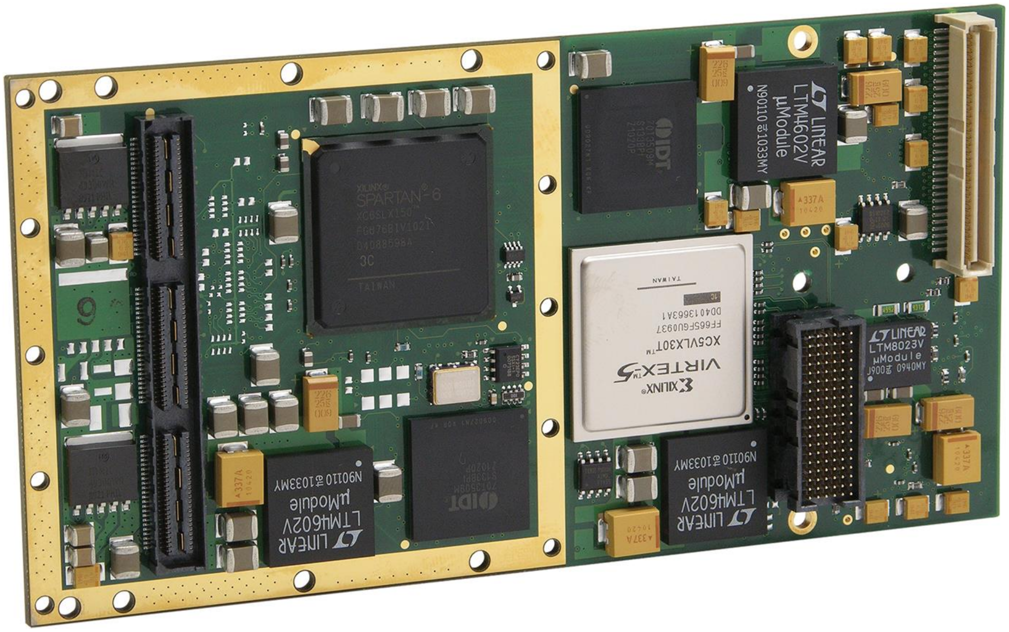
Series XMC-SLX150
Spartan 6 Based FPGA
XMC Module

USER'S MANUAL



ACROMAG INCORPORATED
30765 South Wixom Road
Wixom, MI 48393-2417 U.S.A.
solutions@acromag.com

Tel: (248) 295-0310
Fax: (248) 624-9234

Data and specifications are subject to change without notice.

8500-893C

TABLE OF CONTENTS

IMPORTANT SAFETY CONSIDERATIONS

You must consider the possible negative effects of power, wiring, component, sensor, or software failure in the design of any type of control or monitoring system. This is very important where property loss or human life is involved. It is important that you perform satisfactory overall system design and it is agreed between you and Acromag, that this is your responsibility.

The information of this manual may change without notice. Acromag makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Further, Acromag assumes no responsibility for any errors that may appear in this manual and makes no commitment to update, or keep current, the information contained in this manual. No part of this manual may be copied or reproduced in any form without the prior written consent of Acromag, Inc.

1.0 General Information

2.0 PREPARATION FOR USE

3.0 PROGRAMMING INFORMATION

4.0 THEORY OF OPERATION

5.0 SERVICE AND REPAIR

6.0 SPECIFICATIONS

DRAWINGS

Trademarks are the property of their respective owners.

RELATED PUBLICATIONS

The following manuals and part specifications provide the necessary information for in depth understanding of the AX board.

Spartan 6 Data Book	**http://www.xilinx.com**
IDT70T3519S Spec.	**http://www.idt.com**
IDT70T3509MS Spec.	**http://www.idt.com**
MT47H32M16CC Spec	**http://www.micron.com**
CY23EP05 Specification	**http://www.cypress.com**

1.0 GENERAL INFORMATION

The re-configurable XMC-SLX modules use the Xilinx Spartan 6 XC6SLX150 FPGA. Re-configuration of the FPGA is possible via a direct download into the Xilinx FPGA over the PCIe bus. In addition, on board flash memory can be loaded with FPGA configuration data for automatic Xilinx configuration on power-up. Flash programming is also implemented over the PCIe bus.

Acromag provides an example design that includes an interface to the user rear I/O and front I/O connectors. The example design also, includes an interface to the SRAM with DMA hardware support.

Table 1.1: *The XMC-SLX boards are available in both standard and extended temperature ranges with an option of memory*

MODEL	Dual Port SRAM Size	OPERATING TEMPERATURE RANGE
XMC-SLX150	256K x 64-bit	0°C to +70°C
XMC-SLX150E	256K x 64-bit	-40°C to +85°C
XMC-SLX150-1M	1M x 64-bit	0°C to +70°C
XMC-SLX150E-1M	1M x 64-bit	-40°C to +85°C

KEY FEATURES

- **Reconfigurable Xilinx FPGA** – In system configuration of the FPGA is performed through a flash configuration device or via the PCIe bus. This provides a means for creating custom user defined designs.

- **Dual-Port SRAM –** A 256K x 64-bit dual-port static random access memory (SRAM) is included. One port of the SRAM provides a direct link from the PCIe bus to the SRAM memory. The second port of the SRAM provides a direct link to the Xilinx user programmable FPGA. There is an order option (-1M) to increase the SRAM size to 1 Meg x 64 bit.

- **Interface to Front Multifunction Modules –** Various mezzanine modules ("AXM" model prefex), ordered separately, allow the user to select the Front I/O required for their application.

- **Interface to Rear P4 Connector –** The Spartan 6 FPGA is directly connected to 64 pins of the rear P4 connector. All 2.5volt IO standards supported by the Spartan 6 device are available. The example design provides low voltage differential signaling as 32 LVDS input/output signals.

- **Write Disable Jumper –** User configurable flash memory can be hardware write disabled by removal of an on board zero ohm surface mount resistor.

- **Example Design Provided –** The example VHDL design includes implementation of the Local bus interface, control of digital front and rear I/O, and SRAM read/write interface logic.

PCIe INTERFACE FEATURES

- **PCIe Bus -** A four lane PCI Express 1.1 operating at a bus speed of 2.5 Gbps per lane per direction is provided. This gives up to 2GBytes/sec data rate on the bus.

- **PCIe Bus Master –** The PCIe interface logic becomes the bus master to perform DMA transfers.

- **DMA Operation –** The PCIe bus interface supports two independent DMA channels capable of transferring data to and from the on board SRAM. The example design implements DMA block and demand modes of operation.

- **64, 32, 16, 8-bit I/O** - Register Read/Write is performed through data transfer cycles in the PCIe memory space. All registers can be accessed via 32, 16, or 8-bit data transfers. Access to Dual Port Memory can be accessed via 64, or 32-bit transfers.

- **Compatibility** – Complies with PCIe Base Specification Revision 1.1. Provides one multifunction interrupt. The XMC-VLX is compatible with XMC VITA 42.3 specification for P15.

ENGINEERING DESIGN KIT

Acromag provides an engineering design kit for the SLX boards (sold separately), a "must buy" for first time SLX module purchasers. The design kit (model XMC-SLX-EDK) provides the user with the basic information required to develop a custom FPGA program for download to the Xilinx user-programmable FPGA. The design kit includes a CD containing: schematics, parts list, part location drawing, example VHDL source, and other utility files. The SLX modules are intended for users fluent in the use of Xilinx FPGA design tools.

BOARD DLL CONTROL SOFTWARE

Acromag provides a software product (sold separately) to facilitate the development of Windows (2000/XP/Vista/7®) applications accessing Acromag PMC and XMC I/O board products, PCI and PCIe I/O Cards, and CompactPCI I/O Cards. This software (Model PCISW-API-WIN) consists of low-level drivers and Windows 32 Dynamic Link Libraries (DLLs) that are compatible with a number of programming environments including Visual C++™, Visual Basic .NET® and others. The DLL functions provide a high-level interface to boards eliminating the need to perform low-level reads/writes of registers, and the writing of interrupt handlers.

BOARD VxWORKS SOFTWARE

Acromag provides a software product (sold separately) consisting of board VxWorks® software. This software (Model PMCSW-API-VXW) is composed of VxWorks® (real time operating system) libraries for all Acromag PMC and XMC I/O board products, PCI and PCIe I/O Cards, and CompactPCI I/O Cards. The software is implemented as a library of "C" functions which link with existing user code to make possible simple control of all Acromag PCI and PCIe boards.

BOARD QNX SOFTWARE

Acromag provides a software product (sold separately) consisting of board QNX® software. This software (Model PMCSW-API-QNX) is composed of QNX® (real time operating system) libraries for all Acromag PMC and XMC I/O board products, PCI and PCIe I/O cards, and CompactPCI I/O cards. The software supports X86 PCI bus only and is implemented as library of "C" functions which link with existing user code to make possible simple control of all Acromag PCI and PCIe boards.

BOARD Linux SOFTWARE

Acromag provides a software product consisting of board Linux® software. This software (Model PMCSW-API-LNX) is composed of Linux® libraries for all Acromag PMC and XMC I/O board products, PCI and PCIe I/O cards, and CompactPCI I/O cards. The software supports X86 PCI bus only and is implemented as library of "C" functions which link with existing user code to make possible simple control of all Acromag PCI and PCIe boards.

2.0 PREPARATION FOR USE

UNPACKING AND INSPECTION

Upon receipt of this product, inspect the shipping carton for evidence of mishandling during transit. If the shipping carton is badly damaged or water stained, request that the carrier's agent be present when the carton is opened. If the carrier's agent is absent when the carton is opened and the contents of the carton are damaged, keep the carton and packing material for the agent's inspection.

For repairs to a product damaged in shipment, refer to the Acromag Service Policy to obtain return instructions. It is suggested that salvageable shipping cartons and packing material be saved for future use in the event the product must be shipped.

CAUTION
SENSITIVE ELECTRONIC DEVICES
DO NOT SHIP OR STORE NEAR STRONG ELECTROSTATIC, ELECTROMAGNETIC, MAGNETIC OR RADIOACTIVE FIELDS

WARNING: This board utilizes static sensitive components and should only be handled at a static-safe workstation.

This board is physically protected with packing material and electrically protected with an anti-static bag during shipment. However, it is recommended that the board be visually inspected for evidence of mishandling prior to applying power.

CARD CAGE CONSIDERATIONS

Refer to the specifications for loading and power requirements. Be sure that the system power supplies are able to accommodate the power requirements of the system boards, plus the installed Acromag board, within the voltage tolerances specified.

IMPORTANT: *Adequate air circulation or conduction cooling must be provided to prevent a temperature rise above the maximum operating temperature.*

In an air cooled assembly, adequate air circulation must be provided to prevent a temperature rise above the maximum operating temperature and to prolong the life of the electronics. If the installation is in an industrial environment and the board is exposed to environmental air, careful consideration should be given to air-filtering.

In a conduction cooled assembly, adequate thermal conduction must be provided to prevent a temperature rise above the maximum operating temperature.

BOARD CONFIGURATION

Remove power from the system before installing board, cables, termination panels, and field wiring.

Default Hardware Configuration

The board may be configured differently, depending on the application. When the board is shipped from the factory, it is configured as follows:

- The on board flash memory device is read/write enabled.
- The default configuration of the programmable software control register bits at power-up are described in section 3.
- The control registers must be programmed to the desired configuration before starting data input or output operation.

Front Panel Field I/O Connector

The front panel connector provides the field I/O interface connections via optional mezzanine I/O modules, purchased separately.

Rear P4 Field I/O Connector

The rear I/O P4 connector connects directly to the user-programmable FPGA. The VCCO pins are powered by 2.5 volts and thus will support the 2.5 volt IOStandards. The IOSTANDARD attribute can be set in the user constraints file (UCF). For example, rear I/O can be defined for LVCMOS25 (low voltage CMOS). The example design defines the rear I/O to LVDS_25 (Low-Voltage Differential Signaling) in the user constraints file. Additional information regarding the 2.5V IOStandards is available in the Spartan 6 SelectIO Users Guide available from Xilinx.

The example design defines the rear I/O connector with 32 LVDS I/O pairs. The LVDS pairs are arranged in the same row in table 2.1. For example, RIO0_P and RIO0_N form a signal pair. The P identifies the Positive input while the N identifies the Negative input.

Table 2.1: *Board Rear Field I/O Pin Connections*

The example design implements 2.5volt LVDS I/O to the rear connector. Signal pairs are routed to pins (1,3), (2,4) etc.

