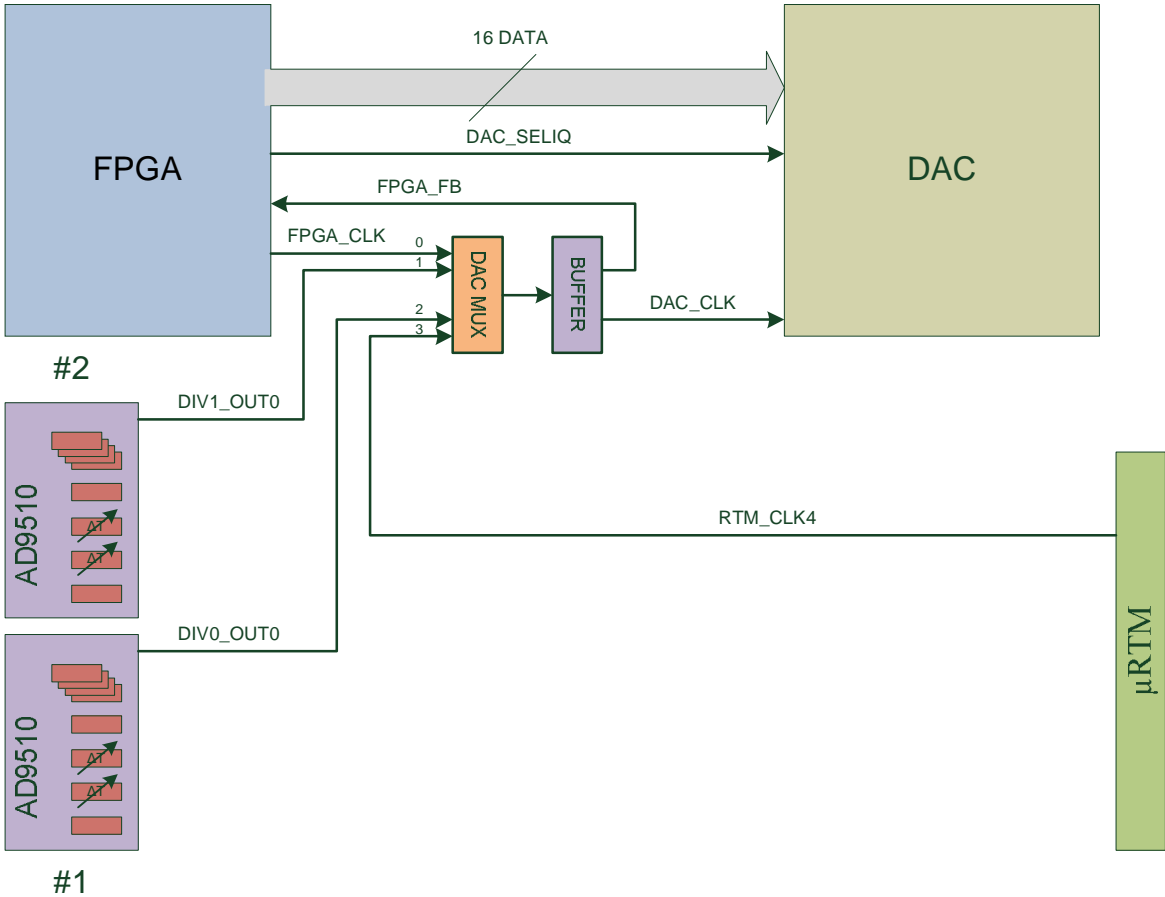
2.6 MGT Clock

The MGT clock scheme of the SIS8300-L is illustrated below.
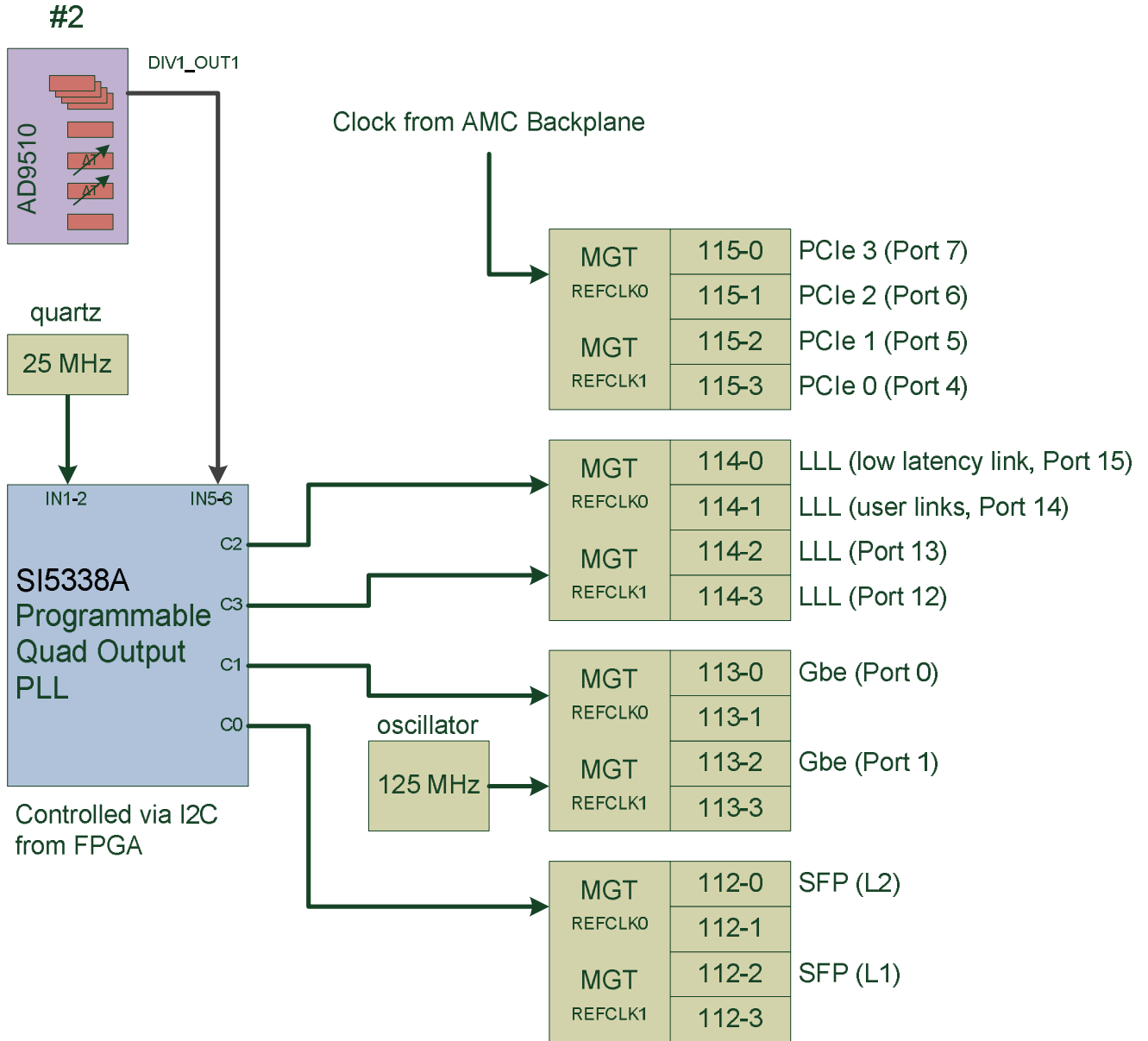
3   Jumper/Connector Pin Assignments

The following subsections describe the pin assignments of jumpers and connectors.

3.1   CON100 JTAG

The SIS8300-L's on board logic can load its firmware from a serial PROM, via the JTAG port on connector CON100, PCI Express or via the MMC.

Hardware like the XILINX HW-USB-JTAG in connection with the appropriate software will be required for in field JTAG firmware upgrades.

CON100 is a 2mm (i.e. metric) 14 pin header that allows you to reprogram the firmware of the SIS8300-L board with a JTAG programmer. The pin out is shown in the schematic below. It is compatible with the cable that comes with the XILINX HW-USB-II-G-JTAG platform cable. CON100 can be found at the right bottom side of the board.
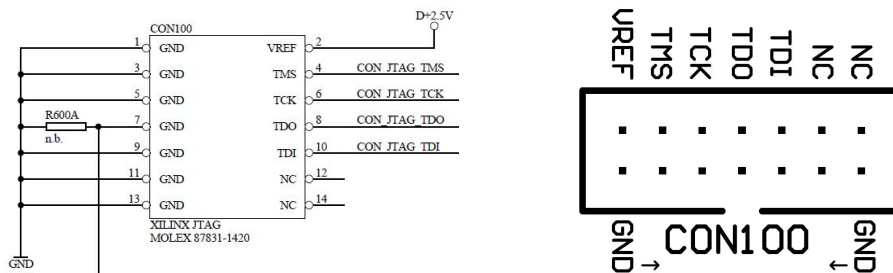
Note: The board has to be powered for reprogramming over JTAG

3.2   J604 Watchdog Reset

J604 can be found next to the left upper edge of U500 (largest chip on the card). With J604 closed the boards watchdog reset is connected to the reset logic. J604 should be opened for JTAG firmware programming.

3.3   J32 AVR JTAG

This 10-pin header is used to connect to the JTAG of the Atmel Atmega128 microcontroller providing the IPMI/MCH functionality of the SIS8300-L.
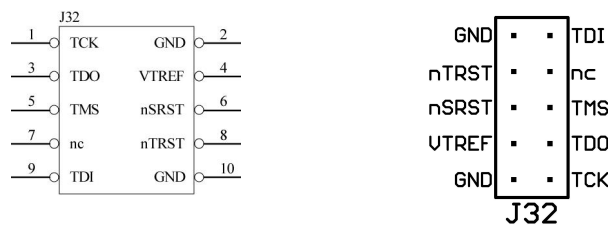
4 LEDs

4.1 AMC LEDs

The AMC LEDs are implemented according to the standard.

4.2 Front Panel LEDs

The SIS8300-L in Gigalink stuffing option has 4 green front panel LEDs.

LED name	Function in Gigalink design
A	PCI Express Access
U	User LED
L1	PCIe Link up
L2	ADC Sampling active
R (L1)	SFP Link 1 receiver loss of signal
T (L1)	SFP Link 1 transmitter fault
R (L2)	SFP Link 2 receiver loss of signal
T (L2)	SFP Link 2 transmitter fault

4.3 SMD LEDs

A number of surface mount red LEDs are on the SIS8300-L to visualize part of the board status.

LED designator	LED comment	Function
D20A	S1	Firmware dependent (Optical Link 1 up)
D20B	S2	Firmware dependent (Optical Link 2 up)
D20C	S3	Firmware dependent
D20D	S4	Firmware dependent
D20E	S5	Firmware dependent
D20F	S6	Firmware dependent
D20G	S7	Firmware dependent
D20H	S8	Firmware dependent
D21D	READY	FPGA ready

5   Front panel

The SIS8300-L is a µTCA for Physics board. A sketch of the front panel is shown below.
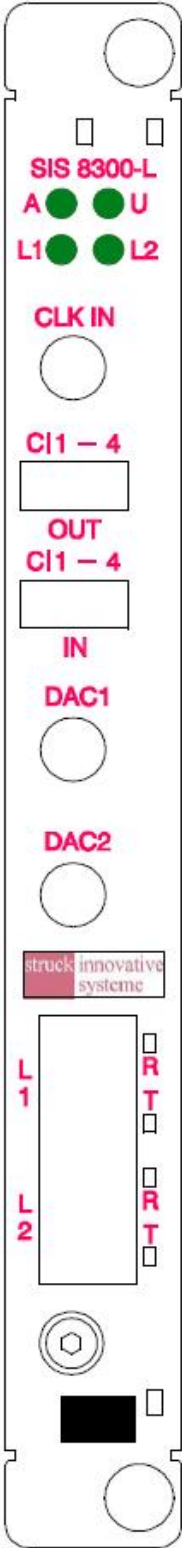
5.1 Harlink LVDS In-/Outputs

The Harlink LVDS Output and Input connectors have 5 signals each. The Clock signal to the left hand side is marked with C and the other 4 signals are labelled with 1-4.

