Programmer's Reference Manual

REV. March 2019

Liger

(VL-EPM-43)

Intel® Core™-based Computer with Dual Ethernet, Video, USB, SATA, Serial I/O, Digital I/O, Trusted Platform Module security, Counter/Timers, Mini PCIe, mSATA, SPX, and PC/104-*Plus*™ Interface

VERSALOGIC CORPORATION

WWW.VERSALOGIC.COM

12100 SW Tualatin Road
Tualatin, OR 97062-7341
(503) 747-2261
Fax (971) 224-4708

Notice:

Although every effort has been made to ensure this document is error-free, VersaLogic makes no representations or warranties with respect to this product and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose.

VersaLogic reserves the right to revise this product and associated documentation at any time without obligation to notify anyone of such changes.

* Other names and brands may be claimed as the property of others.

Product Release Notes

Release 1.2

Updated Digital I/O information on page 2

Release 1.1

Updated Uartmode1 – Uart Mode Register #1 section

Release 1

First release of this document.

Support

The EPM-43 support page, at https://versalogic.com/products/DS.asp?ProductID=284 contains additional information and resources for this product including:

- Reference Manual (PDF format)
- Operating system information and software drivers
- Data sheets and manufacturers' links for chips used in this product
- BIOS information and upgrades
- Utility routines and benchmark software

The VersaTech KnowledgeBase is an invaluable resource for resolving technical issues with your VersaLogic product.

VersaTech KnowledgeBase

Contents

1 Overview

This document provides information for users requiring register-level information for developing applications with the VL-EPM-43.

Related Documents

The following documents available are on the EPM-43 Product Support Web Page:

- *EPM-43 Hardware Reference Manual* – provides information on the board's hardware features including connectors and all interfaces.

This document is available through the software page:

- *VersaAPI Installation and Reference Guide* – describes the shared library of API calls for reading and controlling on-board devices on certain VersaLogic products.

System Resources and Maps

2

Memory Map

Table 1: Memory Map

Address Range	Description
00000h – 9FFFFh	Legacy system (DOS) area
A0000h – B7FFFh	ISA memory area (VGA frame buffer is not accessible)
B8000h – BFFFFh	Text mode buffer
C0000h – CFFFFh	Video BIOS area
D0000h – DFFFFh	PCI ROM expansion area
E0000h – FFFFFh	Legacy BIOS (reserved)

Interrupts

The LPC SERIRQ is used for interrupt interface to the Skylake SoC.

Each of the following devices can have an IRQ interrupt assigned to it and each with an interrupt enable control for IRQ3, IRQ4, IRQ5, IRQ6, IRQ7, IRQ9, IRQ10, and IRQ11:

- 8254 timers (with three interrupt status bits)
- 8 digital I/Os (with 8 interrupt status bits)
- 8 AUX GPIOs (with one interrupt status bit)
- COM 1 UART (with 16550 interrupt status bits)
- COM 2 UART (with 16550 interrupt status bits)
- Watchdog timer (one status bit)
- SPX expansion interface (status is determined by the devices on this interface). This uses selects from four of the “usual” IRQs.
- Thermal event and battery-low interrupts
- ISA interrupts

The ISA bus supports 11 interrupts: IRQ3, IRQ4, IRQ5, IRQ6, IRQ7, IRQ9, IRQ10, IRQ11, IRQ12, IRQ14, and IRQ15. There is an interrupt enable control for each and by default they are all disabled. ISA bus interrupts simply pass through to the SERIRQ (no capture in the FPGA).

Common interrupts can be assigned to multiple devices if software can deal with it (this is common on UARTs being handled by a common ISR).

Interrupt status bits for everything except the UARTs will “stick” and are cleared by a “write-one” to a status register bit. The 16550 UART interrupts behave as defined for the 16550 registers and are a pass-through to the LPC SERIRQ.

Per the VersaAPI standard, anytime an interrupt on the SERIRQ is enabled, the slot becomes active. All interrupts in the SERIRQ are high-true so when the slot becomes active, the slot will be low when there is no interrupt and high when there is an interrupt.