Ch.	Positive Pin Description	Pin	Negative Pin Description	Pin
0	RIO0_P	1	RIO0_N	3
1	RIO1_P	2	RIO1_N	4
2	RIO2_P	5	RIO2_N	7
3	RIO3_P	6	RIO3_N	8
4	RIO4_P	9	RIO4_N	11
5	RIO5_P	10	RIO5_N	12
6	RIO6_P	13	RIO6_N	15
7	RIO7_P	14	RIO7_N	16
8	RIO8_P	17	RIO8_N	19
9	RIO9_P	18	RIO9_N	20
10	RIO10_P	21	RIO10_N	23
11	RIO11_P	22	RIO11_N	24
12	RIO12_P	25	RIO12_N	27
13	RIO13_P	26	RIO13_N	28
14	RIO14_P	29	RIO14_N	31
15	RIO15_P	30	RIO15_N	32
16	RIO16_P	33	RIO16_N	35
17	RIO17_P	34	RIO17_N	36
18	RIO18_P	37	RIO18_N	39
19	RIO19_P	38	RIO19_N	40
20	RIO20_P	41	RIO20_N	43
21	RIO21_P	42	RIO21_N	44
22	RIO22_P	45	RIO22_N	47
23	RIO23_P	46	RIO23_N	48
24	RIO24_P	49	RIO24_N	51
25	RIO25_P	50	RIO25_N	52
26	RIO26_P	53	RIO26_N	55
27	RIO27_P	54	RIO27_N	56
28	RIO28_P	57	RIO28_N	59
29	RIO29_P	58	RIO29_N	60
30	RIO30_P	61	RIO30_N	63
31	RIO31_P	62	RIO31_N	64

This connector is a 64-pin female receptacle header (AMP 120527-1 or equivalent) which mates to the male connector on the carrier/CPU board (AMP 120521-1 or equivalent).

Non-Isolation Considerations

The board is non-isolated, since there is electrical continuity between the logic and field I/O grounds. As such, the field I/O connections are not isolated from the system. Care should be taken in designing installations without isolation to avoid noise pickup and ground loops caused by multiple ground connections.

Standalone Operation

Powering the XMC-SLX150 as an independent board is possible using the J7 and J8 board through holes. As an independent standalone board the XMC- SLX150 board would not be plugged into a PCIe slot. To independently power the board, the required +12, VPWR5/12 volt, and +3.3 volt power supplies must be provided via the J7 and J8 contact holes. The holes have 29 mil openings with 60 mil pads. The location of the contact holes on the board are shown in diagram at the end of this manual.

Flash Write Disable Resistor

By default the flash memory (U6) is read/write enabled. Removal of resistor R172 disables writing the flash configuration device. Refer to Resistor Location Drawing at the end of this manual to identify the board location of R172.

3.0 PROGRAMMING INFORMATION

This Section provides the specific information necessary to program and operate the board.

GETTING STARTED

1. The XMC SLX board is shipped with the user-programmable Xilinx FPGA code stored in flash memory. Upon power-up the XMC SLX will automatically configure the FPGA with the example design code stored in flash. As a first step become familiar with the XMC SLX, as supplied by Acromag. The board will perform all the functions of the example design as described in this manual.

   The Example Design Memory Map section gives a description of the I/O operations performed by the example design. It will allow testing of digital I/O, interrupts, read/write of dual port SRAM, and testing of both DMA channels. It is strongly recommended that you become familiar with the board features by using the example design as provided by Acromag.

   Do not attempt to reconfigure the flash memory until after you have tested and become familiar with the XMC SLX as provided in the example design.

2. After you are familiar with the XMC SLX and have tested it using the example design, you can move on to step 2. Here you will modify the example design VHDL code slightly. It is recommended that you test this modified example design using the reconfiguration direct method. It is not recommended that the flash be overwritten until you have tested your code. The reconfigure direct method will allow programming of the FPGA directly from the PCIe bus. If for some reason the XMC SLX does not perform as expected, you can power the XMC SLX down. Upon power-up, the example design provided by Acromag will again be loaded into the FPGA.

   The document, Using the XMC SLX Engineering Design Kit, provided in the engineering design kit will guide you through the steps required to modify the example design for your custom application.

   See the Direct PCIe bus to Xilinx Configuration section for a description of the steps required to perform reconfiguration directly from the PCIe bus. The registers provided in the FPGA Programming Memory Map are used to implement a direct reconfiguration.

3. After you have thoroughly tested your customized FPGA design, you can erase the flash and write your code to flash. Once the flash is erased you will not be able to go back to the example design by simply powering down and restarting the board.

   See the Flash Configuration section for a description of the steps required to write new or reprogram of example design code to the flash device. The registers provided in the FPGA Programming Memory Map are used to implement a flash erase and reprogram operations.

PCIe CONFIGURATION ADDRESS SPACE

This board is a PCI Express Base Specification Revision 1.1 compliant PCIe bus board.

The PCIe bus is defined to address three distinct address spaces: I/O, memory, and configuration space. This board can be accessed via the PCIe bus memory, and configuration spaces.

The card’s configuration registers are initialized by system software at power-up to configure the card. The board is a Plug-and-Play PCIe card. As a Plug-and-Play card the board’s base addresses are not selected via jumpers but are assigned by system software upon power-up via the configuration registers. A PCIe bus configuration access is used to read/write the PCIe card’s configuration registers.

When the computer is first powered-up, the computer’s system configuration software scans the PCIe bus to determine what PCIe devices are present. The software also determines the configuration requirements of the PCIe card.

The system software accesses the configuration registers to determine how many blocks of memory space the module requires. It then programs the board’s configuration registers with the unique memory base address.

Since this board is relocatable and not fixed in address space, its device driver must use the mapping information stored in the board’s Configuration Space registers to determine where the board is mapped in memory space.

The memory maps in this chapter reflect byte accesses using the “Little Endian” byte ordering format. Little Endian uses even-byte addresses to store the low-order byte. Little-Endian means that the least significant byte is stored at the lowest memory address and the most significant byte is stored at the highest memory address. The Intel x86 family of microprocessors uses “Little Endian” byte ordering.

Big Endian is the convention used in the Motorola 68000 microprocessor family and is the VMEbus convention. In Big Endian, the lower-order byte is stored at odd-byte addresses. Big-endian means that the most significant byte is stored at the lowest memory address and the least significant byte is stored at the highest memory address.

	Low address	Layout of a 64-bit long int						High address
Little Endian	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Big Endian	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0

CONFIGURATION REGISTERS

The PCIe specification requires software driven initialization and configuration via the Configuration Address space. This board provides 512 bytes of configuration registers for this purpose. It contains the configuration registers, shown in Table 3.1, to facilitate Plug-and-Play compatibility.

The Configuration Registers are accessed via the Configuration Address and Data Ports. The most important Configuration Registers are the Base Address Registers and the Interrupt Line Register which must be read to determine the base address assigned to the board and the interrupt request line that goes active on a board interrupt request.

Table 3.1 *Configuration Registers*

Reg. Num.	**D31 D24**	**D23 D16**	**D15 D8**	**D7 D0**
0	**Device ID** 0x5807 (SLX150) 0x5808 (SLX150-1M)		**Vendor ID** 16D5	
1	Status		Command	
2	Class Code=118000			Rev ID=00
3	BIST	Header	Latency	Cache
4	64-bit Memory Base Address for Memory Accesses to PCIe interrupt, and DMA Registers, 4K Space**(BAR0)**			
6	64-bit Memory Base Address/32-bit Data to Spartan 6 User Registers, 4M Space**(BAR2)**			
8	64-bit Memory Base Address/64-bit Data to Dual Port Memory, 8M Space **(BAR4)**[1]			
10	Not Used			
11	**Subsystem ID** 0x5807 (SLX-150) 0x5808 (SLX150-1M)		**Subsystem Vendor ID** 16D5	
12	Not Used			
13,14	Reserved			
15	Max_Lat	Min_Gnt	Inter. Pin	Inter. Line

1. 8M is reserved for both the XMC-SLX150 and the XMC-SLX150-1M. The XMC-SLX150-1M uses the full address space. However, the XMC-SLX150 only uses 2M. Any operation beyond the 2M boundary is not defined and will result in improper board operation.

This board is allocated memory space address (BAR0) to access the PCIe interrupt, and DMA registers. The PCIe bus decodes 4K bytes for these memory space registers. .

This board is allocated a 4M byte block of memory (BAR2) that is addressable in the PCIe bus memory space. BAR2 space is used to access the board's flash configuration functions and the reprogrammable Spartan 6 FPGA registers.

In addition, this board is allocated (BAR4) memory of 8M bytes. The BAR4 memory is addressable in the PCIe bus memory space to access the board's Dual Port Memory transfers. The XMC-SLX150-1M utilizes the entire 8M space. However, the XMC-SLX150 only uses 2M. Read and/or write operations outside of the 2M limit on the XMC-SLX150 is not defined and will result in improper board operation.

BAR0 MEMORY MAP

Table 3.2: *BAR0 Registers*

Note that any registers/bits not mentioned will remain at the default value logic low.

BAR0 Base Addr+	Bit(s)	Description
00H	31:0	Interrupt Control/Status
04H	31:0	DMA Status/Abort Register
08H	31:0	Global Interrupt Enable (Bit-31)
0CH -> FFH	31:0	Reserved
100H	31:0	DMA Channel 0 System Address LSB
104H	31:0	DMA Channel 0 System Address MSB
108H	31:0	DMA Channel 0 Transfer Size in bytes
10CH	7:0	DMA Channel 0 Command
110H	31:0	DMA Channel 0 XMC Board Starting Address
114H	0	DMA Channel 0 Start DMA Transfer Bit
118H to 11CH		Reserved
120H	31:0	DMA Channel 1 System Address LSB
124H	31:0	DMA Channel 1 System Address MSB
128H	31:0	DMA Channel 1 Transfer Size in bytes
12CH	7:0	DMA Channel 1 Command
130H	31:0	DMA Channel 1 XMC Board Starting Address
134H	0	DMA Channel 1 Start DMA Transfer Bit
138H	7:0	Retraining Counter
13CH->FFFH	31:0	Reserved

The BAR0 registers are implemented in the PCIe bus interface chip and not the user programmable FPGA. As such, the user cannot change the logic functions implemented in BAR0. These registers are read/write registers that are software controlled. These registers provide interrupt control/status and DMA control/status. The Interrupt Control/Status is at

BAR0 base address plus 00H offset. The DMA registers are at BAR0 base address plus offset 100H to 134H. These registers control the transfer direction, size, system address, and XMC addresses for DMA channels 0 and 1.

The Dual Port SRAM control registers at BAR2 must also be used to set up a DMA Demand Mode transfer. The Demand mode transfer is initiated by driving signals DREQ0# or DREQ1# active. The SRAM control register method allows a DMA transfer to be initiated when an FPGA generated address counter is equal to the DMA Channel Threshold Register. That is, when the predetermined amount of data is available in the SRAM the hardware will automatically start a DMA transfer.

Interrupt Control/Status Register (Read/Write) - (BAR0 + 00H)

INTERRUPT REGISTER

This Interrupt Control/Status register at BAR0 base address + offset 00H is used to monitor and clear pending board interrupts. An interrupt can originate from the two DMA channels or U7, the user-programmable FPGA. All board interrupts are enabled or disabled via bit-31 of the Global Interrupt Enable register at BAR0 + 08H.

***Table 3.3:** Interrupt Control/Status Register When designing software drivers it is best to treat this register as two 16-bit registers. The upper 16-bits are Interrupt Control bits and the lower 16-bits are Interrupt Status.*

Bit(s)	FUNCTION	
0	This bit when set indicates a pending board interrupt. It reflects a pending interrupt from DMA channel 0 or DMA channel 1 or the U7 FPGA. It will reflect this status even if the Board Interrupt enable bit-31 is disabled.	
	0	No Interrupt Pending
	1	Interrupt Pending
1	DMA Channel 0 Interrupt Pending Status. Bit-16 must be set to logic high for this bit to go active. Write logic high to clear bit.	
	0	No Interrupt Pending
	1	Interrupt Pending
2	DMA Channel 1 Interrupt Pending Status. Bit-17 must be set to logic high for this bit to go active. Write logic high to clear bit.	
	0	No Interrupt Pending
	1	Interrupt Pending
3	U7 Programmable FGPA Interrupt Pending Status. Bit-18 must be set to logic high for this bit to go active.	
	0	No Interrupt Pending
	1	Interrupt Pending
4-15	Not Used (bits are read as logic "0")	
16	DMA Channel 0 Interrupt Enable	
	0	DMA Channel 0 Interrupt Disabled
	1	DMA Channel 0 Interrupt Enabled
17	DMA Channel 1 Interrupt Enable	
	0	DMA Channel 1 Interrupt Disabled
	1	DMA Channel 1 Interrupt Enabled
18	U7 Programmable FPGA Interrupt Enable	
	0	Interrupt Disabled
	1	Interrupt Enabled
19-31	Not Used (bits are read as logic "0")	

A board pending interrupt is identified via bit-0 of this register. Logic high on bit-0 indicates a board pending interrupt. Bit-0 indicates a pending interrupt as long as DMA Channel 0 or DMA Channel 1 or U7 Programmable FPGA interrupt pending status bits 1, 2, or 3 respectively, remain active.