	Clock	1	2	3	4
Top	P	P	P	P	P
Bottom	N	N	N	N	N

5.2 SMA Clock Input

The front panel SMA clock input is designed to accept a maximum peak to peak signal level of 3V into 50 Ohms. The clock input signal is coupled to the internal logic via a capacitor. The schematic of the input stage is shown below.
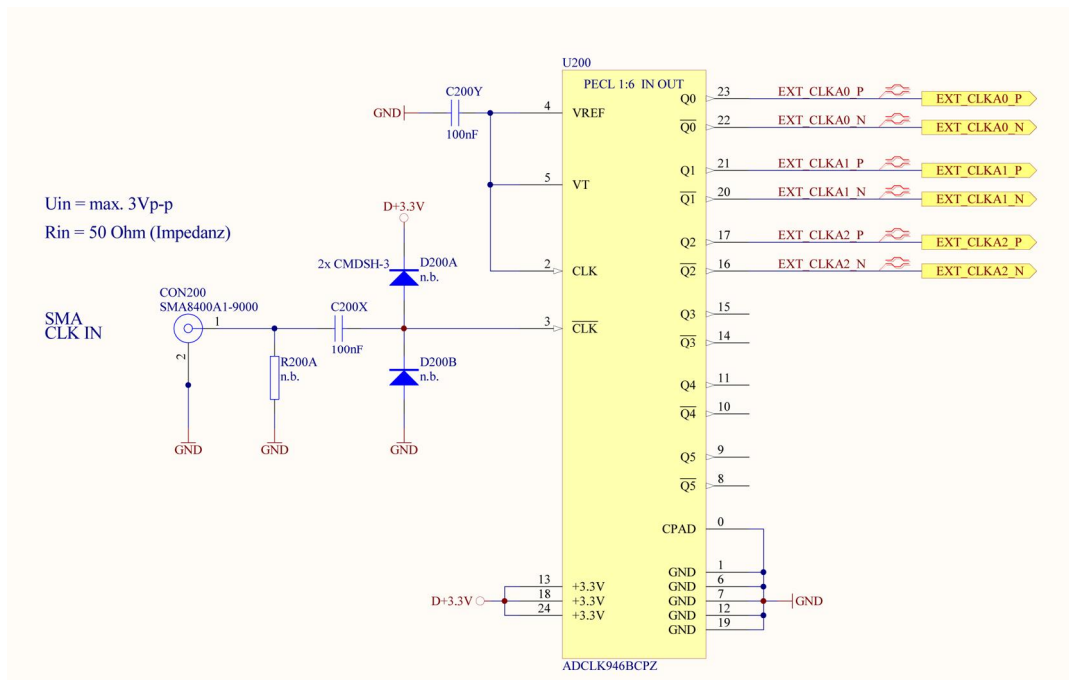
5.3 SFP Card Cage

The dual card cage can host two SFP link media.
They can be enabled or disabled in the SIS8300top.vhd VHDL code as shown below (and are active in the 0x1002 firmware design e.g.):

```
DUAL_OPTICAL_INTERFACE_EN      : integer := 1 ; --
```

Communication is handled through registers 0x14 to 0x17 (refer to the VHDL code)

6 Board Layout

A print of the silk screen of the component side is shown below.
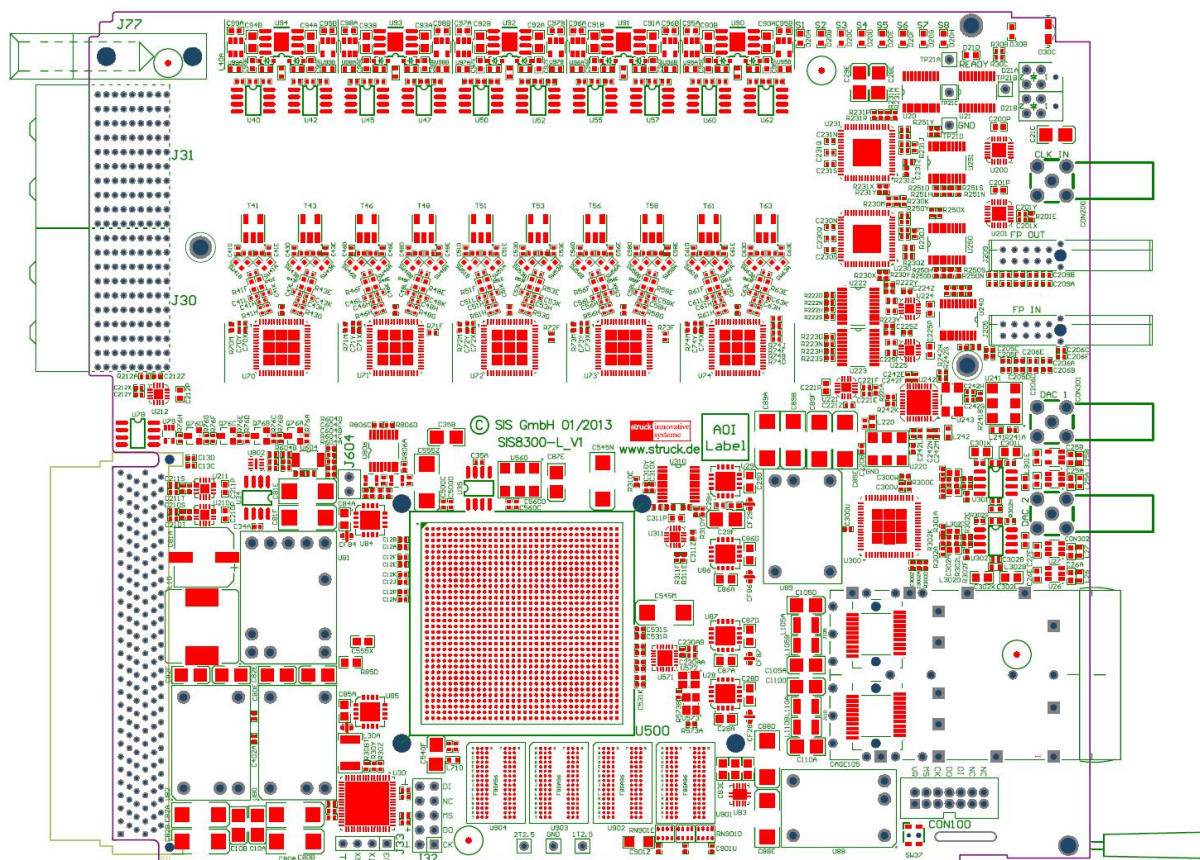
Connector types
The used connectors are listed in the table below.

Designator	Function	Manufacturer	Part Number
CON100	JTAG	Molex	87831-1420
CON200	Clock In	JYEBAO	SMA8400A1-9000
CON301	DAC 1 Out	JYEBAO	SMA8400A1-9000
CON302	DAC 2 Out	JYEBAO	SMA8400A1-9000
CAGE105	SFP Cage, 2 Ports	TYCO	1761014-1
J10	AdvancedMC	HARTING	16211701303000
J32	JTAG Atmega	SAMTEC	HTSW-105-26-G-D
J30	RTM	ERNI	ERmetZD-10x3P-FEM
J31	RTM	ERNI	ERmetZD-10x3P-FEM
J77	RTM Keying	TYCO	(*)
J209	Trigger & Clock Out	HARTING	27 21 121 8000
J205	Trigger & Clock In	HARTING	27 21 121 8000
J604	Watchdog	SAMTEC	HTSW-102-26-G-S

Note (*)**:** The used Key may depend on the hardware configuration of the SIS8300-L

7 Firmware Description

7.1 ADC Sample Logic

The block diagram shows the ADC data handling. Each ADC channel has its own Memory Address Control Logic.
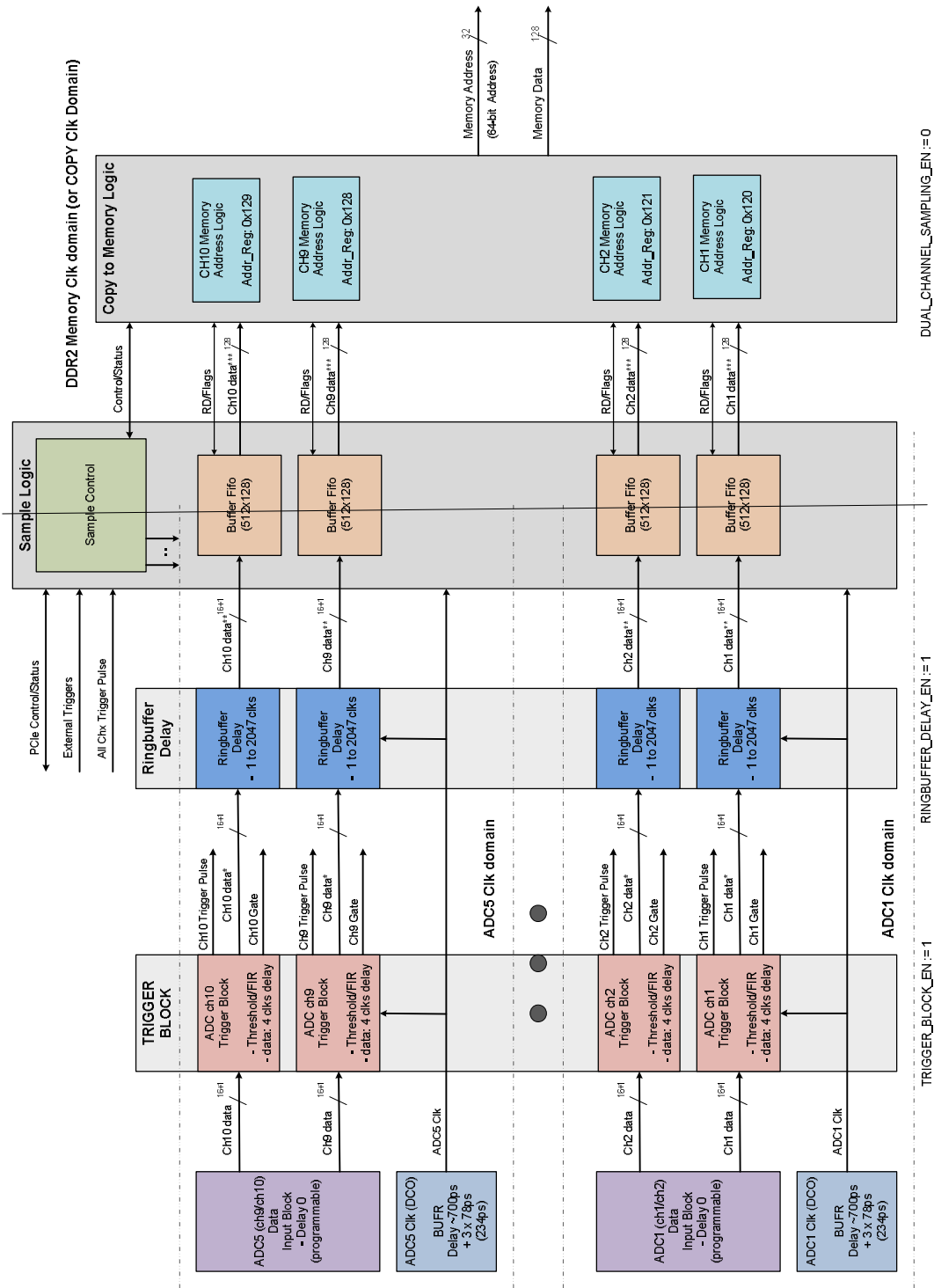
The block diagram shows the ADC data handling. Two ADC channels share the same Memory Address Control Logic.

The "Single-Channel Copy-To-Memory" Logic needs more FPGA resources (+2500 Registers, +700 Slices, +10 BlockRams) than the "Dual-Channel Copy-To-Memory" Logic.
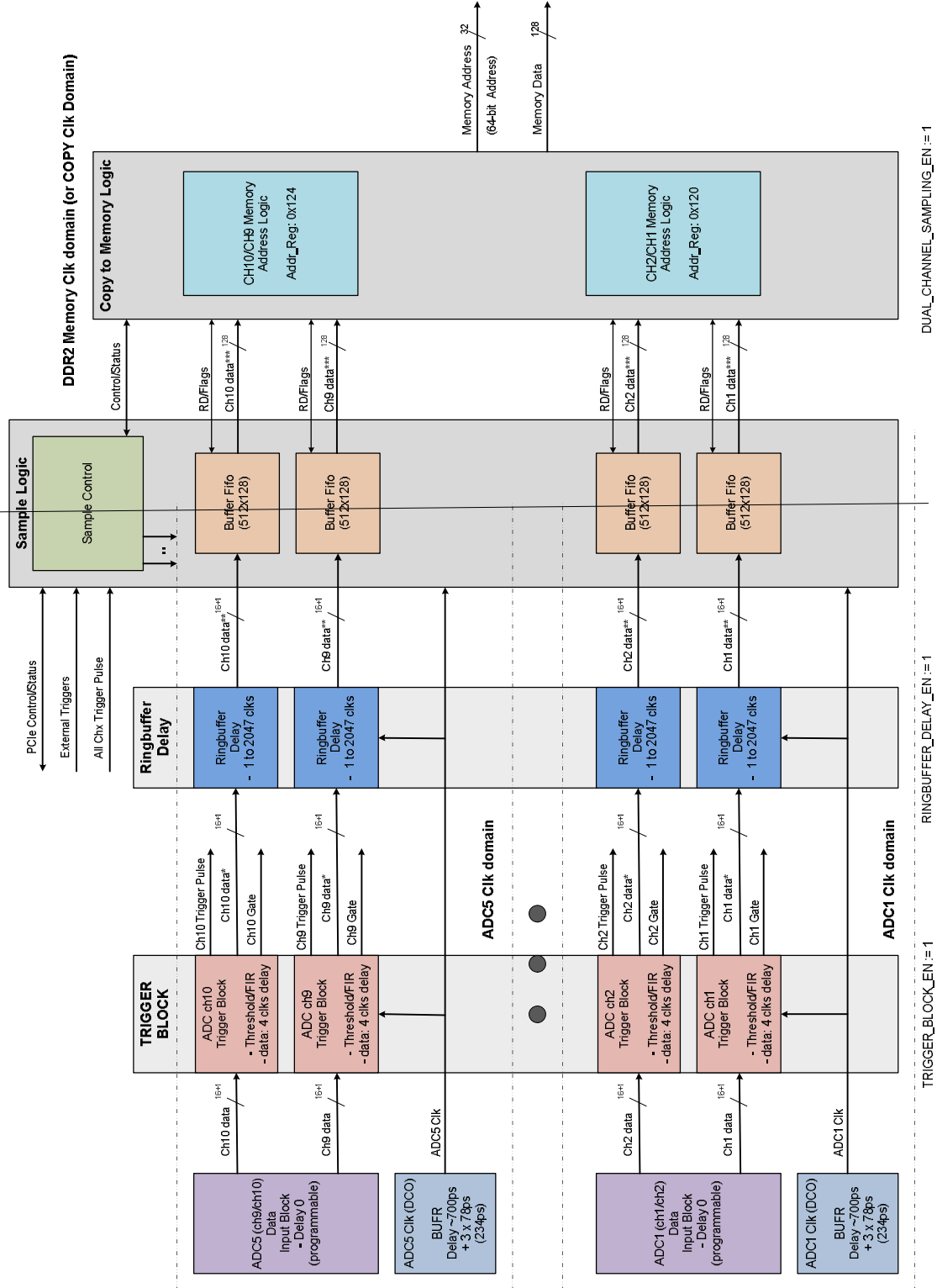
7.2 Memory Interface