Table 2: I/O Map

I/O Address Range	Device/Owner
2F8h – 2FFh	COM2 serial port default
3B0h – 3DFh	Legacy VGA registers
3F8h – 3FFh	COM1 serial port default
400h – 47Fh	ACPI / Power management (reserved)
500h – 5FFh	PCH GPIO (reserved)
C80h – CBBh	EPM-43 FPGA Board Control Registers
CBCh – CBFh	EPM-43 FPGA 8254 Timer Registers

3 FPGA Registers

FPGA I/O Space

The FPGA is mapped into I/O space on the LPC bus. The address range is mapped into 64 byte I/O window.

- FPGA access: LPC I/O space
- FPGA access size: All 8-bit byte accesses (16-bit like registers are aligned on 16-bit word boundaries to make word access possible in software but the LPC bus still splits the accesses into two 8-bit accesses)
- FPGA address range: 0xCC0 to 0xCCF (64-byte window)

The three 8254 timers only require four bytes of addressing and are located at the end of the 64-byte I/O block. The only requirement is that the base address must be aligned on a 4-byte block. Table 4 lists the FPGA's I/O map.

Table 3: FPGA I/O Map

Address Range	Device	Size
0xCC0-0xCCF	FPGA registers	16 bytes

FPGA Register Map

Register Access Key	
R/W	Read/Write
RO	Read-only (status or reserved)
R/WC	Read-status/Write-1-to-Clear
WO	Write-Only
ROC	Read-Only and clear-to-0 after reading
RSVD	Reserved. Only write 0 to this bit; ignore all read values.

Reset Status Key	
POR	Power-on reset (only resets one time when input power comes on)
Platform	Resets prior to the processor entering the S0 power state (that is, at power-on and in sleep states)
resetSX	• If FPGA_PSEN is a '0' in MISCSR1 (default setting), then this is the same as the Platform reset. • If FPGA_PSEN is a programmed to a '1', then it is the same as the Power-On Reset (POR).
n/a	• Reset doesn't apply to status or reserved registers

I/O Address	Offset	Reset	D7	D6	D5	D4	D3	D2	D1	D0
C80	0	Platform	PLED	PRODUCT_CODE						
C81	1	n/a	REV_LEVEL					EXTEMP	CUSTOM	BETA
C82	2	Platform	BIOS_JMP	BIOS_OR	BIOS_SEL	LED_DEBUG	WORKVER	0	GPI_JMP	0
C83	3	Platform	IRQEN	IRQSEL2	IRQSEL1	IRQSEL0	0	IMASK_TC5	IMASK_TC4	IMASK_TC3
C84	4	Platform	INTRTEST	TMRTEST	TMRIN4	TMRIN3	0	ISTAT_TC5	ISTAT_TC4	ISTAT_TC3
C85	5	Platform	TIM5GATE	TIM4GATE	TIM3GATE	TM45MODE	TM4CLKSEL	TM3CLKSEL	TMROCTST	TMRFULL
C86	6	n/a	0	0	0	0	0	0	0	0

I/O Address	Offset	Reset	D7	D6	D5	D4	D3	D2	D1	D0
C87	**7**	n/a	0	0	0	0	0	0	0	0
C88	**8**	Platform	CPOL	CPHA	SPILEN1	SPILEN0	MAN_SS	SS2	SS1	SS0
C89	**9**	Platform	IRQSEL1	IRQSEL0	SPICLK1	SPICLK0	HW_IRQ_EN	LSBIT_1ST	HW_INT	BUSY
C8A	**A**	Platform	msb	<==========>						lsb
C8B	**B**	Platform	msb	<==========>						lsb
C8C	**C**	Platform	msb	<==========>						lsb
C8D	**D**	Platform	msb	<==========>						lsb
C8E	**E**	Platform	0	MUXSEL2	MUXSEL1	MUXSEL0	0	SERIRQEN	SPILB	0
C8F	**F**	n/a	0	0	0	0	0	0	0	0
C90	**10**	POR	0	0	0	0	0	0	AUX_PSEN	MINI_PSDIS
C91	**11**	POR	0	W_DISABLE	ETH1_OFF	ETH0_OFF	0	0	0	0
C92	**12**	Platform	0	0	0	0	0	PBRESET	0	0
C93	**13**	n/a	0	0	0	0	0	0	0	0
C94	**14**	n/a	0	0	0	0	0	0	0	0
C95	**15**	n/a	0	0	0	0	0	0	0	0
C96	**16**	n/a	0	0	0	0	0	0	0	0
C97	**17**	n/a	0	0	0	0	0	0	0	0
C98	**18**	n/a	0	0	0	0	0	0	0	0
C99	**19**	n/a	0	0	0	0	0	0	0	0
C9A	**1A**	n/a	0	0	0	0	0	0	0	0
C9B	**1B**	n/a	0	0	0	0	0	0	0	0
C9C	**1C**	n/a	0	0	0	0	0	0	0	0