A DMA channel 0 pending status can be cleared/released by writing logic high to bit-1, the interrupt pending status bit. Likewise, writing logic high to bit-2 of this register clears DMA channel 1 pending status. U7 Programmable FPGA interrupt Pending status will pass the interrupt status of U7 only when bit-18 is set to logic high.

The Software Reset and Status Register at BAR2 + 8000H can be read to identify the exact source of the Programmable Spartan-6 FPGA interrupt.

Bits 16 to 18 of this register are used to enable or disable interrupts from specific functions. This Interrupt register must have bits 16 and 17 set to logic high in order for DMA interrupts to occur on DMA channels 0 and 1, respectively. Bit-18 must be set to logic high to enable interrupts from U7, the programmable FPGA.

The mezzanine board interrupt enable bits must also be set if interrupts are to originate from the mezzanine board which are passed through the programmable FPGA to this register's pending status bits.

DMA BAR0 REGISTERS

DMA Status/Abort Register (Read/Write) - (BAR0 + 04H)

This DMA Status register at BAR0 base address plus 04H is used to DMA BAR0 REGISTERS

DMA transfers must start aligned to Double Lword boundary when performing 64-bit DMA transfers. identify a DMA transfer complete status and issue DMA channel abort.

The DMA complete status bit 0 or 1 will remain logic high until cleared by writing logic high back to the same bit. The start of a new DMA transfer (hardware initiated) will also clear a set Transfer Complete bit.

The DMA abort bits 8 and 9 corresponding to channels 0 and 1 respectively when set to logic high will abort the current DMA transfer. If asserted and the channel still has outstanding requests, all requests are handled before the transfer is aborted, otherwise the transfer is immediately aborted. If asserted, and the current transfer is a Completion, than a Completion with Completion Abort status is sent and the DMA transfer is stopped.

This register can be read or written via 32-bit transfers.

***Table 3.4** DMA Status Register*

Bit(s)	FUNCTION	
0	DMA Channel 0 Transfer Complete. This bit is cleared by write of logic high to this bit or start of a new DMA transfer.	
	0	Transfer not Complete
	1	Transfer Completed
1	DMA Channel 1 Transfer Complete. This bit is cleared by write of logic high to this bit or start of a new DMA transfer.	
	0	Transfer not Complete
	1	Transfer Completed
2-7	Not Used (bits are read as logic "0")	
8	DMA Channel 0 Abort on write of logic high to this bit.	
	0	No Action
	1	Abort Channel 0 DMA transfer
9	DMA Channel 1 Abort on write of logic high to this bit.	
	0	No Action
	1	Abort Channel 1 DMA transfer
10-15	Not Used (bits are read as logic "0")	
16-19	DMA Channel 0 or State Encoding	
	0000	Transfer Completed Successfully
	0001	Transfer stopped by backend
	0010	Transfer ended with CPL timeout
	0011	Transfer ended with CPL UR error
	0100	Transfer ended with Completion error
	1000	Channel is busy
	1001	Requesting transfer to PCIe Link
	1010	Waiting for Completion
	1011	Waiting for backend read or write data
	11XX	Busy State reserved
20-23	DMA Channel 1 State Encoding (see bit descriptions given for Channels 0 on bits 16-19)	
24-31	Not Used (bits are read as logic "0")	

GLOBAL INTERRUPT ENABLE

Global Interrupt Enable Bit-31 (Read/Write) -
(BAR0 + 08H)

This Global Interrupt Enable bit at BAR0 base address + offset 08H is used to enable all board interrupts. An interrupt can originate from the two DMA channels or U7, the programmable FPGA. All board interrupts are enabled when bit-31 is set to logic high. Likewise, board interrupts are disabled with bit-31 set to logic low. Bit-31 of this register can be read or written.

***Table 3.5** Global Interrupt Enable Bit*

Bit(s)	FUNCTION	
0-30	Not Used (bits are read as logic "0")	
31	XMC Board Interrupt Enable. This bit must be set to enable the PCIe bus interrupt signal to be driven active by the board.	
	0	Board Interrupts Disabled
	1	Board Interrupts Enabled

DMA System Starting Address LSB Registers (Read/Write) - (BAR0 + 100H and 120H)

This register contains the least significant 32 address bits of the DMA System Starting Address.

The DMA System Starting Address register meaning depends on the selected DMA mode (see bit-3 of DMA command register). For Direct DMA Mode this address register specifies the physical address of a contiguous memory buffer where data will be read/written. For scatter-gather DMA mode, this address register points to the first element of the chained-listed of page descriptors.

The DMA System Starting Address Register at BAR0 base address plus 100H (120H) is used to set the DMA channel 0 (1) data starting address. Writing to these registers is possible via 32-bit transfers.

DMA System Starting Address MSB Registers (Read/Write) - (BAR0 + 104H and 124H)

This register contains the most significant 32 address bits of the DMA System Starting Address.

The DMA System Starting Address register meaning depends on the selected DMA mode (see bit-3 of DMA command register). For Direct DMA Mode this address register specifies the physical address of a contiguous memory buffer where data will be read/written. For scatter-gather DMA mode, this address register points to the first element of the chained-listed of page descriptors.

The DMA System Starting Address Register at BAR0 base address plus 104H (124H) is used to set the DMA channel 0 (1) data starting address. Writing to these registers is possible via 32-bit transfers.

DMA Transfer Size Registers (Read/Write) - (BAR0 + 108H and 128H)

The DMA Transfer Size Register is used to set the size of the DMA transfer that moves data between system memory and the board's Dual Port SRAM. The transfer size indicates to total amount of data to transfer, in units of bytes.

The onboard static RAM for the XMC-SLX150 has 2-Megabyte maximum capacity. As such, the maximum value that should be written to this register is 200000. The XMC-SLX150-1M has an 8 Megabyte capacity and the maximum value that can be written is 800000. *WARNING: The XMC-SLX150 will allow for a larger value to be written than available SRAM size, which will cause undesirable operations. This address must be set to a DWORD boundary (multiple of 4) for proper operation.*

The DMA Transfer Size Register at BAR0 base address +108H (128H) is used to set the DMA channel 0 (1) data transfer size. Writing to these registers is possible via 32-bit data transfers.

DMA Command Registers (Read/Write) - (BAR0 + 10CH and 12CH)

The DMA Command register is used to set the priority, Scatter Gather enable, and to indicate the command to be used for the DMA transfer.

Memory Read Burst command is used to program a read transfer. The data is moved from system memory to the boards SRAM memory.

Memory Write Burst command is used to program a write transfer. The data moves from SRAM to system memory.

Writing to these registers is possible via 32-bit data transfers.

***Table 3.6** DMA Command Register*

DMA transfers must start aligned to Double Lword boundary when performing DMA transfers.

Bit(s)	FUNCTION	
0	Not Used (bit is read as logic '0')	
1	Not Used (bit is read as logic '0')	
2	Not Used (bit is read as logic '0')	
3	Scatter-gather: Setting this bit enables scatter-gather mode	
	0	Direct DMA Mode
	1	Enable Scatter-Gather Mode
7 to 4	DMA Command	
	'0110'	Memory Read Burst
	'0111'	Memory Write Burst
8 to 31	Not Used (bit is read as logic '0')	

In Scatter Gather DMA transfer mode, the DMA start address is a pointer to a chained list of page descriptors. Each descriptor contains the address and size of a data block (page), as well as the next descriptor block. The following table describes the 20 byte block Scatter Gather descriptor.

***Table 3.7:** Scatter Gather Descriptor Block*

Offset	Field	Description
00H	Page Address 31:3	The Start address of memory page that must be aligned on an 8-byte boundary (bits2:0 must be "000") If a page is located in 32-bit address space, then bits 63:32 must be set to 0. If a page is located in 64-bit address space, then the full 64-bit address must be initialized.
04H	Page Address 63:32	
08H	Page Size 31:0	The size of the memory page in units of bytes and a multiple of 8 bytes.
0CH	Next Descriptor 31:2 Bit 0 is End of Chain bit	The address of the next page descriptor, which must be aligned on a 4-byte boundary (bits 1:0 must be "00"). Setting bit 0 to logic 1 indicates that the current descriptor is the last descriptor in the chain.
10H	Next Descriptor 63:32	

DMA XMC Board Starting Address Registers (Read/Write) - (BAR0 + 110H and 130H)

The DMA XMC Board Starting Address register specifies the physical address of the board’s Dual Port SRAM memory where data will be read/written.

Data bits 22 down to 2 of this register are used to address the SRAM. Data bit-2 select the low 32-bit SRAM (U4) when logic low and selects the upper 32-bit SRAM (U19) when logic high. Data bits 22 down to 3 correspond to SRAM address lines 19 down to 0. Bits 19 and 18 are only used on the XMC-SLX150-1M model. For the standard model XMC-SLX150, bits 19 and 18 should always be zero.

The DMA XMC Board Starting Address register at BAR0 base address plus 110H (130H) is used to set the DMA channel 0 (1) data starting address. Writing to these registers is possible via 32-bit data transfers.

Start DMA Transfer (Write Only) - (BAR0 + 114H and 134H)

The Start DMA Transfer register is used to software start a DMA transfer. Set bit-0 to logic ‘1’ will start the corresponding channels DMA transfer.

The Start DMA Transfer register at BAR0 base address plus 114H (134H) is used to start the DMA channel 0 (1) transfer. Writing to these registers is possible via 32-bit data transfers.

BAR2 MEMORY MAP

The memory space address map used to program the FPGA and flash device is shown in Table 3.8. Note that the base address for the board (BAR2) in memory space must be added to the addresses shown to properly access these registers. Register accesses as 32, 16, and 8-bit transfers in memory space are permitted. All addresses in BAR2 from 0 to 7FFF hex are fixed and cannot be changed by the user via the programmable Spartan-6 FPGA.

Table 3.8
BAR2 Memory Map

BAR2 Addr+	**D31 D08**	**D07 D00**	**BAR2 Addr+**
0003	Not Used[1]	Configuration Status Register	**0000**
0007	Not Used[1]	Configuration Control Register	**0004**
000B	Not Used[1]	Configuration Data	**0008**
000F	Not Used[1]	Flash Status 1 Register	**000C**
0013	Not Used[1]	Flash Status 2 Register	**0010**
0017	Not Used[1]	Flash Read	**0014**
001B	Not Used[1]	Flash Reset	**0018**
001F	Not Used[1]	Flash Start Write	**001C**
0023	Not Used[1]	Flash Erase Sector	**0020**
0027	Not Used[1]	Flash Erase Chip	**0024**
002B	Not Used[1]	Flash Data Register	**0028**
002F	Not Used[1]	Flash Address 7->0	**002C**
0033	Not Used[1]	Flash Address 15->8	**0030**
0037	Not Used[1]	Flash Address 23->16	**0034**
003B	PCIe bus FPGA System Monitor Status/Control Register		**0038**
003F	PCIe bus FPGA System Monitor Address Register		**003C**
0043	Not Used[1]		**0040**
↓	Not Used[1]		↓
7FFF	Not Used[1]		**7FFC**

[1] The board will return 0 for all addresses that are "Not Used".

EXAMPLE DESIGN MEMORY MAP

Table 3.8: Example Design BAR2 Memory Map

1. The board will return 0 for all addresses that are "Not Used".

BAR2 Addr+	D31 D16	D15 D00	BAR2 Addr+
8003	Software Reset and Status Register		**8000**
8007 ↓ **802B**	Mezzanine Module Memory Space		**8004** ↓ **8028**
802F	Rear I/O Connector Read Register		**802C**
8033	Rear I/O Connector Write Register		**8030**
8037	DMA Control Register		**8034**
803B	FPGA Port SRAM Register Data Lines 31 to 0		**8038**
803F	FPGA Port SRAM Register Data Lines 63 to 32		**803C**
8043	FPGA Port-SRAM Control Register		**8040**
8047	FPGA Port-SRAM Address Register		**8044**
804B	DMA Channel 0 Threshold Register (DP-SRAM)		**8048**
804F	DMA Channel 1 Threshold Register (DP-SRAM)		**804C**
8053	Address Reset Register 0 (DP-SRAM)		**8050**
8057	Address Reset Register 1 (DP-SRAM)		**8054**
805B	XMC Board Identification Code: "A3" for Acromag Example Design		**8058**
805F ↓ **808F**	Not Used[1]		**805C** ↓ **808C**
8093	Not Used[1]	Air Temperature Register	**8090**
8093 ↓ **3FFFFF**	Additional Mezzanine Module Space 8100-> 8137 Otherwise Not Used[1]		**8090** ↓ **3FFFFC**

This memory map reflects byte accesses using the "Little Endian" byte ordering format. Little Endian uses even-byte addresses to store the low-order byte. The Intel x86 family of microprocessors uses "Little Endian" byte ordering. In Big Endian, the lower-order byte is stored at odd-byte addresses.