A block diagram of the DDR3 memory controller is shown below.
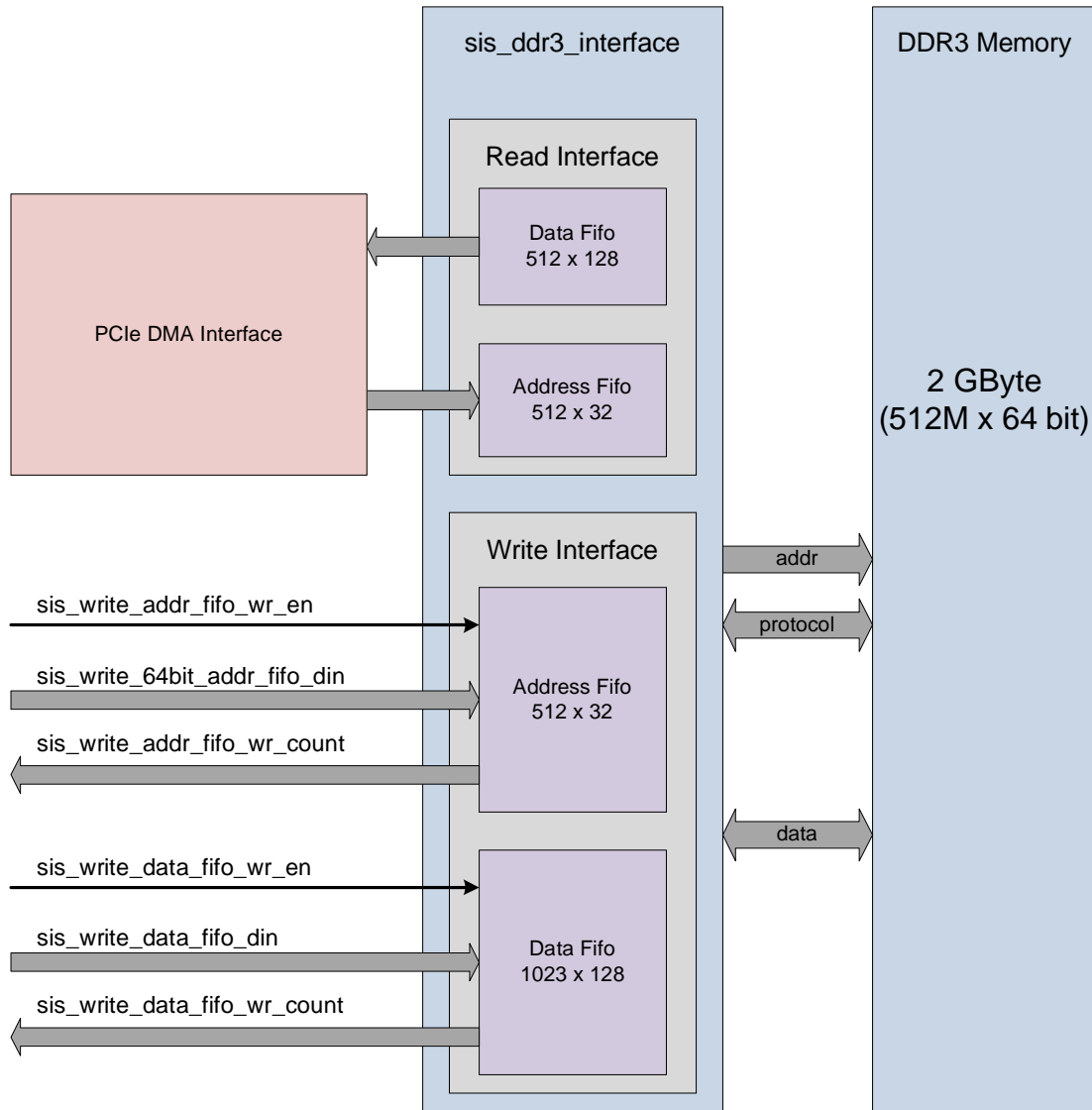
User Interface block diagram for Memory read, write and histogram operations.

7.2.1 Memory Write Interface

The Write Interface consists of the following signals:

```
   sis_write_fifo_wr_clk : in std_logic;
-- data: write fifo
   sis_write_data_fifo_wr_en : in std_logic;
   sis_write_data_fifo_din : in  std_logic_vector(127  downto 0);
   sis_write_data_fifo_wr_count : out  std_logic_vector(9  downto 0);

-- address: write fifo
   sis_write_addr_fifo_wr_en : in std_logic;
   sis_write_64bit_addr_fifo_din : in  std_logic_vector(31  downto 0);
   sis_write_addr_fifo_wr_count : out  std_logic_vector(9  downto 0)  ;
```

A write cycle to the memory consists of one write command to the Address Fifo and two write commands to the Data Fifo.

One write command to the Address FIFO:
a valid "sis_write_addr_fifo_wr_en" signal over one clock period (sis_write_fifo_wr_clk) along with "sis_write_64bit_addr_fifo_din".

Two write commands to the Data FIFO:
a valid "sis_write_data_fifo_wr_en" signal over two clock periods (sis_write_fifo_wr_clk) along with "sis_write_data_fifo_din".

When issuing a write command to the Address Fifo, the second write command to the Data Fifo must be issued no more than one clock cycle later.

It is only allowed to write to the Address-FIFO, if "sis_write_addr_fifo_wr_count" is lower than X"1FF" (not full).
It is only allowed to write to the Data-FIFO, if "sis_write_data_fifo_wr_count" is lower than X"3FE" (not full).

The Memory Controller writes with one "write cycle" 256 bits (4 x 64 bits) to the Memory. Therefore the lower 2 address bits of the written 64-bit address must be 0 and the "next address" will be incremented by 4.

7.2.2 Memory Histogram Interface

The Histogram Interface consists of the following signals:

```
-- Histogramming
-- address: write fifo
   sis_write_histo_addr_fifo_clr: in std_logic;
   sis_write_histo_addr_fifo_wr_clk: in std_logic;
   sis_write_histo_addr_fifo_wr_en: in std_logic;
   sis_write_histo_32bit_addr_fifo_din: in  std_logic_vector(31  downto 0);
   sis_write_histo_addr_fifo_wr_count: out  std_logic_vector(9  downto 0) ;
```

A write command to the Histogram Address Fifo will increment by one the content of the 32-bit Memory value addressed with the written 32-bit Memory Address.

It is only allowed to write to the Histo-Address-FIFO, if
"sis_write_histo_addr_fifo_wr_count" is lower than X"1FF" (not full).

The histogramming memory controller supports an update rate of 5MHz (20 MHz within one 2K memory page amid differing three lowest order bits).

7.3 Memory buffer

The structure of the memory buffer with **512 MByte** (i.e. 4 x 1 GBit memory chips) is illustrated below.

512 MByte : 4 x 64M x 16bit = 256M x 16bit = 32M x 128bit = 16M x 256bit

Memory 256-bit Block Addresses	Memory 64-bit Addresses (4 x 256-bit Address)	Memory 32-bit Addresses (8 x 256-bit Address)	Memory 16-bit Addresses (16 x 256-bit Address)	Memory 8-bit (byte) Addresses (32 x 256-bit Address)
0 x 00 0000	0 x 000 0000	0 x 000 0000	0 x 0000 0000	0 x 0000 0000
0 x 00 0001	0 x 000 0001	0 x 000 0001	0 x 0000 0001	0 x 0000 0001
0 x 00 0002	0 x 000 0002	..	..	..
..	0 x 000 0003	0 x 000 0006	0 x 0000 000E	0 x 0000 001E
0 x 7F FFFF	..	0 x 000 0007	0 x 0000 000F	0 x 0000 001F
0 x 80 0000	0 x 200 0000	..	..	..
0 x 80 0001	0 x 200 0001	0 x 400 0002	0 x 0800 0006	0 x 1000 000E
0 x 80 0002	0 x 200 0002	0 x 400 0003	0 x 0800 0007	0 x 1000 000F
	0 x 200 0003	..	..	..
..	..	0 x 7FF FFF8	0 x 0FFF FFF0	0 x 1FFF FFE0
	0 x 3FF FFFC	0 x 7FF FFF9	0 x 0FFF FFF1	0 x 1FFF FFE1
	0 x 3FF FFFD	..	..	..
	0 x 3FF FFFE	0 x 7FF FFFE	0 x 0FFF FFFE	0 x 1FFF FFFE
0 x FF FFFF	0 x 3FF FFFF	0 x 7FF FFFF	0 x 0FFF FFFF	0 x 1FFF FFFF

The structure of the memory buffer with **1 GByte** (i.e. 4 x 2 GBit memory chips) is illustrated below.

1 GByte : 4 x 128M x 16bit = 512M x 16bit = 64M x 128bit = 32M x 256bit

Memory 256-bit Block Addresses	Memory 64-bit Addresses (4 x 256-bit Address)	Memory 32-bit Addresses (8 x 256-bit Address)	Memory 16-bit Addresses (16 x 256-bit Address)	Memory 8-bit (byte) Addresses (32 x 256-bit Address)
0 x 00 0000	0 x 000 0000	0 x 000 0000	0 x 0000 0000	0 x 0000 0000
0 x 00 0001	0 x 000 0001	0 x 000 0001	0 x 0000 0001	0 x 0000 0001
0 x 00 0002	0 x 000 0002	..	..	..
..	0 x 000 0003	0 x 000 0006	0 x 0000 000E	0 x 0000 001E
0 x FF FFFF	..	0 x 000 0007	0 x 0000 000F	0 x 0000 001F
0 x 100 0000	0 x 400 0000	..	..	..
0 x 100 0001	0 x 400 0001	0 x 800 0002	0 x 1000 0006	0 x 2000 000E
0 x 100 0002	0 x 400 0002	0 x 800 0003	0 x 1000 0007	0 x 2000 000F
	0 x 400 0003	..	..	..
..	..	0 x FFF FFF8	0 x 1FFF FFF0	0 x 3FFF FFE0
	0 x 7FF FFFC	0 x FFF FFF9	0 x 1FFF FFF1	0 x 3FFF FFE1
	0 x 7FF FFFD	..	..	..
	0 x 7FF FFFE	0 x FFF FFFE	0 x 1FFF FFFE	0 x 3FFF FFFE
0 x 1FF FFFF	0 x 7FF FFFF	0 x FFF FFFF	0 x 1FFF FFFF	0 x 3FFF FFFF

7.4   Address Map

Following 32-bit addresses are implemented

Offset	Access	Function
0x00	R	Module Identifier/Firmware Version register
0x01	R	Serial number register
0x02	R/W	XILINX JTAG register
0x03	R	XILINX Virtex 6 configuration memory Error Detection register
0x04	R/W	User Control/Status register (JK)
0x05	R	Firmware Options register
0x10	R/W	ADC Acquisition Control/Status register
0x11	R/W	ADC Sample Control register
0x12	R/W	MLVDS Input/Output Control register
0x13	R/W	Harlink Connector Input/Output Control register
0x14	R/W	Link 1 data FIFO (refer to VHDL code)
0x15	R/W	Link 1 set control (refer to VHDL code)
0x16	R/W	Link 2 data FIFO (refer to VHDL code)
0x17	R/W	Link 2 set control (refer to VHDL code)
0x40	R/W	Clock Distribution Multiplexer control register
0x41	R/W	Clock Distribution IC AD9510 SPI interface register
0x42	R/W	Clock Multiplier IC SI5326 SPI interface register
0x43	R/W	Clock Synthesizer Si5338A for MGT clocks
0x44	R/W	FPGA Boot SPI Flash interface
0x45	R/W	DAC Control register
0x46	R/W	DAC Data register
0x48	R/W	ADC SPI Interface register
0x49	R/W	ADC Input Tap delay register
0xFF	W	Bit 0 = 1: Master Reset (reset all registers)
0x100	R/W	ADC ch1 Trigger Setup register
0x101	R/W	ADC ch2 Trigger Setup register
..		..
..		..
0x109	R/W	ADC ch10 Trigger Setup register
0x110	R/W	ADC ch1 Trigger Threshold register
0x111	R/W	ADC ch2 Trigger Threshold register
..		..
0x119	R/W	ADC ch10 Trigger Threshold register
0x120	R/W	ADC ch1 Memory Sample Start Address / Actual Address register

0x121	R/W	ADC ch2 Memory Sample Start Address / Actual Address register
..		..
0x129	R/W	ADC ch10 Memory Sample Start Address / Actual Address register
0x12A	R/W	ADC chx Sample Length register
0x12B	R/W	ADC chx Ringbuffer Delay register (0 to 2046)
0x12C	R/W	Test Histogram Pattern-Memory Address register
0x12D	R/W	Test Histogram Pattern-Memory Data Write register
0x12E	R/W	Test Histogram Control register
0x12F	R/W	SIS8900 RTM LVDS Test Input/Output Control register
0x200	R/W	DMA_READ_DST_ADR_LO32
0x201	R/W	DMA_READ_DST_ADR_HI32
0x202	R/W	DMA_READ_SRC_ADR_LO32
0x203	R/W	DMA_READ_LEN
0x204	R/W	DMA_READ_CTRL
0x205	R/W	DMA Readout Sample byte swap control
0x210	R/W	DMA_WRITE_SRC_ADR_LO32
0x211	R/W	DMA_WRITE_SRC_ADR_HI32
0x212	R/W	DMA_WRITE_DST_ADR_LO32
0x213	R/W	DMA_WRITE_LEN
0x214	R/W	DMA_WRITE_CTRL
0x216	R/W	DAQ Auto DMA Chain Control
0x220	R/W	IRQ Enable
0x221	R	IRQ Status
0x222	W	IRQ Clear
0x223	KA	IRQ Refresh
0x230	R/W	MEMORY test Mode register
0x231	R/W	RAM FIFO debug register
0x400	R/W	Mapped out of register bank to top level. May be used for user defined register implementation. See Section 7.6.
...		
...		
0x4FF	R/W	

7.5   Register description

7.5.1   Module Id. and Firmware Revision register
#define SIS8300-L_IDENTIFIER_VERSION_REG   0x00

This register holds the module identifier (SIS8301) and the firmware version and revision.

BIT	access	Name	Function
31-16 FFFF0000	RO	Module Identifier	0x8301
15-8 0000FF00	RO	Firmware Version	1..255
7-0 000000FF	RO	Firmware Revision	1..255

Example: The initial version of the SIS8300-L reads 0x83011002

7.5.2   Serial Number register
#define SIS8300-L_SERIAL_NUMBER_REG   0x01

This register holds the Serial Number of the module.

BIT	access	Name	Function
31-16 FFFF0000	RO	reserved	
15-0 0000FFFF	RO	Serial Number	1..65535

7.5.3 XILINX JTAG register

#define SIS8300-L_XILINX_JTAG_REG   0x02

This register is used in the firmware upgrade process over PCIe only. A TCK is generated upon a write cycle to this register.

Bit	write Function	read Function
31	none	TDO
30	...	1 x Shifted TDO
...	...	
4	none	
3	none	
2	none	
1	TMS	
0	TDI	30 x Shifted TDO

The read register function operates as a shift register for TDO. The content of the read register is shifted to the right by one bit with every positive edge of TCK and the status of TDO is transferred to Bit 30. Bit 31 reflects the current value of TDO during a read access.

7.5.4 XILINX Virtex 6 Error Detection register

#define SIS8300-L_XILINX_ECC_REG   0x03

XILINX Virtex 6 configuration memory error detection register.

Bit	read Function
31	0
30	0
...	
16	0
15	Frame ECC output indicating a valid SYNDROME value
14	0
13	CRC Error
12	ECC Error
11	SYNDROME: Status bit 11
..	
1	SYNDROME: Status bit 1
0	SYNDROME: Status bit 0

7.5.5 User Control/Status register
#define SIS8300-L_USER_CONTROL_STATUS_REG   0x04

The control register is implemented as a selective J/K register, a specific function is enabled by writing a 1 into the set/enable bit, the function is disabled by writing a 1 into the clear/disable bit (which location is 16-bit higher in the register). An undefined toggle status will result from setting both the enable and disable bits for a specific function at the same time. The only function at this point in time is user LED on/off.
On read access the same register represents the status register.

Bit	write Function	read Function
31	Clear reserved 15 (*)	0
30	Clear reserved 14 (*)	0
29	Clear reserved 13 (*)	0
28	Clear reserved 12 (*)	0
27	Clear reserved 11 (*)	0
26	Clear reserved 10 (*)	0
25	Clear reserved 9 (*)	0
24	Clear reserved 8 (*)	0
23	Clear reserved 7 (*)	0
22	Clear reserved 6 (*)	0
21	Clear reserved 5 (*)	0
20	Clear reserved 4 (*)	0
19	Clear reserved 3 (*)	0
18	Clear reserved 2 (*)	0
17	Switch off LED test	0
16	Switch off user LED (*)	0
15	Set reserved 15	Status reserved 15
14	Set reserved 14	Status reserved 14
13	Set reserved 13	Status reserved 13
12	Set reserved 12	Status reserved 12
11	Set reserved 11	Status reserved 11
10	Set reserved 10	Status reserved 10
9	Set reserved 9	Status reserved 9
8	Set reserved 8	Status reserved 8
7	Set reserved 7	Status reserved 7
6	Set reserved 6	Status reserved 6
5	Set reserved 5	Status reserved 5
4	Set reserved 4	Status reserved 4
3	Set reserved 3	Status reserved 3
2	Set reserved 2	Status reserved 2
1	Switch on LED test	Status LED test
0	Switch on user LED	Status User LED (1=LED on, 0=LED off)

(*) denotes power up default setting

7.5.6 Firmware Options register
#define SIS8300-L_FIRMWARE_OPTIONS_REG   0x05

This register holds the information of the Xilinx firmware option features.

Bit	read Function
31	reserved
30	...
...	...
16	reserved
15	reserved
14	reserved
13	reserved
12	reserved
11	reserved
10	reserved
9	reserved
8	reserved
7	reserved
6	DUAL_OPTICAL_INTERFACE_EN
5	DUAL_PORT14_15_INTERFACE_EN
4	DUAL_PORT12_13_INTERFACE_EN
3	none
2	DUAL_CHANNEL_SAMPLING
1	RINGBUFFER_DELAY_EN
0	TRIGGER_BLOCK_EN

7.5.7 ADC Acquisition Control/Status register

```
#define SIS8300-L_ACQUISITION_CONTROL_STATUS_REG    0x10
```

Bit	write	read
31		0
..		0
..		0
8		0
7		Status: DDR3 Memory Init OK
6		0
5		Status: internal Sample Logic Buffer FIFO Not Empty
4		Status: internal Sample Logic Busy
3		0
2	'1' : Disable Sampling (Reset Sample Logic)	0
1	'1' : Arm Sampling (Start with next trigger)	Status: Arm for trigger (Wait for trigger)
0	'1' : Start Sampling immediately (Arm and Start/Trigger)	Status: Sampling Busy

The power up default value is 0x80

7.5.8 ADC Sample Control register

```
#define SIS8300-L_SAMPLE_CONTROL_REG      0x11
```

ADC channels can be disabled from storing data to memory by setting the corresponding disable bit in this register.

Bit	write
31	
...	
...	
12	
11	Enable external Trigger
10	Enable internal Trigger
9	Disable Sampling Ch10
8	Disable Sampling Ch9
7	
6	
5	
4	
3	
2	Disable Sampling Ch3
1	Disable Sampling Ch2
0	Disable Sampling Ch1

The power up default value is 0x0

7.5.9 MLVDS Input/Output Control register

```
#define SIS8300-L_MLVDS_IO_CONTROL_REG    0x12
```

Bit	Write	Read
31	Enable LVDS Output Bit 7	Enable LVDS Output Bit 7
30	Enable LVDS Output Bit 6	Enable LVDS Output Bit 6
..		
25	Enable LVDS Output Bit 1	Enable LVDS Output Bit 1
24	Enable LVDS Output Bit 0	Enable LVDS Output Bit 0
23	LVDS Output Bit 7	LVDS Output Bit 7
22	LVDS Output Bit 6	LVDS Output Bit 6
...		
17	LVDS Output Bit 1	LVDS Output Bit 1
16	LVDS Output Bit 0	LVDS Output Bit 0
15	LVDS Input 7 External Trigger Enable	LVDS Input 7 External Trigger Enable
14	LVDS Input 6 External Trigger Enable	LVDS Input 6 External Trigger Enable
...		
9	LVDS Input 1 External Trigger Enable	LVDS Input 1 External Trigger Enable
8	LVDS Input 0 External Trigger Enable	LVDS Input 0 External Trigger Enable
7	LVDS Input 7 External Trigger falling edge	LVDS Input Bit 7
6	LVDS Input 6 External Trigger falling edge	LVDS Input Bit 6
...		
1	LVDS Input 1 External Trigger falling edge	LVDS Input Bit 1
0	LVDS Input 0 External Trigger falling edge	LVDS Input Bit 0

Note: external trigger in signals are synchronized with the FPGA CLK05

7.5.10 Harlink Connector Input/Output Control register

```
#define SIS8300-L_HARLINK_IO_CONTROL_REG          0x13
```

Bit	Write	Read
31	No function	0
30	No function	0
..		
21	No function	0
20	Harlink Test Output Enable	Harlink Test Output Enable
19	Harlink Test Output 4 (*)	Harlink Test Output 4
18	Harlink Test Output 3 (*)	Harlink Test Output 3
17	Harlink Test Output 2 (*)	Harlink Test Output 2
16	Harlink Test Output 1 (*)	Harlink Test Output 1
15	Harlink Input 4 External Trigger falling edge	Harlink Input 4 External Trigger falling edge
14	Harlink Input 3 External Trigger falling edge	Harlink Input 3 External Trigger falling edge
13	Harlink Input 2 External Trigger falling edge	Harlink Input 2 External Trigger falling edge
12	Harlink Input 1 External Trigger falling edge	Harlink Input 1 External Trigger falling edge
11	Harlink Input 4 External Trigger Enable	Harlink Input 4 External Trigger Enable
10	Harlink Input 3 External Trigger Enable	Harlink Input 3 External Trigger Enable
9	Harlink Input 2 External Trigger Enable	Harlink Input 2 External Trigger Enable
8	Harlink Input 1 External Trigger Enable	Harlink Input 1 External Trigger Enable
7	No function	0
6	No function	0
5	No function	0
4	No function	0
3	No function	Harlink Input 4
2	No function	Harlink Input 3
1	No function	Harlink Input 2
0	No function	Harlink Input 1

(*): only if "Harlink Test Output Enable" = 1
(**): only if "Harlink Test Output Enable" = 0

Harlink Connector Input(1) : external trigger In
Harlink Connector Output(1) : adc chx (or) trigger out (**)

Note: external trigger in signals are synchronized with the FPGA CLK05

7.5.11 Link1 Data FIFO

```
#define SIS8300-L_LINK1_FIFO_DATA_REG          0x14
```

Bit	Write	Read
31	Transmit Data D31	Received Data D31
	D30	D30
…	…	…
	D1	D1
0	Transmit Data D0	Received Data D0

The power up default value is 0x0

7.5.12 Link1 Control Register

```
#define SIS8300-L_LINK1_FIFO_CONTROL_REG          0x15
```

Bit	Write	Read
31	Fifo reset	0
30	Receive-fifo enable	0
29	0	0
28	..	0
27		Port 14 linkup
26	…	Port 12 linkup
25	0	Opt 1 linkup
24	Set to send a protocol word	Indicates an protocol word into receive-fifo
23	0	0
22	…	0
21		0
20		Out_fifo_write count D8
19		D7
18		…
17		
16		
15		
14		…
13		D1
12		Out_fifo_write count D0
11		0
10		0
9		0
8		In_fifo_read count D8
7		D7
6		...
5		
4	…	
3	GTX reset (depending on 'Link select')	
2	Opt1 and Opt2 reset	…
1	Link select D1(0: Opt1 2: Port12 3: Port14)	D1
0	Link select D0	In_fifo_read count D0

Note: The connection between the links must be initialized:
Set Bit 24 = 1 and write `0x1C000000` to `0x14`

7.5.13 Link2 Data FIFO

```
#define SIS8300-L_LINK2_FIFO_DATA_REG        0x16
```

Bit	Write	Read
31	Transmit Data D31	Received Data D31
	D30	D30
…		…
	D1	D1
0	Transmit Data D0	Received Data D0

The power up default value is 0x0

7.5.14 Link2 Control Register

```
#define SIS8300-L_LINK2_FIFO_CONTROL_REG          0x17
```

Bit	Write	Read
31	Fifo reset	0
30	Receive-fifo enable	0
29	0	0
28	..	0
27		Port 15 linkup
26	…	Port 13 linkup
25	0	Opt 2 linkup
24	Set to send a protocol word	Indicates an protocol word into receive-fifo
23	0	0
22	…	0
21		0
20		Out_fifo_write count D8
19		D7
18		…
17		
16		
15		
14		…
13		D1
12		Out_fifo_write count D0
11		0
10		0
9		0
8		In_fifo_read count D8
7		D7
6		...
5		
4	…	
3	GTX reset (depending on 'Link select')	
2	Opt1 and Opt 2 reset	…
1	Link select D1(0: Opt2 2: Port13 3: Port 15)	D1
0	Link select D0	In_fifo_read count D0

Note: The connection between the links must be initialized:
Set Bit 24 = 1 and write `0x1C000000` to `0x16`

7.5.15 Clock Distribution Multiplexer control register

```
#define SIS8300-L_CLOCK_DISTRIBUTION_MUX_REG   0x40
```

The SIS8300-L has 5 IDT ICS853S057 clock multiplexer chips, which are labelled A to E in the clock distribution schematic in section 2.4. The multiplexer control register holds the two select bits for the 5 multiplexer chips as shown in the table below.

The assignment of the inputs to the resources (i.e. clock inputs) is listed in subsection 7.5.15.1.

BIT	access	Name	Function
31-12 FFFFF000	R/W	reserved	no
11-10 00000C00	R/W	MUXE_SEL	Multiplexer E select bits
9-8 00000300	R/W	MUXD_SEL	Multiplexer D select bits
7-6 000000C0	R/W	reserved	no
5-4 00000030	R/W	MUXC_SEL	Multiplexer C select bits
3-2 0000000C	R/W	MUXB_SEL	Multiplexer B select bits
1-0 00000003	R/W	MUXA_SEL	Multiplexer A select bits

7.5.15.1   Multiplexer A Input Signals:

U222 - Sel0 and Sel1 (MUX1A_SEL) = Multiplexer A select lines

Sel1	Sel0	Selected Input - Net Name	Clock source Description
0	0	RTMCLK_0	Clock 2 from µRTM card
0	1	TCLKB_0	Clock 2 (Telecom Clock B) from AMC Connector (Backplane)
1	0	TCLKA_0	Clock 1 (Telecom Clock A) from AMC Connector (Backplane)
1	1	OSC_CLK0	Onboard Clock chip (250MHz)

7.5.15.2   Multiplexer B Input Signals:

U223 - Sel0 and Sel1 (MUX1B_SEL) = Multiplexer B select lines

Sel1	Sel0	Selected Input - Net Name	Clock source Description
0	0	RTMCLK_1	Clock 2 from µRTM card
0	1	TCLKB_1	Clock 2 (Telecom Clock B) from AMC Connector (Backplane)
1	0	TCLKA_1	Clock 1 (Telecom Clock A) from AMC Connector (Backplane)
1	1	OSC_CLK1	Onboard Clock chip (250MHz)

7.5.15.3   Multiplexer C Input Signals:

U240 - Sel0 and Sel1 (MUXAB_SEL) = Multiplexer C select lines

Sel1	Sel0	Selected Input - Net Name	Clock source Description
0	0	EXT_CLKB0	Clock from Harlink Connector "CI1-4 IN" (frontpanel)
0	1	EXT_CLKA0	Clock from SMA Connector "CLK IN" (frontpanel)
1	0	MUXA_CLK1	Multiplexer A Output Signal
1	1	MUXB_CLK1	Multiplexer B Output Signal

7.5.15.4   Multiplexer D Input Signals:

U250 - Sel0 and Sel1 (MUX2A_SEL) = Multiplexer D select lines

Sel1	Sel0	Selected Input - Net Name	Clock source Description
0	0	MUXA_CLK0	Multiplexer A Output Signal
0	1	MUL_CLK1	Clock Multiplier (U242) Output 2 Signal
1	0	EXT_CLKB1	Clock from Harlink Connector "CI1-4 IN" (frontpanel)
1	1	EXT_CLKA1	Clock from SMA Connector "CLK IN" (frontpanel)

7.5.15.5   Multiplexer E Input Signals:

U251 - Sel0 and Sel1 (MUX2B_SEL) = Multiplexer E select lines

Sel1	Sel0	Selected Input - Net Name	Clock source Description
0	0	MUXB_CLK0	Multiplexer B Output Signal
0	1	MUL_CLK0	Clock Multiplier (U242) Output 1 Signal
1	0	EXT_CLKB2	Clock from Harlink Connector "CI1-4 IN" (frontpanel)
1	1	EXT_CLKA2	Clock from SMA Connector "CLK IN" (frontpanel)

7.5.16 Clock Distribution AD9510 Serial Interface (SPI) interface register
```
#define SIS8300-L_AD9510_SPI_REG        0x41
```

The parameters of the Clock Distribution IC AD9510 chips can be configured with the SPI (serial Peripheral Interface).