Flash Configuration

The SLX150 board uses a flash configuration device to store programming information for the Xilinx FPGA. The flash configuration device and FPGA are hardwired together so that during power-up the contents of the configuration device are downloaded to the FPGA. The flash configuration data can be reprogrammed using the PCIe bus interface. The following is the general procedure for reprogramming the flash memory and reconfiguration of the Xilinx FPGA:

1) Disable auto-configuration by setting bit-0 (Stop Configuration) of the Configuration Control register to logic high.
2) Clear the Xilinx FPGA of its previous configuration by setting the Configuration Control register bit-2 to logic high. Software must also keep bit-0 set to a logic high.
3) Read INIT as logic high (Bit-1 of Configuration Status register) before programming is initiated.
4) Verify that the Flash Chip is not busy by reading bit-7, of the Flash Status 2 register at base address plus 10H, as logic 0 before starting a new Flash operation.
5) Erase the current flash contents by using the Flash Erase Sector method. Flash erase sectors are implemented by setting bit-0 of the Flash Erase Sector register to logic high. There are 128 flash sectors, which are addressed via the most significant seven flash address lines. The most significant seven flash address lines are set via the Flash Address 23-17 register at base address plus 34H. Issuing a Flash Erase Sector command will erase the contents of the flash chip only in the sector specified.
6) Verify that the Flash Chip is not busy by reading bit-7, of the Flash Status 2 register at base address plus 10H, as logic 0 before going to the next step.
7) Download the Configuration file to the flash configuration chip via the PCIe bus.
   i) Write the byte to be sent to the Flash Data register at base address plus 28H.
   ii) Write the address of the Flash Chip to receive the new data byte to the Flash Address registers at base address plus 2CH, 30H, and 34H. Issuing a Flash Start Write will automatically increment this address after the prior Flash Write has been completed. Thus, the address will not need to be set prior to issuing the next Flash Start Write. The first byte of the configuration file should be written to address 0 of the Flash Chip. The Flash Start Write operation will take 9μ seconds to complete.
   iii) Issue a Flash Start Write command to the Flash Chip by writing logic 1 to bit-0 of base address plus 1CH.
   iv) Verify that the Flash Chip is not busy by reading bit-7 as logic 0 of the Flash Status 2 register at base address plus 10H before going back to step i to write the next byte.
8) Enable auto-configuration by setting bit-0 (Stop Configuration) of the Configuration Control register to logic low.
9) Verify that the configuration is complete by reading DONE (bit-0 of Configuration Status Register) as logic high.
10) Thereafter, at power-up the configuration file will automatically be loaded into the FPGA.

Direct PCIe bus to Xilinx Configuration

Configuration of the Xilinx FPGA can be implemented directly from the PCIe bus. The following is the general procedure for re-configuration of the Xilinx FPGA via the PCIe bus:

1) Disable auto-configuration by setting bit-0 (Stop Configuration) of the Configuration Control register to logic high.
2) Clear the Xilinx FPGA of its previous configuration by setting the Configuration Control register bit-2 to logic high.
3) Read INIT as logic high (Bit-1 of Configuration Status register) before programming is initiated.
4) Download the Configuration file directly to the Xilinx FPGA by writing to the Configuration Data register. The entire configuration file must be written to the Xilinx FPGA one byte at a time to the Configuration Data register at base address plus 08H.
5) Verify that the configuration is complete by reading DONE (bit-0 of Configuration Status Register) as logic high. DONE is expected to be logic high immediately after the last byte of the configuration file is written to the Xilinx FPGA.
6) At each power-up the configuration file will need to be reloaded into the FPGA.

CONFIGURATION CONTROL REGISTERS

Configuration Status Register (Read Only) – (BAR2 + 0000H)

This read only register reflects the status of configuration complete and Xilinx configuration clear bits. This Configuration Status register is read at base address plus 0H.

Table 3.9: *Configuration Status Register*

Bit(s)	FUNCTION	
0	DONE:	
	0	Xilinx FPGA is not configured
	1	Xilinx FPGA configuration is complete
1	INIT:	
	0	INIT is held low until the Xilinx is clear of its current configuration
	1	INIT transitions high when the clearing of the current Xilinx configuration is complete
2 to 7	Not Used (bits are read as logic “0”)	

Configuration Control (Read/Write) – (BAR2 + 04H)

CONFIGURATION CONTROL REGISTERS

This read/write register is used to stop Xilinx configuration, and clear Xilinx configuration memory. This Configuration Control register is accessed at base address plus 04H.

__Table 3.10:__ Configuration Control Register

Bit(s)	FUNCTION	
0	Stop Xilinx Configuration:[1]	
	0	Enable Xilinx FPGA configuration
	1	Stop Xilinx FPGA configuration (This bit should be set to logic high until after the Flash device is written with valid program data).
1	Not Used (bit is read as logic "0")	
2	Clear Current Xilinx Configuration:	
	0	Logic low has no effect.
	1	Logic high resets the Xilinx configuration logic. Re-configuration can begin after INIT transitions high.
3 to 7	Not Used (bits are read as logic "0")	

1. __WARNING__ Spartan 6 FPGA Configuration MUST BE STOPPED (set bit 0 to logic high) prior to any JTAG communication. Failure to do so will cause JTAG errors.

Configuration Data (Write Only) – (BAR2 + 08H)

This write only register is used to write Xilinx configuration data directly to the Xilinx FPGA from the PCIe bus. The Configuration Data register is accessed at base address plus 08H. The entire configuration file must be written to the Xilinx FPGA one byte at a time. Configuration complete is verified by reading DONE (bit-0 of the Configuration Status Register) as logic high.

A write to the Configuration Data register while auto-configuration from Flash is active will cause the Xilinx configuration to fail. Auto-configuration is stopped by writing logic 1 to bit-0 of the Configuration Control register at base address plus 04H.

The Xilinx FPGA should also be cleared of its current configuration prior to loading of a new configuration file. The FPGA is cleared of its current configuration by writing logic 1 to bit-2 at address plus 04H.

Flash Status 1 (Read Only) – (BAR2 + 0CH)

FLASH CONTROL REGISTERS

This read only register is used to read the DQ5 status of the flash chip. The Flash Status 1 register is at base address plus 0CH.

__Table 3.11:__ Flash Status 1 Register

Bit(s)	FUNCTION	
0 to 4	Not Used (bits are read as logic "1 or 0")	
5	DQ5:	
	0	Chip enabled for reading array data.
	1	The system must issue the Flash Reset command to re-enable the device for reading array data if DQ5 goes high. DQ5 will go high during a Flash Start Write, Flash Erase Chip, or Flash Erase Sector operation.
6 and 7	Not Used (bits are read as logic "1 or 0")	

FLASH CONTROL REGISTERS

Flash Status 2 (Read Only) – (BAR2 + 10H)

This read only register is used to read the ready or busy status of the flash chip. The Flash Status 2 register is at base address plus 10H. The system must first verify that that Flash Chip is not busy before executing a new Flash command. The Flash Chip is busy if bit-7 of this register is set to logic 1. The Flash will always be Busy while bit-0 of the Configuration Control register is set to logic "0".

Table 3.12: *Flash Status 2 Register*

Bit(s)	FUNCTION	
0 to 6	Not Used (bits are read as logic "0")	
7	Busy / Ready~ Set bit-0 of the Configuration Control register to logic "1" before monitoring this busy bit.	
	0	Flash Chip is Ready
	1	Flash Chip is Busy

Flash Read (Read Only) – (BAR2 + 14H)

A Flash Read command is executed by reading this register at base address plus 14H. Prior to issue of a Flash Read the Flash Address registers must be set with the desired address to be read. See the Flash Address registers at base address plus 2CH, 30H, and 34H.

The system must issue the Flash Reset command to re-enable the device for reading array data if DQ5 goes high. DQ5 can go high during a Flash Start Write, Flash Erase Chip, or Flash Erase Sector operation. DQ5 can be monitored via the Flash Status 1 register at base address plus 0CH.

Flash Reset (Write Only) – (BAR2 + 18H)

This write only register is used to initiate a reset of the flash chip. A Flash Reset command is executed by writing logic 1 to bit-0 of this register at base address plus 18H. Writing the flash reset command resets the chip to reading data mode. Flash reset can be useful when busy is held active.

Flash Start Write (Write Only) – (BAR2 + 1CH)

This write only register is used to initiate the write of a byte to the flash chip. A Flash Start Write command is executed by writing logic 1 to bit-0 of this register at base address plus 1CH. Prior to issuing of a Flash Start Write the Flash Data and Address registers must be set with the desired data and address to be written. See the Flash Data and Address registers at base address plus 28H, 2CH, 30H, and 34H.

Issuing of a Flash Start Write will automatically increment this address after the previously issued Flash Write has completed. Thus, the address will not need to be set prior to issuing of the next Flash Start Write if consecutive addresses are to be written.

Flash Erase Sector (Write Only) – (BAR2 + 20H)

FLASH CONTROL REGISTERS

A Flash Erase Sector command is executed by writing logic 1 to bit-0 of this register at base address plus 20H. Verify that the Flash Chip is not busy from a previous operation before beginning a new operation. This is accomplished by reading bit-7, of the Flash Status 2 register, as logic 0.

There are 128 flash sectors, which are addressed via the most significant seven flash address lines. The most significant seven flash address lines are set via the Flash Address 23-17 register at base address plus 34H. Issuing of a Flash Erase Sector command will erase the contents of the flash chip only in the sector specified.

A flash bit cannot be programmed from logic 0 to logic 1. Only an erase chip operation can convert logic 0 back to logic 1. **Prior to reprogramming of the flash chip a Flash Erase Chip or Flash Erase Sector command must be performed.**

The system can determine the status of the erase operation by reading the Flash Ready/Busy status. Bit-7 of the Flash Status 2 register, at base address plus 10H, will read as logic 0 when chip erase is completed.

Any other flash commands written to the flash chip during execution of the flash erase sector operation are ignored. Note that a hardware reset during the erase sector operation will immediately terminate the operation.

Flash Erase Chip (Write Only) – (BAR2 + 24H)

This write only register is used to erase the entire contents of the flash chip. A flash bit cannot be programmed from logic 0 to logic 1. Only an erase chip operation can convert logic 0 back to logic 1. **Prior to reprogramming of the flash chip a Flash Erase Chip command must be performed.**

A Flash Erase Chip command is executed by writing logic 1 to bit-0 of this register at base address plus 24H. Verify that the Flash Chip is not busy from a previous operation before beginning a new operation. This is accomplished by reading bit-7, of the Flash Status 2 register, as logic 0.

The system can determine the status of the erase operation by reading the Flash Ready/Busy status. Bit-7 of the Flash Status 2 register, at base address plus 10H, will read as logic 0 when chip erase is completed.

Any other flash commands written to the flash chip during execution of the flash erase chip operation will be ignored. Note that a hardware reset during the chip erase operation will immediately terminate the operation.

Flash Data Register (Read/Write) – (BAR2 + 28H)

This read/write register holds the data byte which is sent to the flash chip upon issuing of a Flash Start Write command.

Although only the least significant 8 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfers.

FLASH REGISTERS

Flash Address 7->0 (Read/Write) – (BAR2 + 2CH)

This read/write register holds the least significant byte of the address to which the flash chip is written upon issue of a Flash Start Write command.

Although only the least significant 8 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Flash Address 15->8 (Read/Write) – (BAR2 + 30H)

This read/write register sets bits 15 to 8 of the address to which the flash chip is written upon issue of a Flash Start Write command.

Although only the least significant 8 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Flash Address 23->16 (Read/Write) – (BAR2 + 34H)

This read/write register sets bits 23 to 16 of the address to which the flash chip is written upon issue of a Flash Start Write command.

Reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfer.

SYSTEM MONITOR REGISTERS U5 PCIe bus

This system monitor is within the Virtex 5 LX30T that controls the PCIe Interface.

System Monitor Status/Control Register (Read/Write) – (BAR2 + 38H)

This read/write register will access the system monitor register at the address set in the System Monitor Address Register.