Bit	Write	read	
31	Cmd Bit 1	Write/Read Logic BUSY Flag	
30	Cmd Bit 0		
29	Set "Function" Output Level	Status of Set "Function" Output Level	
28	Select "Function" synchronisation CLK	Status of Select "Function" synchronisation CLK	
...			
...			
25		Status AD9510 #2	
24	AD9510 #2 Select Bit	Status AD9510 #1	
23	Read Cycle Bit		
22			
21			
20	Address Bit 12		
19	Address Bit 11		
...	...		
...	...		
12	Address Bit 4		
11	Address Bit 3		
10	Address Bit 2		RW CMD
9	Address Bit 1		
8	Address Bit 0		
7	Write Data Bit 7 (MSB)	Read Data Bit 7 (MSB)	
6	Write Data Bit 6	Read Data Bit 6	
..	...		
1	Write Data Bit 1	Read Data Bit 1	
0	Write Data Bit 0 (LSB)	Read Data Bit 0 (LSB)	

The power up default value is 0x20000000

Command Bit (31:30) Explanation:

Cmd Bit 1	Cmd Bit 0	Command
0	0	No Function
0	1	R/W CMD
1	0	Function CMD Generates a pulse at the Function Input pin of the AD9510 which is synchronous to the selected clock. The clock selection is done via Bit 28 (Function Syn CLK). The actual function depends on the programming of the selected AD9510
1	1	Reserved

Select "Function" synchronisation CLK Bit (28) Explanation:

Bit 28	Clock Source
0	PCI Clock
1	FPGA CLK 69

Note:
   1. enable READ by writing 0x90 to addr 0x0
   2. and set Read Cycle Bit

Note:
Please refer to the SIS8300-L_AD9510_SPI_Setup routine as illustration and to the AD9510 documentation for details.

7.5.17 Clock Multiplier IC SI5326 SPI interface register

```
#define SIS8300-L_CLOCK_MULTIPLIER_SPI_REG    0x42
```

Several parameters of the Clock Multiplier SI5326   chip can be configured with the SPI (serial Peripheral Interface).
Please refer to the documentation of the SI5326 chip for details.

Bit	Write	read
31	Cmd Bit 1	Write/Read Logic BUSY Flag
30	Cmd Bit 0	Reset, Decrement or Increment Cmd BUSY Flag
29		
...		
17		Si53xx LOL Status
16		Si53xx INT_C1B Status
15	Instruction Byte Bit 7	
..		
..		
8	Instruction Byte Bit 0	
7	Address/Data Byte Bit 7	Read Data Bit 7 (MSB)
..		
..	..	Read Data Bit 1
0	Address/Data Byte Bit 0	Read Data Bit 0 (LSB)

The power up default value is 0x0

Cmd Bit 1	Cmd Bit 0	Command
0	0	Execute SPI Write/Read Cmd
0	1	Reset Cmd
1	0	Decrement Cmd
1	1	Increment Cmd

Reset Cmd: generates an 1us reset pulse

Decrement Cmd: generates an 1us Skew Decrement pulse
Increment Cmd: generates an 1us Skew Increment pulse

Note: INC/DEC Time between consecutive pulses must be greater than 16ms !

7.5.18 Clock Synthesizer IC Si5338A I2C interface register
```
#define SIS8300-L_MGTCLK_SYNTH_I2C_REG    0x43
```

Several parameters of the Clock Synthesizer Si5338A chip can be configured with the I2C Interface.
Please refer to the documentation of the Si5338A chip for details.

Bit	Write	read
31	unused	Write/Read Logic BUSY Flag
30	unused	0
29	unused	0
28	unused	0
27	unused	0
..		
16	unused	0
15	unused	0
14	unused	0
13	Byte Read cycle	0
12	Byte Write cycle	0
11	Issue STOP condition	0
10	Issue REPEATSTART condition	0
9	Issue START condition	0
8	Master I2C ACK bit, written during reads	Device I2C ACK bit, read during writes
7	Write Data Bit 7 (MSB)	Read Data Bit 7 (MSB)
..		
1	Write Data Bit 1	Read Data Bit 1
0	Write Data Bit 0 (LSB)	Read Data Bit 0 (LSB)

The power up default value is 0x0

7.5.19 FPGA Boot SPI Flash interface

```
#define SIS8300-L_SPI_FLASH_REG    0x44
```

Bit	Write	read
31	unused	SPI logic busy
30	unused	0
..		
35	Release Flash interface	0
24	Clear Flash chipselect	0
23	unused	0
..		
11	unused	0
10	Start Byte exchange	0
9	Claim Flash interface	Flash interface status
8	Set Flash chipselect	Flash chipselect status
7	Write Data Bit 7 (MSB)	Read Data Bit 7 (MSB)
..		
1	Write Data Bit 1	Read Data Bit 1
0	Write Data Bit 0 (LSB)	Read Data Bit 0 (LSB)

The SPI Flash control lines are internally multiplexed between this register and the onboard MMC controller pins. In order to gain access to the flash via this register, the multiplexer has to be set via bit 10 and cleared via bit 25 after all transfers are done.

7.5.20 DAC Control register

```
#define SIS8300-L_DAC_CONTROL_REG   0x45
```

Bit	write	read
31	Tap Delay Write Pulse	
30		
29		
28	Tap Delay value bit 4	
27	Tap Delay value bit 3	
26	Tap Delay value bit 2	
25	Tap Delay value bit 1	
24	Tap Delay value bit 0	
23		
...		
...		
20		
19		
18		
17	DAC Clock Multiplexer select bit 1	DAC Clock Multiplexer select bit 1
16	DAC Clock Multiplexer select bit 0	DAC Clock Multiplexer select bit 0
15		
...		
12		
11		
10		
9		
8	DAC DCM Reset pulse	
7		
6		
5	Power Down	0: power down, 1: power up
4	TORB Two's-Complement/Binary Select	0: binary, 1: Two's complement
..		
1	Test Mode Bit 1	
0	Test Mode Bit 0	

The power up default value is 0x0

select bit 1	select bit 0	DAC clock source
0	0	FPGA clock
0	1	Clock Divider AD9510 #2 Out 0
1	0	Clock Divider AD9510 #1 Out 0
1	1	RTM Clk4

Test Mode Bit 1	Test Mode Bit 0	DAC Test Mode
0	0	Data from DAC Data register
0	1	Ramp Test Mode
1	0	ADC1/ADC2 -> DAC1/DAC2
1	1	reserved

Note: ADC 1 Clock is used as DAC clock

7.5.21 DAC Data register

```
#define SIS8300-L_DAC_DATA_REG      0x46
```

Bit	write
31	DAC2 Data 15
...	
...	
16	DAC2 Data 0
15	DAC1 Data 15
...	
...	
0	DAC1 Data 0

The power up default value is 0x0, data=0 → +1 V, data=0xFFFF → - 1V output (with TORB=1, i.e. in Two's complement mode)

Note: The default DAC range at DAC Out SMA Connector is -1V,…,+1V into a 50 Ω load

7.5.22 ADC Serial Interface (SPI) interface register

```
#define SIS8300-L_ADC_SERIAL_INTERFACE_REG    0x48
```

Several parameters of the ADC AD9268 chip can be configured with the SPI (serial Peripheral Interface).
Please refer to the documentation of the ADC AD9268 chip for details.

Bit	write	read
31	ADC Synch cmd	Write/Read Logic BUSY Flag
...		
...		
26	ADC Select Mux Bit 2	
25	ADC Select Mux Bit 1	
24	ADC Select Mux Bit 0	
23	Read Cmd	
22		
21		
20	Address Bit 12	
19	Address Bit 11	
12	Address Bit 4	
11	Address Bit 3	
10	Address Bit 2	
9	Address Bit 1	
8	Address Bit 0	
7	Write Data Bit 7 (MSB)	Read Data Bit 7 (MSB)
6	Write Data Bit 6	Read Data Bit 6
..		
1	Write Data Bit 1	Read Data Bit 1
0	Write Data Bit 0 (LSB)	Read Data Bit 0 (LSB)

The power up default value is 0x0

ADC Synch Cmd : generates an synch pulse with AD9510 #1 FPGA clock

7.5.23 ADC Input Tap delay registers

```
#define SIS8300-L_ADC_INPUT_TAP_DELAY        0x49
```

The input tap delay registers are used to adjust the FPGA data strobe timing.

Bit	31-13	12	11	10	9	8	7-6	5-0
Function	None	ADC 9/10 Select	ADC 7/8 Select	ADC 5/6 Select	ADC 3/4 Select	ADC 1/2 Select	None	Tap delay value (x 78ps)

Bit	write	read
31		Tap Delay Logic BUSY Flag
...		
..		
..		
12	ADC 9/10 Select Bit	
11	ADC 7/8 Select Bit	
10	ADC 5/6 Select Bit	
9	ADC 3/4 Select Bit	
8	ADC 1/2 Select Bit	
7	Tap delay value Bit 7	Tap delay value Bit 7
6	Tap delay value Bit 6	Tap delay value Bit 6
..		
1	Tap delay value Bit 1	Tap delay value Bit 1
0	Tap delay value Bit 0	Tap delay value Bit 0

7.5.24 Trigger registers

The Trigger Block contains Logic to generate internal triggers (only implemented if the Firmware Option register bit TRIGGER_BLOCK_EN = 1).
Two types are implemented: A "threshold trigger" and a "FIR trigger".

7.5.24.1   Trigger setup registers

```
#define SIS8300-L_TRIGGER_SETUP_CH1_REG        0x100
..
#define SIS8300-L_TRIGGER_SETUP_CH10_REG       0x109
```

These read/write registers hold the 8-bit wide trigger pulse length (in sample clocks), the Peaking and Gap Time of the trapezoidal FIR filter.
(Gap Time = SumG Time – Peaking Time)

Bit	Function	
31	Reserved	
..	..	
26	Enable Trigger	
25	GT trigger condition	
24	FIR Trigger Mode (0: Threshold Trigger; 1: FIR Trigger)	
23	Puls Length bit 7	
22	Puls Length bit 6	
21	Puls Length bit 5	
20	Puls Length bit 4	Trigger Pulse Length
19	Puls Length bit 3	
18	Puls Length bit 2	
17	Puls Length bit 1	
16	Puls Length bit 0	
15	reserved	
14	reserved	
13	reserved	SumG time (only FIR trigger)
12	SumG bit 4	(time between both sums)
11	SumG bit 3	
10	SumG bit 2	
9	SumG bit 1	
8	SumG bit 0	
7	reserved	
6	reserved	Peaking time P (only FIR trigger)
5	reserved	
4	P bit 4	$\sum_{i=x}^{x+P} S_i$
3	P bit 3	
2	P bit 2	
1	P bit 1	
0	P bit 0	

The power up default value reads 0x 00000000

Si: Sum of ADC input sample stream from x to x+P
P: Peaking time (number of values to sum)
SumG: SumGap time (distance in clock ticks of the two running sums)

The maximum SumG time: 16 (clocks)
The minimun SumG time: 1 (clocks)
Values > 16 will be set to 16
Value = 0 will be set to 1

The maximum Peaking time: 16 (clocks)
The minimun Peaking time: 1 (clocks)
Values > 16 will be set to 16
Value = 0 will be set to 1

7.5.24.2   Trigger Threshold registers

```
#define SIS8300-L_TRIGGER_THRESHOLD_CH1_REG          0x110
..
#define SIS8300-L_TRIGGER_THRESHOLD_CH10_REG         0x119
```

These read/write registers hold the threshold values for the 10 ADC channels.

7.5.24.2.1   Trigger Threshold
FIR Trigger Mode = 0

Bit	31-16	15-0
Function	Threshold value OFF	Threshold value ON

default after Reset: 0x0

A trigger output pulse is generated on two conditions:

* GT is set (GT) in trigger setup register:
  the trigger Out pulse will be issued if the actual sampled ADC value **goes** above the threshold value ON **and** OFF. A new Trigger Out Pulse will be suppressed until the ADC value **goes** below the threshold value OFF.

* GT is cleared (LT) in trigger setup register:
  the trigger Out pulse will be issued if the actual sampled ADC value **goes** below the threshold value ON **and** OFF. A new trigger Out pulse will be suppressed until the ADC value **goes** above the threshold value OFF.

  the trigger Out pulse will be issued if the actual sampled ADC value **goes** below the threshold value.

GT: greater than
LT: lower than

7.5.24.2.2 FIR Trigger Threshold

FIR Trigger Mode = 1

Bit	31-20	19-0
Function	None	**Trapezoidal t**hreshold value

default after Reset: 0x0

The value of the Sum (trapezoidal value) depends on the peaking time P. Therefore the selection of the value of the Trapezoidal threshold depends on P also.

Trapezoidal value calculation:
Trapezoidal value = (SUM2 – SUM1)

Where

$$SUM1 \; = \; \sum_{i=x}^{x+P} Si$$

$$SUM2 \; = \; \sum_{j=x+sumG}^{x+P+sumG} Sj$$

The FIR filter logic generates the Trapezoidal by subtraction of the two running sums. This implies, that the internal value of the trapezoid is on average 0.

A trigger output pulse is generated:

- GT is set (GT):
  the Trigger Out Pulse will be issued if the actual trapezoidal value **goes** above the programmable trapezoidal threshold value

- GT is cleared (LT):
  the Trigger Out Pulse will be issued if the actual trapezoidal value **goes** below the **negated** programmable trapezoidal threshold value

7.5.25 Memory Sample Start Address / Actual Sample Address registers

```
#define SIS8300-L_SAMPLE_START_ADDRESS_CH1_REG        0x120
..
#define SIS8300-L_SAMPLE_START_ADDRESS_CH10_REG       0x129
```

If the Firmware Option register bit DUAL_CHANNLE_SAMPLING = 0 then all 10 registers are used.
If the Firmware Option register bit DUAL_CHANNLE_SAMPLING = 1 then only the first 5 registers are used.

The **write function** to these registers defines the memory start address.
The value is given in 256-bit blocks.

Write Function: ADC chx Memory Sample Start Address

Bit	31-24	23-0
	reserved	Memory Sample Start Address (256-bit blocks) (16-bit word address x 16)

default after Reset: 0x0

Explanation (memory sample start address)
The contents of the **sample memory start address** register is assigned as memory data storage address with the arm command (key address arm sampling) or with the enable command (key address enable sampling).

The **read function** from these registers give the information of the actual sampling address for the given ADC channel.
(at the moment: only valid if the logic is not busy!)

Read Function : ADC chx Actual Sample Address

Bit	31-24	23-0
	reserved	Actual Sample Address (in 256-bit Blocks) (16-bit word address x 16)

The value is given in 256-bit Blocks (16bit word address x 16)

7.5.26 Sample Length register

```
#define SIS8300-L_SAMPLE_LENGTH_REG            0x12A
```

This register defines the number of sample blocks of each ADC channel.

The size of one sample block for each ADC channel is 256-bit (16 x 16-bit word).

Bit	31-24	23-0
Function	reserved	Sample Block Length

default after Reset: 0x0

7.5.27 Ringbuffer Delay register

```
#define SIS8300-L_PRETRIGGER_DELAY_REG          0x12B
```

This register defines the number of pre trigger delay samples for all channels.
The maximum pre trigger delay value is 2046.

Bit	31-12	11-0
Function	reserved	Delay value

7.5.28 Test Histogram Pattern-Memory Address register

```
#define SIS8300-L_TEST_HISTO_MEM_ADDR                0x12C
```

This register defines the Write Address of the Test Histogram Pattern Memory (4Kx28).

Bit	31-12	11-0
Function	reserved	Write Address

default after Reset: 0x0

7.5.29 Test Histogram Pattern-Memory Data Write register

```
#define SIS8300-L_TEST_HISTO_MEM_DATA_WR             0x12D
```

The write function to this register writes the data to the Test Histogram Pattern Memory (4Kx28).

Bit	31-28	27-0
Function	reserved	Data (Histogram index)

default after Reset: 0x0

7.5.30 Test Histogram Control register

```
#define SIS8300-L_TEST_HISTO_CONTROL          0x12E
```

Bit	Write	Read
31	Test Histogram Control Bit 1	Test Histogram Control Bit 1
30	Test Histogram Control Bit 0	Test Histogram Control Bit 0
29	reserved	Copy Logic Busy Flag
28	reserved	
...		
17	reserved	
16	reserved	
15	Copy Length Bit 15	Read Copy Length Bit 15 (MSB)
..		
1	Copy Length Bit 1	Read Copy Length Bit 1
0	Copy Length Bit 0	Read Copy Length Bit 0 (LSB)

The power up default value is 0x0

Control Bit 1	Control Bit 0	Command
0	0	Reset
0	1	Copy "Length-1" values from "Test Histogram Pattern-Memory" (start with addr=0) to the Histogram-Fifo
1	0	Continuously histogramming of ADC channel 1 and 2 (average of 128 values, every (125/128) us) - Histogram of Channel 1: Byte addr 0x0 – 0x3ffff (64K) - Histogram of Channel 2: Byte addr 0x40000 – 0x7ffff (64K)
1	1	Continuously histogramming of Coincidence channel 1 / channel 2 (average of 128 values, every (125/128) us) - Coincidence Histogram 256 x 256 : Byte Addr 0x0 = 0x3 ffff (64K)

7.5.31 SIS8900 RTM LVDS Test Input/Output Control register
```
#define SIS8300-L_RTM_LVDS_IO_CONTROL_REG        0x12F
```

Bit	Write	Read
31	-	0
30	-	0
29	-	0
28	-	0
27	-	0
26	Enable RTM LVDS Output Bit 8	Enable RTM LVDS Output Bit D 8
25	Enable RTM LVDS Output Bit 7	Enable RTM LVDS Output Bit D 7
24	Enable RTM LVDS Output Bit 6	Enable RTM LVDS Output Bit D 6
23	-	0
22	-	0
21	-	0
20	-	0
19	-	0
18	RTM LVDS Output Bit D 8	RTM LVDS Output Bit D 8
17	RTM LVDS Output Bit D 7	RTM LVDS Output Bit D 7
16	RTM LVDS Output Bit D 6	RTM LVDS Output Bit D 6
15	-	0
14	-	0
13	-	0
12	-	0
11	RTM Z3-TCLK enable	RTM Z3-TCLK enable
10	RTM Z3-ILOCK enable	RTM Z3-ILOCK enable
9	RTM Z3-ILOCK1	RTM Z3-ILOCK1
8	RTM Z3-ILOCK0	RTM Z3-ILOCK0
7	-	0
6	-	0
5	-	RTM LVDS Input Bit D5
4	-	RTM LVDS Input Bit D4
3	-	RTM LVDS Input Bit D3
2	-	0
1	-	0
0	-	0

7.5.32 Read DMA System Destination address (lower 32bits)

```
#define DMA_READ_DST_ADR_LO32                        0x200
```

This register holds the lower 32bits of the destination address in system memory into which the card will transfer data.

Bit	31-0
Function	System memory address (lower 32bits)

7.5.33 Read DMA System Destination address (upper 32bits)

```
#define DMA_READ_DST_ADR_HI32                        0x201
```

This register holds the upper 32bits of the destination address in system memory into which the card will transfer data.

Bit	31-0
Function	System memory address (upper 32bits)

7.5.34 Read DMA Card Memory Source address

```
#define DMA_READ_SRC_ADR_LO32                        0x202
```

This register holds the 32bit source address in the cards address space which is used to select the data source which is read from.

Bit	31-0
Function	Card address space

The address layout is:

    512MB Models:

```
Address 0x0 - 0x1FFFFFFF: DDR3 Memory readout
Address 0x80000000 - 0xAFFFFFFF: Repeated User DMA space
```

    1GB Models:

```
Address 0x0 - 0x3FFFFFFF: DDR3 Memory readout
Address 0x80000000 - 0xAFFFFFFF: Repeated User DMA space
```

7.5.35 Read DMA Transfer length

```
#define DMA_READ_LEN                                    0x203
```

This register holds the amount of data which is going to be transferred.

Bit	31-0
Function	DMA Transfer length

7.5.36 Read DMA Control

```
#define DMA_READ_CTRL                                   0x204
```

This register starts the Read DMA process and allows to poll the transfer status.

Bit	write	read
31	unused	0
...		0
1	unused	0
0	Start DMA	DMA running

7.5.37 Readout DMA Sample byte swap

```
#define DMA_READ_BYTESWAP                               0x205
```

This register allows swapping each byte in a sample for optimizing data handling on big/little endian machines.

Example for disabled swapping:

Byte address offset:	Sample value
00	Sample 0 lo byte (LSB)
01	Sample 0 hi byte (MSB)
02	Sample 1 lo byte (LSB)
03	Sample 1 hi byte (MSB)

Example for enabled swapping:

Byte address offset:	Sample value
00	Sample 0 hi byte (MSB)
01	Sample 0 lo byte (LSB)
02	Sample 1 hi byte (MSB)
03	Sample 1 lo byte (LSB)

Bit	write	read
31	unused	0
...		0
1	unused	0
0	Byteswap enable	Byteswap enable status

7.5.38 Write DMA System Source address (lower 32bits)

```
#define DMA_WRITE_DST_ADR_LO32                          0x210
```

This register holds the lower 32bits of the destination address in system memory from which the card will transfer data.

Bit	31-0
Function	System memory address (lower 32bits)

7.5.39 Write DMA System Source address (upper 32bits)

```
#define DMA_WRITE_DST_ADR_HI32                          0x211
```

This register holds the upper 32bits of the destination address in system memory from which the card will transfer data.

Bit	31-0
Function	System memory address (upper 32bits)

7.5.40 Write DMA Card Memory Destination address

```
#define DMA_WRITE_DST_ADR_LO32                          0x212
```

This register holds the 32bit destination address in the cards address space which is used to select the data source which is written to.

Bit	31-0
Function	Card address space

Depending on the populated amount of dram on the module the address layout is:

512MB Models:

```
Address 0x0 – 0x1FFFFFFF: DDR3 Memory
```

1GB Models:

```
Address 0x0 - 0x3FFFFFFF: DDR3 Memory
```

7.5.41 Write DMA Transfer length

```
#define DMA_WRITE_LEN                                0x213
```

This register holds the amount of data which is going to be transferred.

Bit	31-0
Function	DMA Transfer length

7.5.42 Write DMA Control

```
#define DMA_WRITE_CTRL                              0x214
```

This register starts the Write DMA process and allows to poll the transfer status.

Bit	write	read
31	unused	0
...		0
1	unused	0
0	Start DMA	DMA running

7.5.43 DAQ Done DMA Chain Control

```
#define DAQ_DMA_CHAIN                                    0x216
```

This register allows the chaining of the DAQ Done Signal into the DMA Start Signal.

Bit	write	read
31	unused	0
...		0
1	unused	0
0	DAQ Done DMA Start Chain enable	Chain enabled

7.5.44 IRQ Enable

```
#define IRQ_ENABLE                                       0x220
```

This register enables each interrupt source for interrupt generation. The register is implemented as a J-K register.

Bit	write	read
31	Disable User IRQ	0
30	Disable DAQ Done IRQ	0
29	unused	0
...		
18	unused	0
17	Disable Write DMA Done IRQ	0
16	Disable Read DMA Done IRQ	0
15	Enable User IRQ	User IRQ enabled status
14	Enable DAQ Done IRQ	DAQ Done IRQ enabled status
13	unused	0
...		
2	unused	0
1	Enable Write DMA Done IRQ	Write DMA Done IRQ enabled status
0	Enable Read DMA Done IRQ	Read DMA Done IRQ enabled status

7.5.45 IRQ Status

```
#define IRQ_STATUS                                0x221
```

This register lists the latched interrupt bits for which an interrupt has been generated.

Bit	write	read
31	unused	0
...		0
16	unused	0
15	unused	User IRQ happened
14	unused	DAQ Done IRQ happened
13	unused	0
...		
2	unused	0
1	unused	Write DMA Done IRQ happened
0	unused	Read DMA Done IRQ happened

7.5.46 IRQ Clear

```
#define IRQ_CLEAR                                 0x222
```

This register clears any handled interrupts and allows the logic to generate new interrupts.

Bit	write	read
31	unused	0
...		0
16	unused	0
15	User IRQ clear	0
14	DAQ Done IRQ clear	0
13	unused	0
...		
2	unused	0
1	Write DMA Done IRQ clear	0
0	Read DMA Done IRQ clear	0

7.5.47 IRQ Refresh

```
#define IRQ_REFRESH                               0x223
```

This register refreshes the interrupt logic. This might be needed in the case an interrupt happens while the software interrupt service routine was still handling the previous interrupt.

Bit	write	read
any	Refresh IRQ logic	0

7.5.48 RAM FIFO debug register

```
#define RAM_FIFO_DEBUG                          0x231
```

This register provides fifo information of internal read and write fifo pipes in the DDR3 memory controller. It also allows to selectively reset each data path.

Bit	write	read
31	Reset read address/data fifos	Reset status
30	Reset write address/data fifos	Reset status
29	Fifo count select bit 1	Fifo count select bit 1 status
28	Fifo count select bit 0	Fifo count select bit 0 status
27	unused	Selected fifo status bit 27
...		
0	unused	Selected fifo status bit 0

Fifo 0 status bits:

Bit	write	read
27	unused	0
...		
12	unused	0
11	unused	Read data fifo almost empty
10	unused	Read data fifo empty
9	unused	Read data fifo fill count bit 9
...		
0	unused	Read data fifo fill count bit 0

Fifo 1 status bits:

Bit	write	read
27	unused	0
...		
10	unused	0
9	unused	Read address fifo fill count bit 9
...		
0	unused	Read address fifo fill count bit 0

Fifo 2 status bits:

Bit	write	read
27	unused	0
...		
10	unused	0
9	unused	Write data fifo fill count bit 9
...		
0	unused	Write data fifo fill count bit 0

Fifo 3 status bits:

Bit	write	read
27	unused	0
...		
10	unused	0
9	unused	Write address fifo fill count bit 9
...		
0	unused	Write address fifo fill count bit 0

7.6   External register interface

The external register interface provides the user with the possibility to implement up to 256 32bit registers on the top level of the HDL design. The registers are embedded into the devices regular register space from address 0x400 to 0x4FF.