For example, the address of the System Monitor Status register that is to be accessed is first set via the System Monitor Address register at BAR2 plus 3CH. Next, this register at BAR2 plus 38H is read. Bits 22 to 16 of this register hold the address of system monitor register that is accessed. Data bits 15 to 6 of this register hold the temperature, Vccint, or Vccaux value. Data bits 5 to 0 are not used. Valid addresses are given in column one of Table 3.12.

Reading or writing this register is possible via 32-bit data transfers.

The 10-bits digitized and output from the ADC can be converted to temperature by using the following equation.

$$Temperature(°C) = \frac{ADCcode \times 503.975}{1024} - 273.15$$

The 10-bits digitized and output from the ADC can be converted to voltage by using the following equation.

$$SupplyVoltage(volts) = \frac{ADCcode}{1024} \times 3V$$

System Monitor Address Register (Write Only) – (BAR2 + 3CH)

This write only register is used to set the system monitor address register with a valid address for the System Monitor internal status or control registers. Valid addresses are given in the following table. Additional addresses can be found in the Xilinx System Monitor document UG192 (available from Xilinx). Writing this register is possible via 32-bit data transfers.

The address value written to this register can be read on bits 22 to 16 of the System Monitor Status/Control register at BAR2 plus 38H.

Table 3.13: *System Monitor Register Map*

Address	Status Register
00h	Temperature
01h	Vccint
02h	Vccaux
20h	Maximum Temperature
21h	Maximum Vccint
22h	Maximum Vccaux
24h	Minimum Temperature
25h	Minimum Vccint
26h	Minimum Vccaux

BAR2 U7 FPGA REGISTERS

Software Reset and Status Register (Read/Write) – (BAR2 + 8000H)

This read/write register is used to Software reset the board, monitor the status of board interrupts, and select the on board active clock.

Bits 0 to 7 of this register are used to monitor the interrupt pending status of interrupts originating from the front mezzanine module.

USERo CLOCK CONTROL

Note USERo selects the Local bus clock.

Bit 8 of this register controls the USERo signal. The USERo control signal is used to select between the 125MHz clock and the user defined clock (PLL_CLK). The user defined clock is defined in the example code of the FPGA and output on signal PLL_CLK. The Digital Clock Manager of the FPGA offers a wide range of clock management features including clock multiplication and division for generation of a user defined clock (PLL_CLK). A 125MHz crystal generated clock signal (FPGA_CLK_PLL) is input to the FPGA for use in generation of the user-defined clock signal PLL_CLK. The PLL_CLK can be a minimum of 62.5MHz and a maximum of 125MHz. Since the PLL_CLK signal is generated and driven by the FPGA, it will only be available after the FPGA is configured. See the example VHDL file included in the engineering design kit and the Xilinx documentation on the Digital Clock Manager for more information.

The USERo signal is controlled via a bit-8 of the Software Reset and Status Register at BAR2 plus 8000H. The USERo control bit-8 at by default is set to a logic low to select the PLL_CLK clock as the board clock frequency. Bit-8 set to logic high will select the 125MHz clock as the board clock frequency.

Bits 15 to 13 of this register will read “001” for all Acromag digital I/O mezzanine modules. These bits will read “010” when the AXM-A30 high speed analog input mezzanine module is present.

Bits 27 and 28 are DMA acknowledgement bits and will read a logic high while the corresponding DMA channel transfer is active.

Bit-29 indicates the completion of initialization and calibration of the DDR2 controller.

Bit 31 of this register when set to a logic “1” will issue a reset signal to the FPGA hardware.

Reading or writing to this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Table 3.14: *Software Reset and Status Register*

1. *All bits labeled “Not Used” will return logic “0” when read.*

BIT	FUNCTION	
7-0	Mezzanine interrupt status is identified via data bits 0 to 7. Read of a “1” indicates that an interrupt is pending for the corresponding data bit. A pending interrupt will remain active until disabled via the mezzanine interrupt control registers.	
	Logic “0”	Interrupt Not Pending
	Logic “1”	Interrupt Pending
8	USERo Control	
	Logic “0”	Board clock = PLL_CLK (Default)
	Logic “1”	Board clock 125MHz
12-9[1]	Not Used[1]	
15-13	Mezzanine Identification Code: “001” for all Acromag digital I/O mezzanine boards “010” for the AXM-A30 mezzanine board	
26-16	Not Used[1]	
27	DACK0 Status Logic high is a valid acknowledgement for DMA channel 0	
28	DACK1 Status Logic high is a valid acknowledgement for DMA channel 1	
29-30	Not Used[1]	
Bit-31	The most significant bit of this register when set to a logic “1” will issue a software reset.	
	Logic “0”	No Operation
	Logic “1”	Software reset issued to Xilinx user-programmable FPGA

Rear I/O Connector Read Register (Read Only) - (BAR2 + 802CH)

REAR Read REGISTER

The Rear I/O Connector Read Register is used to read the LVDS input status of 16 channels. This example design has 16 channels, identified in Table 3.14, programmed as LVDS input only channels. Table 2.1 shows each channel and it's corresponding P4 connector pin assignment.

This Rear I/O Connector Read register is a read only register and writing to this register has no effect on the LVDS input channels. Reading from this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Rear I/O Connector Write Register (Read/Write) - (BAR2 + 8030H)

REAR Write REGISTER

The Rear I/O Connector Write Register is used to set 16 LVDS output channels. This example design has 16 channels, identified in Table 3.11, fixed as LVDS output only channels. Table 2.1 shows the P4 connector pins and their corresponding channel identifiers.

This Rear I/O Connector Write register is written to set the LVDS output channels and can also be read to verify the output channel settings. Reading from this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Write/Read Data Register Bit	Rear Connector Write Output Channels	Rear Connector Read Input Channels
0	1	0
1	2	3
2	4	5
3	6	7
4	8	9
5	10	11
6	12	13
7	14	15
8	16	17
9	18	19
10	20	21
11	22	23
12	24	25
13	26	27
14	28	29
15	30	31
16-31	Not Used[1]	Not Used[1]

Table 3.15: *Rear I/O Registers*

Column 1 identifies the write data bit that drives the output channel listed in column 2. Column 1 also identifies the read data bit that returns the input channel listed in column 3. For example data bit 0 drives output channel 1 when written and returns channel 0 register setting when read.

1. *All bits labeled "Not Used" will return logic "0" when read.*

DMA REGISTERS AT BAR2

DMA Control Register (Read/Write) - (BAR2 + 8034H)

The DMA Control Register is used to request a DMA Demand mode transfer. The hardware signals **DREQ0 and DREQ1** are driven active by software setting of bits 0 or 1 of this register to request the DMA transfer. The transfer must include the Static RAM Memory as either the source or the destination.

For software triggered DMA, bit-0 is used to request a DMA channel 0 transfer while bit-1 is used to request a channel 1 transfer. The bit must be set to logic high to request a transfer. Once set, the bit will remain asserted until the DMA transfer has started. If both bits are set simultaneously, the channel 0 DMA transfer will be implemented first followed by channel 1.

In a user application a data ready condition, such as a memory buffer full condition, can be physically tied (via logic in the FPGA) to the DREQ0 or DREQ1 FPGA signals to cause the DMA transfer to start.

Note that the Spartan 6 FPGA has input signals DACK0_n and DACK1_n that are active during a DMA transfer. These signals are not accessible via a register, but can be used in custom firmware.

DUAL PORT SRAM REGISTERS

FPGA- SRAM Data Register (Read/Write) – (BAR2 + 8038H and 803CH)

The FPGA-SRAM Data Read Register is provided to access the SRAM port that links directly to the user-programmable Spartan-6 FPGA. Reading or writing BAR2 + 8038H accesses the SRAM least significant data lines 31 to 0. Reading or writing BAR2+ 803CH accesses the most significant SRAM data lines 63 to 32. Reading or writing these registers is only possible in 32-bit transfers. The address for the SRAM read or write is initialized by the Dual Port SRAM Internal Address register at BAR2 + 8044H. With each additional read or write to BAR2+ 803CH the address is automatically incremented.

Writing the SRAM would proceed by first setting the Address register at BAR2 + 8044H. Next the least significant 32-bit data word is written to BAR2 + 8038H. Finally, after the most significant 32-bit data word is written at BAR2+ 803CH the address is automatically incremented.

FPGA-Port SRAM Control Register (Read/Write) – (BAR2 + 8040H)

This read/write register is used to control the Dual-Port SRAM including enabling write, automatic DMA transfer and automatic address reset on DMA thresholds.

The default power-up state of this register is logic low. A reset will set all bits in this register to "0". Reading or writing to this register is possible via 32-bit, 16-bit or 8-bit data transfers.

***Table 3.16:** FPGA-Port SRAM Control Register*

BIT	FUNCTION	
0	This bit controls the vhdl signal SRAM_ENABLE. This signal must be set to logic high to enable writes to SRAM from the FPGA. The SRAM Internal Address register must also be set with the start address at which the data begins filling the SRAM.	
	Logic "0"	Disable Write and Enable Read
	Logic "1"	Enable Write and Disable Read
1	If enabled via this bit a DMA channel 0 request will be issued when the internal address counter is equal to the DMA Channel 0 Threshold Register. This will have the same effect as writing a 1 to bit 0 of the DMA Control Register at BAR2 plus 8034H. See Synchronous DP-SRAM in Section 4.0 for further details on using this feature.[2]	
	Logic "0"	Disable Auto DMA Request Channel 0
	Logic "1"	Enable Auto DMA Request Channel 0
2	If enabled via this bit a DMA Channel 1 request will be issued when the internal address counter is equal to the DMA Channel 1 Threshold Register. This will have the same effect as writing a 1 to bit 1 of the DMA Control Register at BAR2 plus 8034H.[2]	
	Logic "0"	Disable Auto DMA Request Channel 1
	Logic "1"	Enable Auto DMA Request Channel 1
3	If enabled via this bit the Internal Address Counter will be loaded with the value in Address Reset Register 0 when the counter is equal to the DMA Channel 0 Threshold Register. See the Address Reset Register description for further details. DMA does not have to be enabled to use this feature.[3]	
	Logic "0"	Disable Add. Reset on DMA Ch. 0 Threshold
	Logic "1"	Enable Add. Reset on DMA Ch. 0 Threshold
4	If enabled via this bit the Internal Address Counter will be loaded with the value in Address Reset Register 1 when the counter is equal to the DMA Channel 1 Threshold Register. See the Address Reset Register description for further details. DMA does not have to be enabled to use this feature.[3]	
	Logic "0"	Disable Add. Reset on DMA Ch. 1 Threshold
	Logic "1"	Enable Add. Reset on DMA Ch. 1 Threshold
5-15	Not Used[1]	

1. *Bits are not used and will return logic "0" when read.*

2. *All DMA transfer settings in the DMA Registers at BAR0 should be set prior to enabling automatic DMA transfers.*

3. *WARNING: Before enabling Address Reset on DMA Thresholds (bits 3 & 4), verify that the "DMA Ch. 0 Threshold Register" is not equal to the "Address Reset Register 0" and the "DMA Ch. 1 Threshold Register" is not equal to the "Address Reset Register 1." If these registers are equal and automatic reset is enabled an infinite loop will be created within the internal logic of the FPGA.*

DP-SRAM REGISTERS

Warning: To guarantee functionality disable DP-SRAM write cycles (via bit 0 of the DP-SRAM Control Registers) before writing to the DP-SRAM Internal Address Register.

FPGA-Port SRAM Internal Address Register (Read/Write) – (BAR2 + 8044H)

The FPGA-Port SRAM Internal Address Register is used to view and set the internal SRAM address. The FPGA will only write using 64-bit data transfers allowing for 3FFFF hex unique memory accesses for the base XMC-SLX150 model and FFFFF hex unique addresses for the XMC-SLX150-1M model. **Reading** this register will return the internal SRAM address. *Due to delays during data processing and the PCIe transfer the actual internal address may be slightly greater than the address read.* **Writing** to this register will set the Internal SRAM Address to the provided value. Bits 0 to 19 of this register are used. Writing logic '1' to bit 31 of this register or a system reset will cause the Internal SRAM Address to reset to "00000H" (the start of the SRAM memory). **Reading or writing to this register is possible via 32-bit data transfers, only.**

The SRAM Internal Address will automatically be incremented upon a write or read of the most significant SRAM Data Port at BAR2+ 803CH.

Table 3.17: *FPGA-Port SRAM Internal Address Register*

FPGA-Port SRAM Internal Address Register		
D31	**D30-D20**	**D19-D0[1]**
SRAM Internal Address Reset	Not Used (Read as logic '0')	SRAM Internal Address

1. Bits 18 and 19 are only used in the XMC-SLX150-1M model. They read as logic '0' in the XMC-SLX150 base model.

Note: An SRAM DMA Request will occur only after a data write cycle to the address defined by the DMA Threshold Registers.

FPGA-Port SRAM DMA Channel 0/1 Threshold Registers (Read/Write) – (BAR2 + 8048H/ 804CH)

The FPGA-Port SRAM DMA Channel 0/1 Threshold Registers are used to initiate an automatic DMA transfer. When the internal address counter is equal to the value in the DMA Channel 0 Threshold Register, a Channel 0 DMA request will be initiated. Similarly, when the internal address counter is equal the value in the DMA Channel 1 Threshold Register and there is valid data at that address, a Channel 1 DMA request will be initiated. This feature must be enabled via bits 1 and 2 (for Channels 0 & 1, respectively) of the FPGA-SRAM Control Register. Note that DMA settings must be set prior to the initiated transfer on both the BAR0 and BAR2 registers. A DMA transfer in progress is indicated via bits 0 and 1, for DMA Channels 0 and 1, respectively, in the DMA Control Register. See the DMA Registers section of this manual for further details. **Reading** of the Threshold register will return the corresponding DMA Threshold. **Writing** the Threshold registers will set the corresponding DMA Threshold to the provided value. Bits 0 to 17 of this register are used in the XMC-SLX150. Bits 0 to 19 of this registers are used in the XMC-SLX150-1M. **Reading or writing to this register is possible via 32-bit data transfers only.**

The default and reset values of these registers is shown in Table 3.18.

FPGA-Port SRAM DMA Channel 0/1 Threshold Registers Reset Values			
Register	**D30-D20**	**D19-D0**[1]	
		SLX150	SLX150-1M
DMA Channel 0 Threshold Reg.	Not Used (Read as logic '0')	1FFFF	7FFFFH
DMA Channel 1 Threshold Reg.	Not Used (Read as logic '0')	3FFFF	FFFFFH

Table 3.18: *Dual-Port DMA Threshold Registers Reset Values*

1.Bits 18 and 19 are Not Used in the XMC-SLX150.

DP-SRAM REGISTERS

FPGA-Port SRAM Address Reset Registers 0/1 (Read/Write) – (BAR2 + 8050H/ 8054H)

The FPGA-Port SRAM Address Reset Registers are used to reset the internal address counter to a user-defined value immediately upon reaching the DMA Threshold value. For example, after an SRAM write cycle where the internal address counter is equal to the value defined in the DMA Channel 0 Threshold Register, the internal address counter will then be loaded with the value defined in the Address Reset Register 0. Similarly, after a SRAM write cycle where the internal address counter is equal to the value defined in the DMA Channel 1 Threshold Register, the internal address counter will then be loaded with the value defined in the Address Reset Register 1. This allows for the internal address counter to be changed without any interruption in the transfer of data from the front connector input to the DP-SRAM. This feature must be enabled via bits 3 and 4 (for Channel 0 & 1 thresholds, respectively) of the FPGA Port-SRAM Control Register. Note that the DMA transfers *do not* have to be enabled for this feature to function. **Reading** of either register will return the corresponding internal address reset value. **Writing** this register will set the corresponding internal address reset register to the provided value. Bits 0 to 17 of this register are used in the XMC-SLX150. Bits 0 to 19 of this registers are used in the XMC-SLX150-1M. The most significant bits are not used and will return logic '0' when read. A system reset will cause these registers to reset to "00000H".

WARNING: The "DMA Ch. 0 Threshold Register" must not equal the "Address Reset Register 0" and the "DMA Ch. 1 Threshold Register" must not equal the "Address Reset Register 1." If these registers are equal and the address reset is enabled via the FPGA Port-SRAM Control Register an infinite loop will be created within the internal logic of the FPGA.

ID Code REGISTER

XMC Board Identification Code Register (Read Only) - (BAR2 + 8058H)

The XMC Board Identification Code register at BAR2 plus 8058H stores an ID code that can used to uniquely identify the XMC Spartan 6 card. This register will read A3 hex as provided by the Acromag example design. The user can change the hardware setting of this register in the programmable FPGA code. This ID code can be used to properly assign software drivers to multiple XMC boards that may have the same device and vender ID in a given system.

Reading from this register is possible via 32-bit, 16-bit or 8-bit data transfers.

SYSTEM MONITOR REGISTERS U7 FPGA

Temperature Data Register (Read only) – (BAR2 + 8090H)

This read only register provides the data from the external temperature monitor on the XMC-SLX. This register is automatically updated every second. Data bits 12 to 0 of this register hold the temperature information.

Reading this register is possible via 16-bit or 32-bit data transfers.

The 12-bits digitized output from the temperature sensor reads from -55°C minimum to 150°C maximum. The number is stored in two's compliment format with each bit equal to 0.0625°C. Refer to table 3.19 for data values.

Table 3.19: *Temperature Data Conversions*

External Temperature Data	
D12-D0	**Temperature**
0 0000 0000 0000	0°C
0 0000 0000 0001	0.06285°C
0 1001 0110 0000	150°C (Max)
1 1111 1111 1111	-0.06285°C
1 1110 0111 0000	-25°C
1 1100 1001 0000	-55°C (Min)

STATIC RAM MEMORY

DUAL PORT MEMORY

A 256K x 64-bit Dual Port synchronous SRAM (DP-SRAM) is provided on the XMC-SLX150 board. A 1Meg x 64-bit Dual Port synchronous SRAM is provided on the XMC-SLX150-1M board. One port of the SRAM connects directly to the local bus to allow for PCIe access. The remaining port connects directly with the user-programmable FPGA. This design allows for the user to maximize data throughput between the Field I/O’s and the controlling processor.

There are two automatic DMA initiators available that will trigger upon a user set threshold. Furthermore, upon a DMA transfer, the internal counter can be reset to a user specified value. See DMA Registers for more information on these operations. These features can be individually controlled through the SRAM Control Registers.

BAR4 MEMORY MAP

Static RAM Memory (Read/Write) – (BAR4 + 000000H to 7FFFFFH)

The Static RAM memory space is used to provide read or write access to on board SRAM memory. This memory space allows access to the SRAM from the port on the PCIe bus side of the SRAM. The Static RAM device has a 1Meg x 64-bit capacity on the XMC-SLX150-1M. The Static RAM device on the base model XMC-SLX150 has only a 256K x 64-bit capacity and only uses a portion of this memory space. Reading or writing to this memory space using DMA access is also only possible via 64-bit transfers.

The FPGA-Port SRAM Register at BAR2 + 8038H and 803CH are provided for testing the SRAM port that links directly to the user-programmable Spartan-6 FPGA.

4.0 THEORY OF OPERATION

This section contains information regarding the design of the board. A description of the basic functionality of the circuitry used on the board is also provided. Refer to the Block Diagram shown in Drawing 4502-131 as you review this material.

PCIe INTERFACE LOGIC

The PCIe bus interface logic on this board provides a 2.5Gbps interface to the carrier/CPU board per PCI Express Specification 1.1. The interface to the carrier/CPU board allows complete control of all board functions.

The PCIe bus endpoint interface logic is contained within a user interface FPGA. This logic includes support for PCIe commands, including: configuration read/write, and memory read/write. In addition, the PCIe interface requester and or completion accesses. Payload of up to 256 bytes is supported.

The logic also implements interrupt requests via message signaled interrupts. Messages are used to assert and de-assert virtual interrupt lines on the link to emulate the Legacy PCI interrupt INTA# signal.

The board performs DMA transfers on channels 0 and 1. The DMA transfers can be started via software or hardware. Hardware signal DREQ0# driven active by the programmable FPGA will start a DMA channel 0 transfer. Hardware signal DREQ1# driven active will start a DMA channel 1 transfer. To identify the pins corresponding to these signals, see the user constraints (.UCF) file provided in the engineering design kit. The DACK0# and DACK1# signals will go active upon the start of a DMA transfer and remain active until its completion. The example device driver software (purchased separately) can be used to exercise DMA block (software) and demand (hardware) modes of operation.

SYNCHRONOUS Dual-Port SRAM

A 256K x 64-bit synchronous Dual-Port SRAM is provided on the base model XMC-SLX150. The XMC-SLX150-1M model increase the DPSRAM size to 1M x 64-bit. One port of the SRAM interfaces to the PCIe bus interface chip, the Xilinx Virtex-5 LX30T device (U5). The other port connects directly to the programmable FPGA (U7). This configuration allows for a continuous data flow from the field inputs through the FPGA to the SRAM and then to the PCIe bus. Both ports of the SRAM operate in Pipeline mode. This allows for faster operational speed but does cause a one-cycle delay during read operations. The pins corresponding to the control signals, address, and data buses are in the user constraint (.UCF) file provided in the engineering design kit.

The SRAM port connected directly to the user-programmable FPGA (U7) supports continuous writes or single cycle reads. The SRAM port connected to the PCIe bus, through U5, supports reading and writing using a burst, double word (4 byte), or DMA transfers. For double word (4byte) accesses address and control signals are applied to the SRAM during one clock cycle, and either a write will occur on the next cycle or a read in two clock cycles. DMA accesses operate using the continuous burst method for maximum data throughput. The control signal, starting address, and data (if writing) are applied to the SRAM during one clock cycle. Then, during a write DMA transfer, new data is applied to the bus every subsequent clock cycle until the transfer is complete.

During DMA transfers the address is held constant and incremented internally in the Dual-Port SRAM. Please refer to the IDT70T3509SM Data Sheet (See Related Publications) for more detailed information.

THEORY OF OPERATION CONTINUED

The local bus interface between the PCIe bus interface chip (U5) and the user-programmable FPGA (U7) consists of the following signals.

The Local Address bus (LA) bits 21 to 2 are used to decode the 4M byte address space allocated by the PCIe bus to BAR2. Also, LA(26) bit-26 of the local address bus is logic high when the PCIe bus is performing an access to BAR2 address space.

Local Bus Signals

LBE0_n, LBE1_n, LBE2_n and LBE3_n are the local bus byte enables. LBE0_n when logic low indicates that the least significant byte on data lines D7 to D0 is selected for the read or write transfer. Likewise LBE3_n when logic low indicates that the most significant byte on data lines D31 to D24 is selected for the read or write transfer.

The Local Data (LD) bus bits 31 to 0 are bi-directional signals used for both read and write data transfers.

ADS_n, the address data strobe signal, will pulse low for one local bus clock cycle at the start of a new read or write access. The ADS_n signal is driven by the PCIe bus interface chip (U5).

Readyn must be driven low, on read or write cycle, by the programmable FPGA (U7) and held low until RdyAck_n is driven low by the PCIe bus interface chip (U5). This is shown on the read and write diagrams that follow.

Rdyack_n is driven low by the PCIe bus FPGA (U5) to signify the end of the read or write cycle.

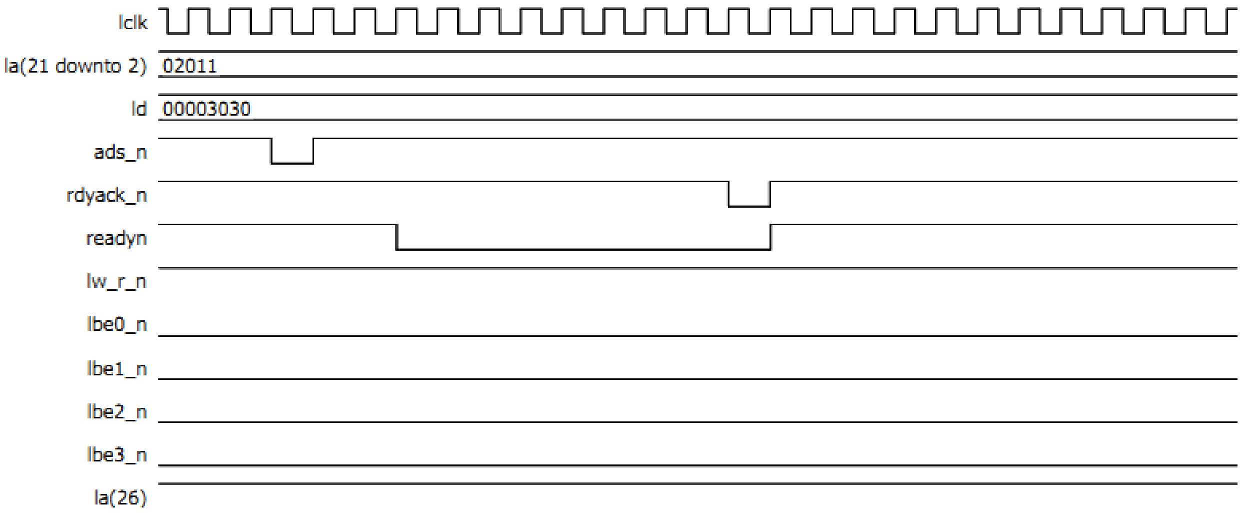
The LW_R_n signal, when logic high, indicates a write transfer in which data is moving from the PCIe bus to the reprogrammable FPGA (U7). This signal, when logic low, indicates a read transfer in which data is moving from the reprogrammable FPGA (U7) to the PCIe bus.