The External register interface consists of the following signals:

```
reg_0x400_0x4FF_adr : out std_logic_vector(7 downto 0);
reg_0x400_0x4FF_wr_data : out std_logic_vector(31 downto 0);
reg_0x400_0x4FF_rd_data : in std_logic_vector(31 downto 0);
reg_0x400_0x4FF_wr_en : out std_logic;
reg_0x400_0x4FF_rd_en : out std_logic;
```

reg_0x400_0x4FF_adr:
    8bit wide addressbus which selects the next register to be read from or written to.

reg_0x400_0x4FF_wr_data:
    32bit wide databus which holds the data to be written to the addressed register.

reg_0x400_0x4FF_rd_data:
    32bit wide databus to which the user logic must provide the read data from the addressed register.

reg_0x400_0x4FF_wr_en:
    Write enable pulse to indicate that a write request has been issued from the PCIe interface.

reg_0x400_0x4FF_rd_en:
    Read enable pulse to indicate to that a read request has been issued from the PCIe interface.

The interface is synchronous to the User Blockram DMA interface clock. See 7.7.

See the following graphs on how the device expects user logic to interact with the interface.
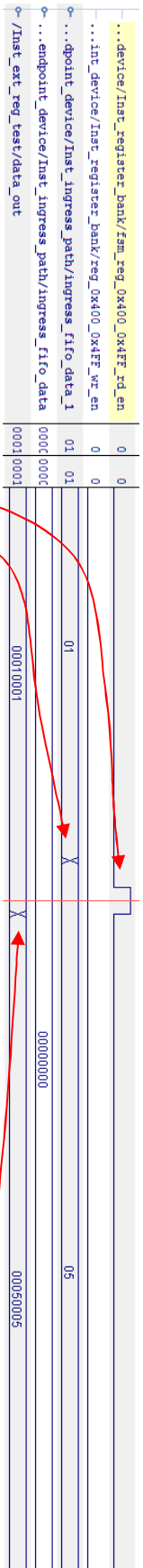
The signals from top to bottom are:
1. register read pulse
2. register write pulse
3. register address bus (8 bits wide)
4. register data write bus (32 bits wide)
5. register data read bus (32 bits wide from user defined register logic)

A read request from the external register space causes the signals to change as follows:

The address bus changes its value to the current selected register.

The read enable signal is pulsed for 1 clockperiod.

The user data from the external register is expected to be valid 1 clockperiod after the read pulse.
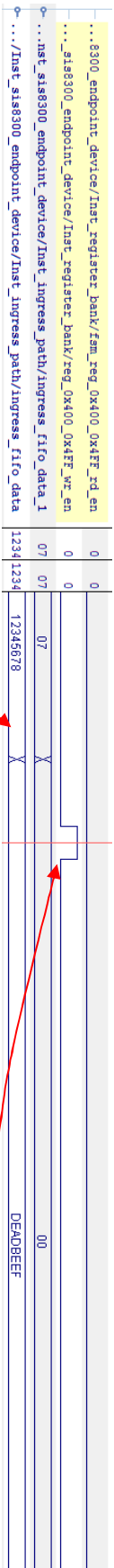
The signals from top to bottom are:
1. register read pulse
2. register write pulse
3. register address bus (8 bits wide)
4. register data write bus (32 bits wide)

A write request to the external register space causes the signals to change as follows:

The address and write data bus change their values to the current selected register and the new register data.

The write enable signal is pulsed for 1 clockperiod.

7.7 User Blockram DMA Interface

The User Blockram DMA interface consists of the following signals:

```
bram_dma_clk : out std_logic;
bram_dma_adr : out std_logic_vector(31 downto 0);
bram_dma_rd_en : out std_logic;
bram_dma_rd_data : in std_logic_vector(63 downto 0);
```

bram_dma_clk:

     Free running 125MHz clock to which the data and control signals are synchronous to

bram_dma_adr:

     32bit wide addressbus which is mapped over the 512MB of onboard sample RAM.
     The addresses ranges from 0x00000000 to 0x1FFFFFFF (512MB)

bram_dma_rd_en:

     (Optional) Read enable pulse for connected logic. The read enable pulse is valid 1
     clockperiod before the data is expected to be valid on the databus.

bram_dma_rd_data:

     64bit wide (due to PCIe endpoint design) databus which holds the data to be
     transmitted over PCIe. The data needs to be valid 1 clockperiod after ram_dma_rd_en
     is valid.

Note:

This interface was built to be directly able to connect to a Xilinx CoreGen generated
blockram module with a 64bit wide read bus. Due to PCIe endpoint design constraints the
user has to read the blockram via DMA in 8byte steps.

7.8 User Interrupt Interface

The User Interrupt interface consists of the following signals.

```
user_irq : in std_logic;
user_irq_clear : out std_logic;
```

user_irq:

     User interrupt pulse input to the PCIe endpoint. The connected logic has to supply a
     pulse of at least 1 clockperiod in length to trigger the interrupt logic. If the driver/user
     software has enabled the corresponding interrupt line, an interrupt is generated on the
     PCIe interface.

user_irq_clear:

     A 1 clock period wide pulse which is triggered, when the driver software has serviced
     the issued interrupt in the interrupt service routine. This pulse may be used to reset any
     logic that depends on user feedback through the software interface.

The interface is synchronous to the User Blockram DMA interface clock. See 7.7.

8 RTM management

Connected RTMs shall be compliant to the PICMG MTCA.4 specification in a way that they must have an on board I²C EEProm (on address 0x50) and a NXP PCF8574-compatible port expander (on address 0x7C).

Required port expander connection map for normal operation:

Port pin	Function
P0	HotSwap Switch (low active)
P1	LED Blue (low active)
P2	LED Red (low active)
P3	LED Green (low active)
P4	PowerGood (low active)
P5	Reset (low active)
P6	PowerEnable (low active)
P7	EEprom Write Protect

The EEprom shall contain any relevant device information (FRU records) about the RTM (refer to PICMG AMC.0). Additionally the EEprom shall contain the new record types defined in PICMG MTCA.4.

In order to be able to decide whether a connected RTM is compatible to the SIS8300-L the RTM record shall contain one of the Zone 3 Identifier records listed in the table below.

Supported Zone 3 Identifier Records (Interface Identifier OEM):

IANA PEN (Private enterprise number)	Zone 3 OEM record	Descripton
0x0092BD (37565)	0x83000001	SIS8300-L Zone3 - v1 compatibility
0x00053F (1343)	0x08020000	DESY DWC Zone3 compatibility
0x0092BD (37565)	0x89000001	SIS8900 Zone3 - v1 compatibility

9   FPGA Firmware upgrade

Two methods can be used to update the FPGA firmware on cards with intact firmware. The first one uses the Xilinx iMPACT software to program the FPGA. The second uses a LINUX tool (which can be found on the product DVD) to update the SPI memory via the PCI Express interface of the card. . The latter method can not be used on cards with flawed firmware.

9.1   iMPACT

Double click with the left mouse button on the SPI/BPI icon and select the *.mcs file to be programmed (e.g. SIS8300L_prom_v83011003.mcs) after a Boundary Scan and the initialization of the chain.
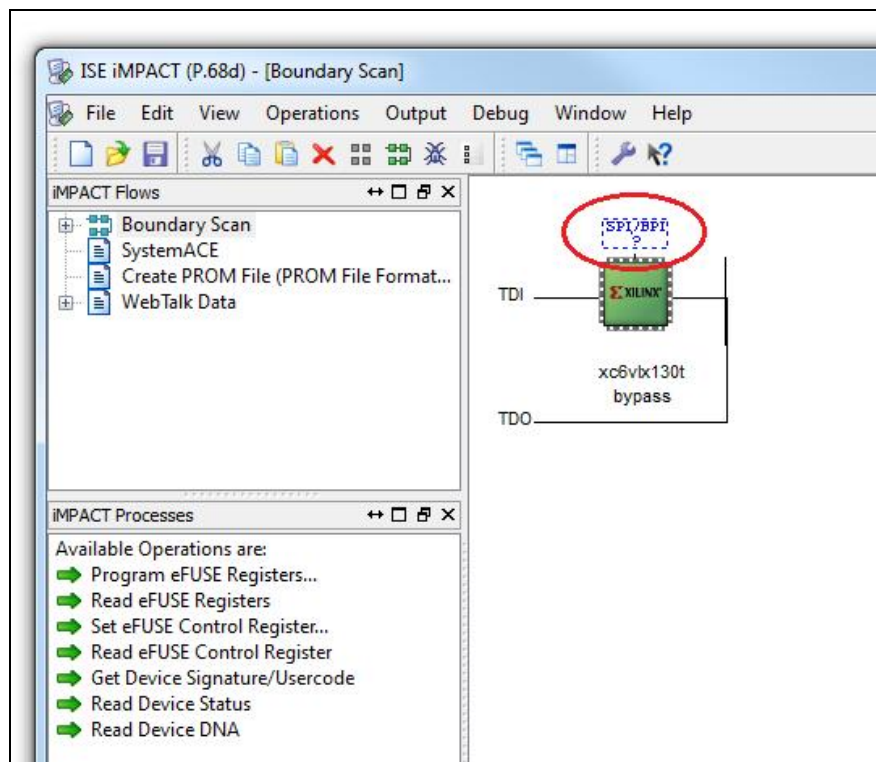
The screen shot below shows the parameters for the popup menu (SPI PROM, W25Q648V/CV, Data Width: 1):
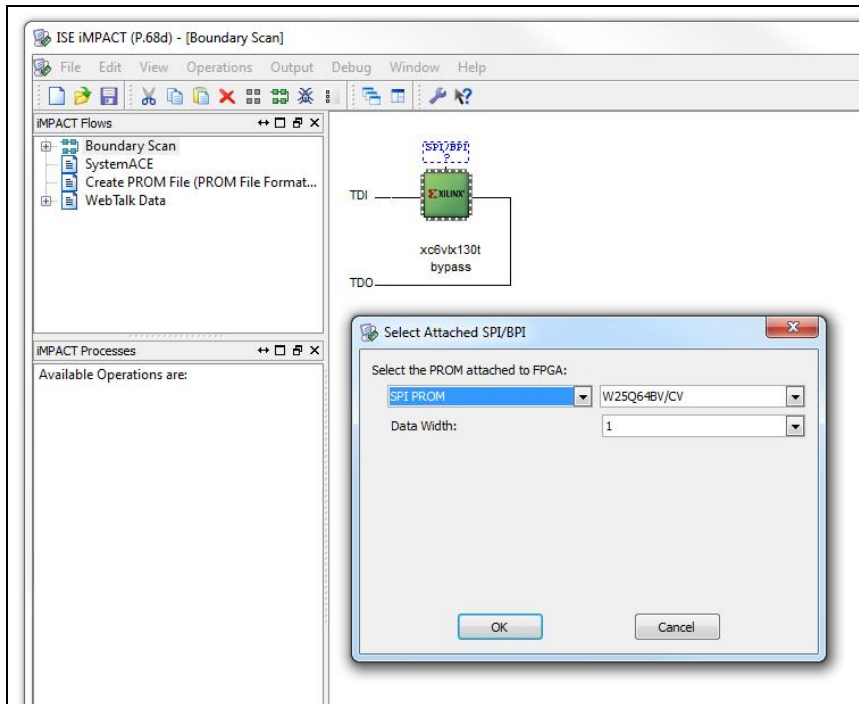
Finally the new firmware can be programmed into the select component (
Note: watchdog settings 3.2 (i.e. jumper J604 open).
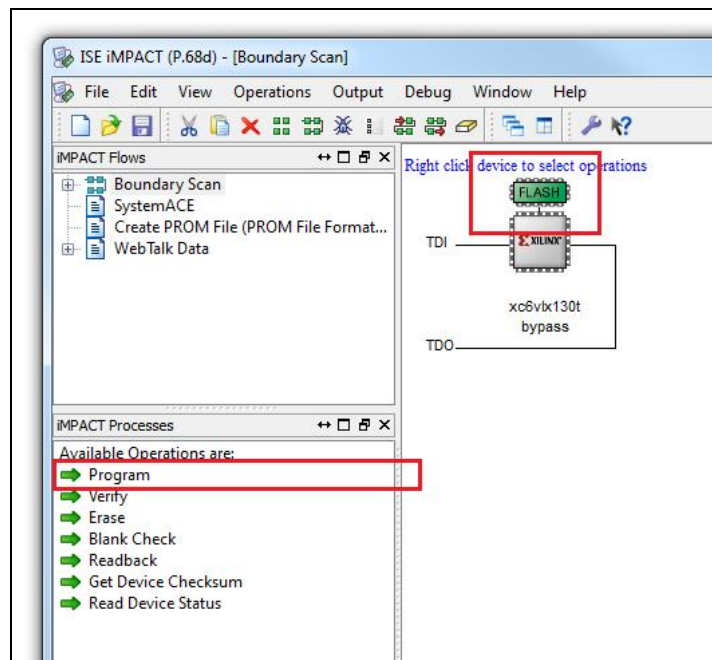
9.2   *Linux tool 'flashupdater'*

To be able to use this tool, it is required that the driver is loaded and the devices are mounted properly. Use the commands below for verification:

```
# lsmod | grep sis
```
Result (if driver loaded): `sis8300drv`

```
# ls /dev/ | grep sis
```
Result: for each installed device one entry (`sis8300-0` e.g.)

Now the update can be started:

```
./flashupdate <path/to/file>.bin
```

10 Appendix

10.1 Power Consumption

The currents drawn by the SIS8300-L are listed in the table below.

Voltage	Current
3,3 V	100 mA
12 V	4 A

These currents are typical values during normal operation. They can vary depending on the loaded firmware design.