For a read cycle the LD data signals should be driven active prior to driving Readyn active. LD and Readyn should be held active until Rdyack_n is detected active.

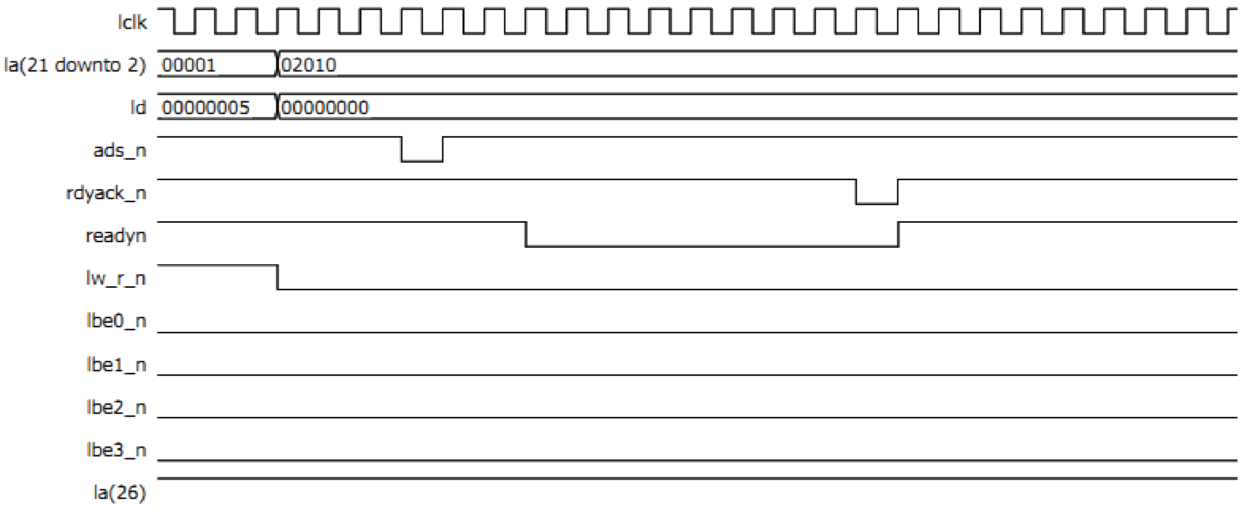
The LA, LBE0_n, LBE1_n, LBE2_n, LBE3_n, LD, LW_R_n signals are guaranteed setup at least 2 (LCLK/FPGA_CLK) clock cycles before signal ADS_n goes active. These signals are also guaranteed to be held active a minimum of 2 clock cycles after Readyn goes active. Also Rdyack_n will not pulse active sooner than 2 (LCLK/FPGA_CLK) clock cycles after Readyn goes active.

Both LCLK and FPGA_CLK are driven by the same zero delay Cypress clock driver. Signal ADS_n is driven by an OLogic register using the LCK at the bus interface FPGA. ADS_n is received at the programmable FPGA using FPGA_CLK at an ILogic register. This interface is considered synchronous to the LCLK/FPGA_CLK by Acromag.

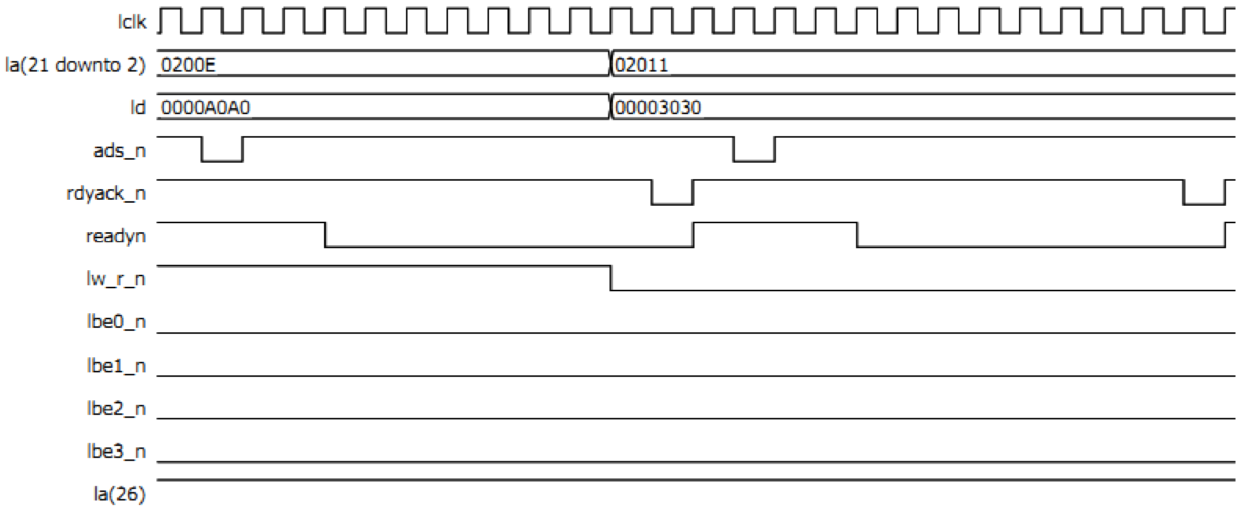
Local Bus Write Cycle Diagram



Local Bus Read Cycle Diagram



Local Bus Write and Read Cycle Diagram



Local Bus CLOCK CONTROL

Clk, the local bus clock, as seen in the following timing diagrams can be one of two sources. By default clk is a Digital Clock Manager (DCM) generated clock frequency. Clk can also be selected directly from the board 125MHz frequency.

The Local bus clk signal is controlled by USERo. The board clock is routed to the Dual Port SRAM and user-programmable FPGA (U7) using a low skew clock driver (Cypress CY23EP05). The on board 125MHz crystal oscillator is input to the user-programmable FPGA via signal FPGA_CLK_PLL. After the user-programmable FPGA (U7) is configured, an FPGA DCM generated clock signal (PLL_CLK) is selected as the board clock (the default condition). By setting bit-8 of the Software Reset and Status register, at BAR2 plus 8000H, to a logic high the 125MHz clock may be selected as the board clock. By setting bit-8 to a logic low the PLL_CLK becomes the board clock frequency. The default state of bit-8 is logic low.

5.0 SERVICE AND REPAIR

Surface-Mounted Technology (SMT) boards are generally difficult to repair. It is highly recommended that a non-functioning board be returned to Acromag for repair. The board can be easily damaged unless special SMT repair and service tools are used. Further, Acromag has automated test equipment that thoroughly checks the performance of each board. When a board is first produced and when any repair is made, it is tested, placed in a burn-in room at elevated temperature, and retested before shipment.

SERVICE AND REPAIR ASSISTANCE

Please refer to Acromag's Service Policy Bulletin or contact Acromag for complete details on how to obtain parts and repair.

PRELIMINARY SERVICE PROCEDURE

Before beginning repair, be sure that all of the procedures in Section 2, Preparation For Use, have been followed. Also, refer to the documentation of your carrier/CPU board to verify that it is correctly configured. Replacement of the board with one that is known to work correctly is a good technique to isolate a faulty board.

CAUTION: POWER MUST BE TURNED OFF BEFORE REMOVING OR INSERTING BOARDS

WHERE TO GET HELP

If you continue to have problems, your next step should be to visit the Acromag worldwide web site at http://www.acromag.com. Our web site contains the most up-to-date product and software information.

Acromag's application engineers can also be contacted directly via telephone or via email for technical assistance through the number or email address listed at the bottom of this page. When needed, complete repair services are also available.

6.0 SPECIFICATIONS

Single XMC Board

PHYSICAL

Height	13.5 mm (0.531 in)
Stacking Height	10.0 mm (0.394 in)
Depth	149.0 mm (5.866 in)
Width	74.0 mm (2.913 in)
Board Thickness	2.21 mm (0.08 in)

Unit Weight

XMC-SLX: 3.59oz (0.1016Kg), typical

Connectors

- **P15** XMC.3 144-pin Samtec ASP-103614-05 connector Complies with ANSI/VITA 42.3-2006
- **P14** Rear I/O Interface: One 64-pin female receptacle header (AMP 120527-1 or equivalent). This connector provide 64 rear I/O connections.

- Front Field I/O Connector on XMC module (Samtec QSS-075-01-L-D-A) Mating Mezzanine Connector (Samtec QTS-075-01-L-D-A) with 5mm stack height or (Samtec QTS-075-02-L-D-A) with 8mm stack height.

Table 6.1: Power Requirements for Example Design. Power will vary dependent on the application.

5V Maximum rise time of 100m seconds

Power Requirements		**XMC Modules**
VPWR[1] 12V (±5%)	Typical	820mA
	Max.	984mA
3.3V (±5%)	Typical	700mA
	Max.	840mA

1. VPRW may alternative be 5V (±5%). A 5V supply will consume approximately the same wattage as a 12V source.

On Board 1.2V Power to Spartan 6 FPGA	**Current Rating** (Maximum available for the user-programmable FPGA)
1.2V (±5%)	4A Maximum

ENVIRONMENTAL

Operating Temperature: 0 to +70°C. –40°C to +85°C (E Version)

Conduction Cooled XMC mezzanine card: Complies with ANSI/VITA 20-2001 (R2005). The XMC SLX, without a faceplate, is fully compatible with a conduction cooled host card.

Relative Humidity: 5-95% Non-Condensing.

Storage Temperature: -55°C to 125°C.

Non-Isolated: Logic and field commons have a direct electrical connection.

ENVIRONMENTAL

Designed to comply with EMC Directive 2004/108/EC Class B.

Radiated Field Immunity (RFI): Complies with IEC 61000-4-3 with no register upsets.

Conducted R F Immunity (CRFI): Complies with IEC 61000-4-6 with no register upsets.

Surge Immunity: Not required for signal I/O per IEC 61000-4-5.

Electric Fast Transient (EFT) Immunity: Complies with IEC 61000-4-4 Level 2 (0.5KV at field I/O terminals).

Electrostatic Discharge (ESD) Immunity: Complies with EN61000-4-2 Level 3 (8KV enclosure port air discharge) Level 2 (4KV enclosure port contact discharge).

Radiated Emissions: Meets or exceeds European Norm 61000-6-3:2007 for class B equipment. Shielded cable with I/O connections in shielded enclosure is required to meet compliance.

FPGA (XMC-SLX150)

Xilinx XC6SLX150-3FG676

- 184,304 CLB Flip Flops
- 1,355Kbit Distributed RAM
- 268 18Kbit Block RAMs
- 180 DSP48A1 Slices
- 6 Clock Management Tiles

REAR I/O

The rear I/O P4 XMC connector connects directly to banks 0 of the FPGA. The bank 0 Vcco pins are powered by 2.5 volts and thus will support the 2.5 volt IOStandards. Refer to the Spartan-6 SelectIO Users Guide (available from Xilinx) for more information on the IOStandards available. The example design defines the rear I/O with 2.5 volt LVDS.

- Maximum Recommended Clock Rate…..150MHz (6.7ns clock period)
- Vcco Supply Voltage ……………..…..….2.5 volt
- V_{OH} Output High Voltage…………….….…..1.602 volt
- V_{OL} Output Low Voltage…………………..…0.898 volt
- V_{ODIFF} Differential Output Voltage ……..…350m volt typical
- V_{OCM} Output Common Mode Voltage…...1.25 volt typical
- V_{IDIFF} Differential Input Voltage…………..100m volt minimum
- V_{ICM} Input Common Mode Voltage………0.3 volt min, 1.2 volt typical,
  2.35 volt max

FRONT I/O

See the mezzanine module users manual for front I/O specifications. This PMC module uses the 150 pin Samtec connector part number QSS-075-01-L-D-A which mates with the mezzanine module connector part number QTS-075-02-L-D-A-K

Write Disable Jumper

Write Disable Jumper: Removal of surface mount resistor R172 disables write to the to the Xilinx FPGA configuration flash device. The location of R172 is shown in diagram 4502-088.

Board Crystal Oscillator: 125MHz
Frequency Stability: ± 0.0020% or 20ppm

Dual Port SRAM

XMC-SLX150
256k x 64-bit Integrated Devices Technology IDT70T3519S133BC, 133 Megahertz Speed

XMC-SLX150-1M
1Meg x 64-bit Integrated Devices Technology IDT70T3509MS133BP, 133 Megahertz Speed

Flash Memory

16M x 8-bit 128 addressable sectors of which 41 are used for FPGA Configuration

PCIe Bus Interface

PCI Express 1.1: PCI Express electrical and protocol standards. Performs 2.5 Gbps data rate per lane and per direction.