10.2 Ordering options

The available part numbers are listed in the table below.

Struck part number	FPGA
04506	SIS8300-L with XC6VLX130T-2FFG1156C
05178	As above w/o FP SMAs and Harlink

Configuration options are:

AC coupling
DC with AD8138 preamplifiers
DC with AD8139 preamplifiers (lower noise)

Note: The DACs are routed to the front panel for part number 04506 and to the Zone 3 connector for part number 05178.

10.3 RTM/Zone 3 connectors J30 and J31

J30 and J31 are 90 pin right angle female connectors providing 30 contact pairs each (60 signal contacts and 30 ground contacts). Every contact pair is surrounded by a "L" shaped male shield blade. The shielding contact is designated with the names of the corresponding signal pair (signal pin a and b is affiliated with shielding contact ab e.g.). The picture below shows the connector contact layout as seen from the rear side of the board.
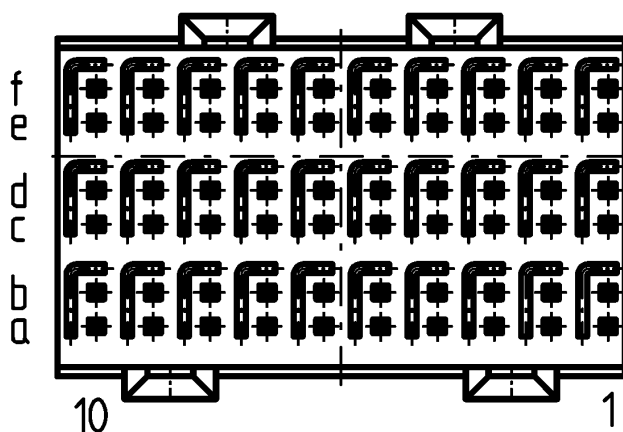
10.3.1 J31 connector pin assignments

The J31 connector routes the differential analog input signals of the ADC channels and ground to the RTM. The characters "TF" in signal names stand for signals to the AC coupled transformer input stages. In same fashion "PA" stands for DC coupled preamplifier input stage.

| Col → Row ↓	ef	f	e	cd	d	c	ab	b	a
10	GND	CH0_PA-	CH0_PA+	GND	GND	GND	GND	CH0_TF-	CH0_TF+
9	GND	CH1_TF-	CH1_TF+	GND	GND	GND	GND	CH1_PA-	CH1_PA+
8	GND	CH2_PA-	CH2_PA+	GND	GND	GND	GND	CH2_TF-	CH2_TF+
7	GND	CH3_TF-	CH3_TF+	GND	GND	GND	GND	CH3_PA-	CH3_PA+
6	GND	CH4_PA-	CH4_PA+	GND	GND	GND	GND	CH4_TF-	CH4_TF+
5	GND	CH5_TF-	CH5_TF+	GND	GND	GND	GND	CH5_PA-	CH5_PA+
4	GND	CH6_PA-	CH6_PA+	GND	GND	GND	GND	CH6_TF-	CH6_TF+
3	GND	CH7_TF-	CH7_TF+	GND	DAC1-	DAC1+	GND	CH7_PA-	CH7_PA+
2	GND	CH8_PA-	CH8_PA+	GND	GND	GND	GND	CH8_TF-	CH8_TF+
1	GND	CH9_TF-	CH9_TF+	GND	DAC2-	DAC2+	GND	CH9_PA-	CH9_PA+

10.3.2 J30 connector pin assignments

The J30 connector is used to route power, data and system management pins to the RTM board.

Col → Row ↓	ef	f	e	Cd	d	c	ab	b	a
10	GND	GND	GND	GND	GND	GND	GND	GND	GND
9	GND	CLK1-	CLK1+	GND	nc	nc	GND	CLK0-	CLK0+
8	GND	CLK5-	CLK5+	GND	CLK2-	CLK2+	GND	CLK4-	CLK4+
7	GND	GND	GND	GND	GND	GND	GND	GND	GND
6	GND	GPO1-	GPO1+	GND	GPO0-	GPO0+	GND	TCLK-	TCLK+
5	GND	D8-	D8+	GND	D7-	D7+	GND	D6-	D6+
4	GND	D5-	D5+	GND	D4-	D4+	GND	D3-	D3+
3	GND	SFP-TX-	SFP-TX+	GND	SFP-RX-	SFP-RX+	GND	SFP-CLK-	SFP-CLK+
2	GND	SCL_I	SCL_E	GND	SCL	MP+3.3V	GND	PWR+12V	PWR+12V
1	GND	SDA_I	SDA_E	GND	SDA	PS#	GND	PWR+12V	PWR+12V

10.3.3 Note on AC/DC input stage selection

The AC (transformer) or DC (operational amplifier OpAmp) input path is selected on the SIS8300-L card via 0603 solder bridges as illustrated for channels 0 and 1 on the screenshot below. The designators for all channels can be found in the table below.
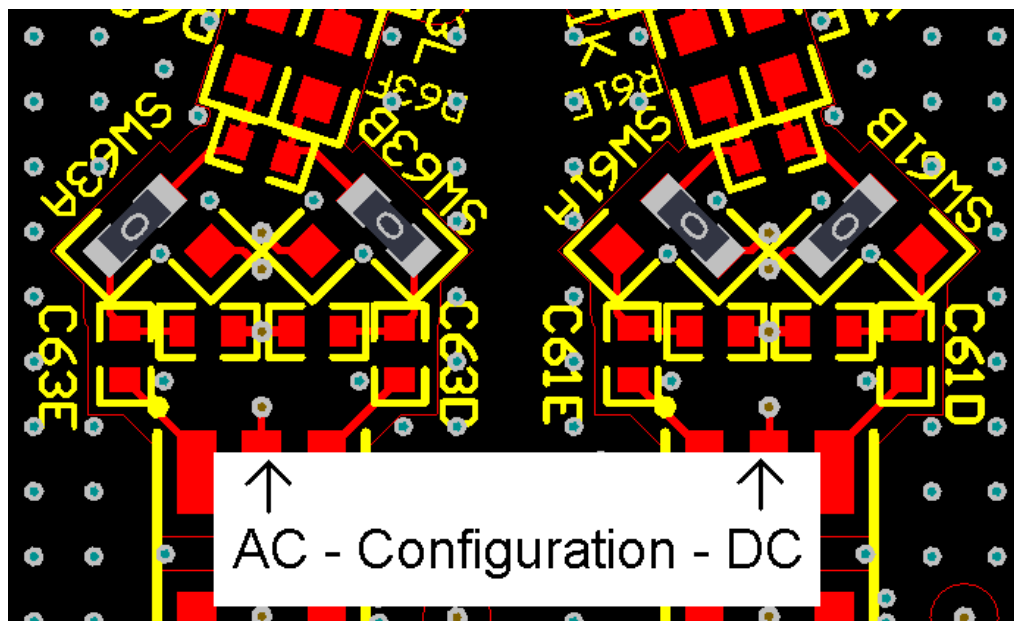
ADC Channel	Designator	Input for ADC
0	SW63A	ADC_CH0+
0	SW63B	ADC_CH0-
1	SW61A	ADC_CH1+
1	SW61B	ADC_CH1-
2	SW58A	ADC_CH2+
2	SW58B	ADC_CH2-
3	SW56A	ADC_CH3+
3	SW56B	ADC_CH3-
4	SW53A	ADC_CH4+
4	SW53B	ADC_CH4-
5	SW51A	ADC_CH5+
5	SW51B	ADC_CH5-
6	SW48A	ADC_CH6+
6	SW48B	ADC_CH6-
7	SW46A	ADC_CH7+
7	SW46B	ADC_CH7-
8	SW43A	ADC_CH8+
8	SW43B	ADC_CH8-
9	SW41A	ADC_CH9+
9	SW41B	ADC_CH9-

10.4 RTM connector schematics
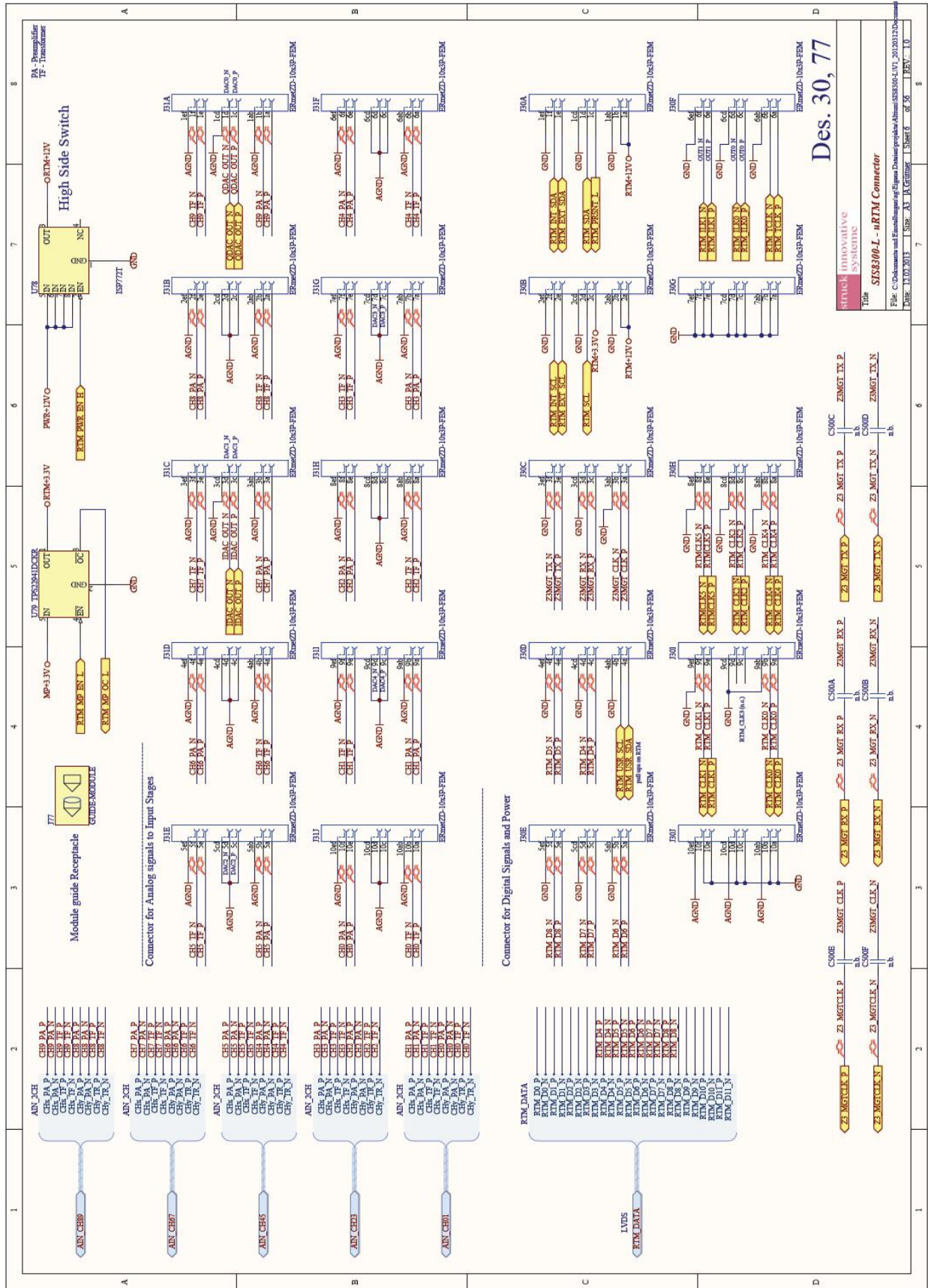
11 Index

12 V 70
3,3 V 70
8-bit 12
A 13
AC 72
AD9268 46
AD9510 39
ADC Sample Logic 17
Address Map 23
AdvancedMC 7
AMC 7
AMC.0 66
Appendix 70
arm 51
Atmega 16
Atmega128 12
Atmel 8, 12
AVR JTAG 12
binary 44
Block diagram 7
board layout 16
clock
   input 15
clock distribution 9, 10, 11
CON600 12
connector types 16
CRC 26
DAC 16
DAC
   range 45
DAC
   impedance 45
DAC 70
DC 72
DDR3 19
design 7
DESY 5
ECC 26
ERNI 16
external register interface 63
FIFO
   Link 1 23
   Link 2 23
firmware
   version 25
Firmware 17
Firmware Options
   register 28
FPGA 7
FPGA Firmware upgrade 67
front panel 14
FRU 66
functionality 7
Harlink 15, 70
HARTING 16
ICS853S057 37
IDT 37
iMPACT 12, 67
introduction 5
IPMI 8, 12
J32 8, 12
J604 12, 68
J75 71, 72

JTAG 12, 16, 23
   AVR 12
   over PCIe 26
   register 26
jumper 12
JYEBAO 16
L1 13, 15
L2 13, 15
LED 13
   20A 13
   A 13
   D105A 13
   D105B 13
   D110A 13
   D110B 13
   D20B 13
   D20C 13
   D20D 13
   D20E 13
   D20F 13
   D20G 13
   D20H 13
   D21D 13
   L1 13
   L2 13
   test 27
   U 13
   user 27
LEDs
   AMC 13
   Front Panel 13
   SMD 13
Link 1 23
Link 2 23
LVDS 15
M 48
MCH 12
memory buffer 22
memory histogram interface 21
memory interface 19
memory write interface 20
microcontroller 8, 12
MOLEX 16
MTCA.4 66
multiplexer A 38
multiplexer B 38
multiplexer C 38
multiplexer D 38
multiplexer E 38
NXP 66
ordering options 70
P 48
parallel load 12
PCF8574 66
PCI Express 7
PICMG 5
platform management 8
Power Consumption 70
register
   ADC Acquisition Control/Status 29
   ADC IOB delay 47
   ADC Sample Control 30
   ADC serial interface 46