ANSI/VITA 42.0: Complies with XMC module mechanicals and connectors

ANSI/VITA 42.3: XMC module with PCI Express Interface

4K Memory Space Required (BAR0): 64 bit Base Address Register 0 for access to configuration registers

4M Memory Space Required (BAR2): 64 bit Base Address Register 2 for access to the user-programmable FPGA (U7).

8M Memory Space Required (BAR4): 64 bit Base Address Register 1 for access to Dual Port SRAM.

Interrupts: Source of interrupt can be from the U7 programmable FPGA, or DMA Channels.

Certificate of Volatility

Acromag Model XMC-SLX150(E) XMC-SLX150(E)-1M	Manufacturer: Acromag, Inc. 30765 Wixom Rd Wixom, MI 48393

Volatile Memory

Does this product contain Volatile memory (i.e. Memory of whose contents are lost when power is removed)

■ Yes □ No

Type (SRAM, SDRAM, etc.) SRAM	Size: 256K x 72 or 1Meg x 72	User Modifiable ■ Yes □ No	Function: Data storage for FPGA	Process to Sanitize: Power Down
Type (SRAM, SDRAM, etc.) FPGA based RAM	Size: Variable up to 1Mbyte	User Modifiable ■ Yes □ No	Function: Data storage for FPGA	Process to Sanitize: Power Down

Non-Volatile Memory

Does this product contain Non-Volatile memory (i.e. Memory of whose contents is retained when power is removed)

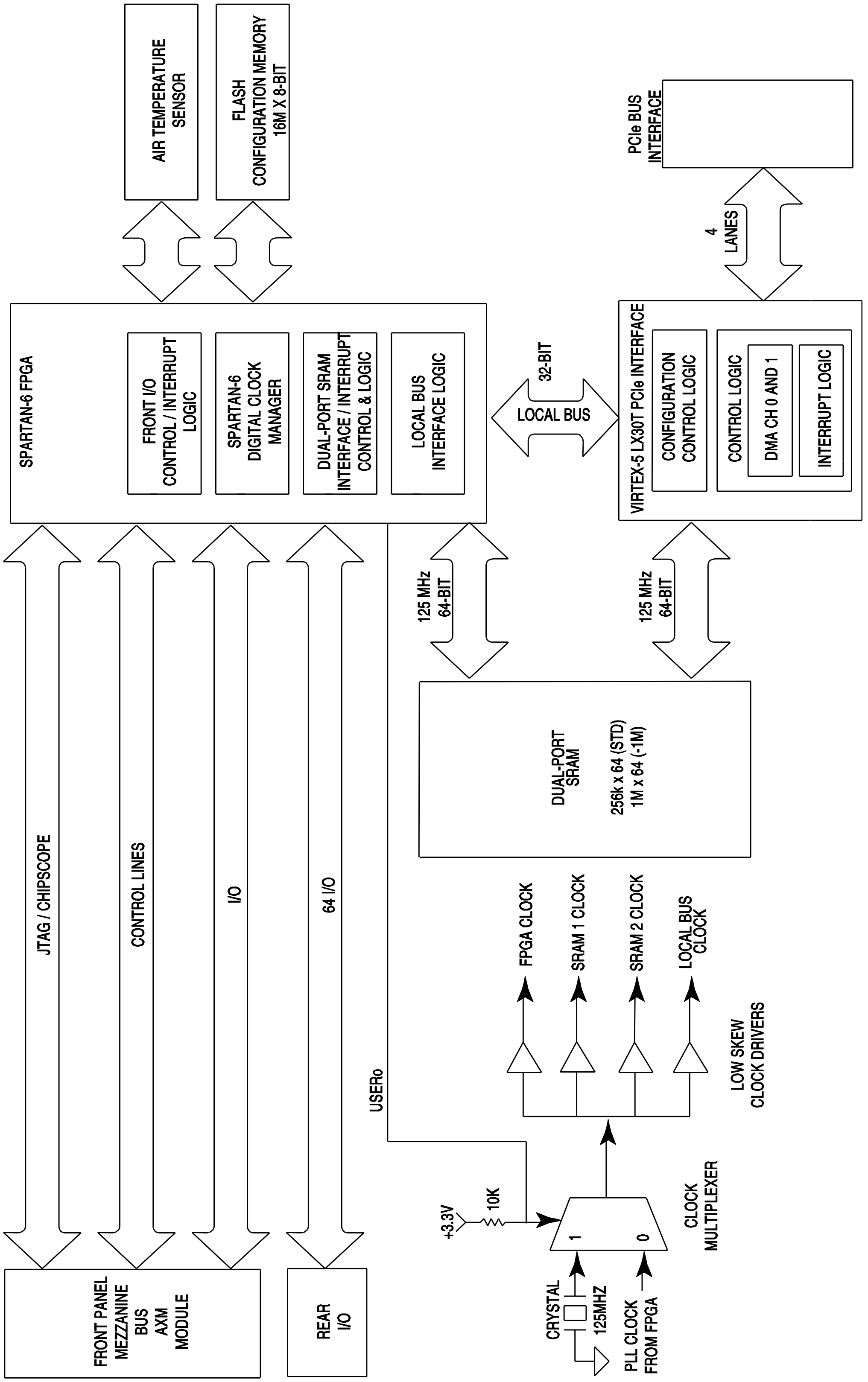
■ Yes □ No

Type(EEPROM, Flash, etc.) Flash	Size: 16Mbyte	User Modifiable ■ Yes □ No	Function: Storage of Code for FPGA	Process to Sanitize: Clear Flash memory by writing a logic 1 to bit-0 of the Flash Erase Chip Register at BAR2 + 24H
Type(EEPROM, Flash, etc.) Flash	Size: 32Mbit	User Modifiable □ Yes ■ No	Function: Storage of Code for PCIe bus Interface Device	Process to Sanitize: Not Applicable
Type(EEPROM, Flash, etc.) EEPROM	Size: 2kbit	User Modifiable □ Yes ■ No	Function: IMPI - FRU information	Process to Sanitize: Not Applicable

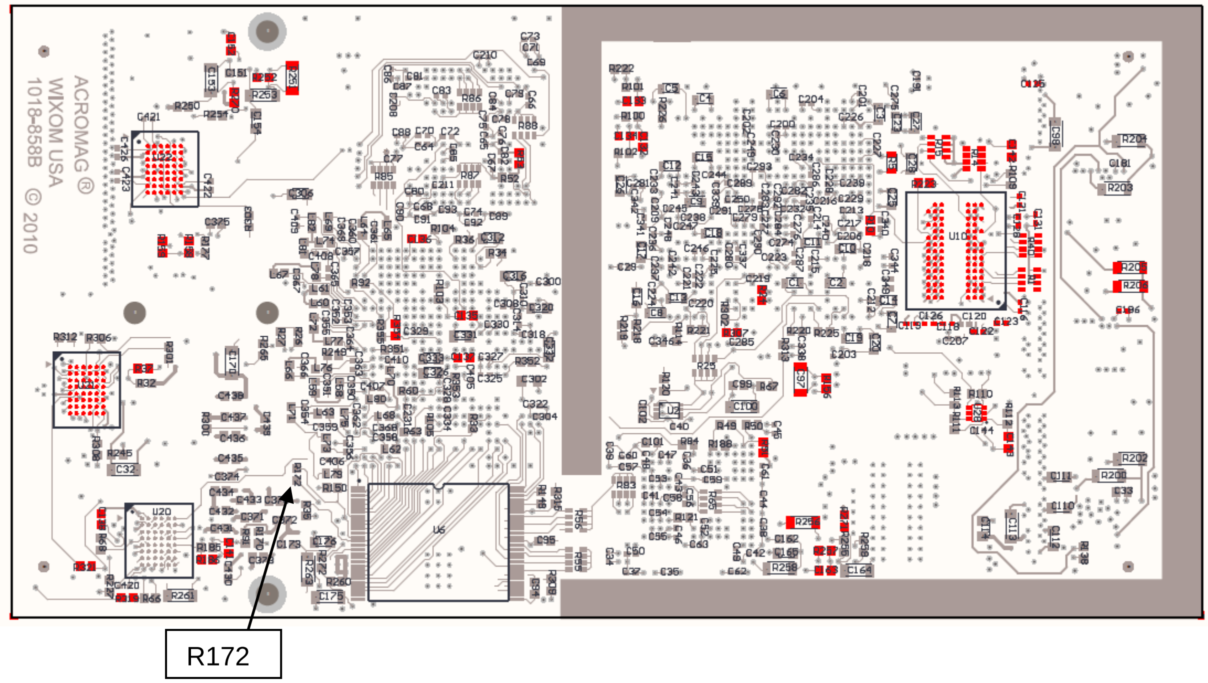
Acromag Representative

Name: Joseph Primeau	Title: Dir. of Sales and Marketing	Email: jprimeau@acromag.com	Office Phone: 248-295-0823	Office Fax: 248-624-9234

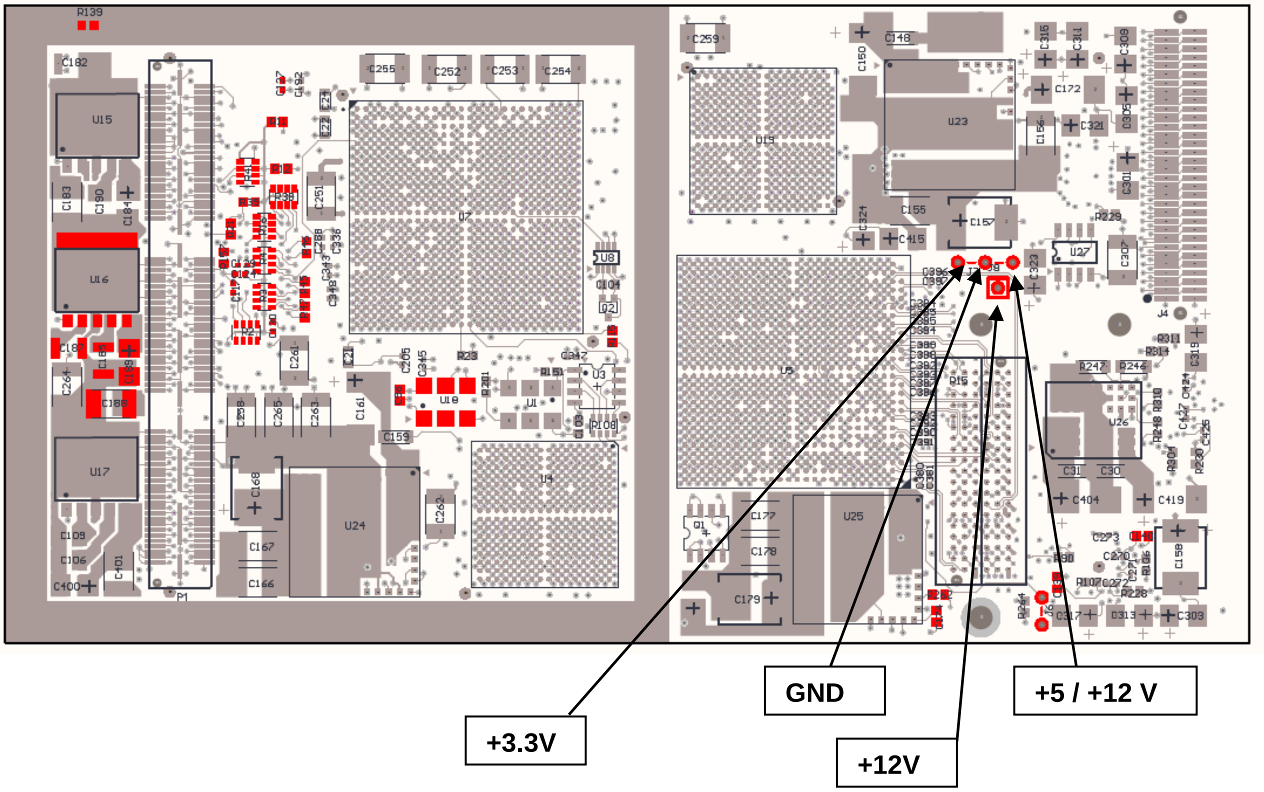
XMC-SLX Block Diagram



R172 Resistor Location
Board Bottom View



XMC SLX External Power Location
Board Top View



Revision History

The following table shows the revision history for this document:

Release Date	Version	EGR/DOC	Description of Revision
10 MAR 11	A	CAB/KLK	Initial Release.
08 APR 11	B	CAB/KLK	Update manual to include additional JTAG instructions and further information on local bus read/write cycles. (Ref. ECO 11D002)
08 DEC 14	C	CAB/ARP	Update memory map.