I/O Address	Offset	Reset	D7	D6	D5	D4	D3	D2	D1	D0
C9D	1D	n/a	0	0	0	0	0	0	0	0
C9E	1E	n/a	0	0	0	0	0	0	0	0
C9F	1F	n/a	0	0	0	0	0	0	0	0
CA0	20	n/a	0	0	0	0	0	0	0	0
CA1	21	resetSX	DIR_GPIO8	DIR_GPIO7	DIR_GPIO6	DIR_GPIO5	DIR_GPIO4	DIR_GPIO3	DIR_GPIO2	DIR_GPIO1
CA2	22	resetSX	POL_GPIO8	POL_GPIO7	POL_GPIO6	POL_GPIO5	POL_GPIO4	POL_GPIO3	POL_GPIO2	POL_GPIO1
CA3	23	resetSX	OUT_GPIO8	OUT_GPIO7	OUT_GPIO6	OUT_GPIO5	OUT_GPIO4	OUT_GPIO3	OUT_GPIO2	OUT_GPIO1
CA4	24	n/a	IN_GPIO8	IN_GPIO7	IN_GPIO6	IN_GPIO5	IN_GPIO4	IN_GPIO3	IN_GPIO2	IN_GPIO1
CA5	25	Platform	IMASK_GPIO8	IMASK_GPIO7	IMASK_GPIO6	IMASK_GPIO5	IMASK_GPIO4	IMASK_GPIO3	IMASK_GPIO2	IMASK_GPIO1
CA6	26	Platform	ISTAT_GPIO8	ISTAT_GPIO7	ISTAT_GPIO6	ISTAT_GPIO5	ISTAT_GPIO4	ISTAT_GPIO3	ISTAT_GPIO2	ISTAT_GPIO1
CA7	27	resetSX	MODE_GPIO8	MODE_GPIO7	MODE_GPIO6	MODE_GPIO5	MODE_GPIO4	MODE_GPIO3	MODE_GPIO2	MODE_GPIO1
CA8	28	Platform	IRQEN	IRQSEL2	IRQSEL1	IRQSEL0	0	RESET_EN	WDT_EN	WDT_STAT
CA9	29	Platform	msb	<==========>						lsb
CAA	2A	Platform	0	0	0	0	0	0	COM2_MODE	COM1_MODE
CAB	2B	Platform	IRQEN	IRQSEL2	IRQSEL1	IRQSEL0	0	0	0	0
CAC	2C	Platform	0	0	0	0	0	0	0	FAN_OFF
CAD	2D	n/a	0	0	0	0	0	0	0	0
CAE	2E	Platform	msb	<==========>						lsb
CAF	2F	Platform	msb	<==========>						lsb
CB0	30	Platform	IRQEN	IRQSEL2	IRQSEL1	IRQSEL0	IMASK_BATTLOW	IMASK_EVENT	IMASK_THERM	IMASK_ALERT
CB1	31	Platform	BATTLOW	0	0	0	ISTAT_BATTLOW	ISTAT_EVENT	ISTAT_THERM	ISTAT_ALERT
CB2	32	Platform	IRQEN	IRQSEL2	IRQSEL1	IRQSEL0	UART1_BASE3	UART1_BASE	UART1_BASE1	UART1_BASE

I/O Address	Offset	Reset	D7	D6	D5	D4	D3	D2	D1	D0
								2		0
CB3	**33**	Platform	IRQEN	IRQSEL2	IRQSEL1	IRQSEL0	UART2_BASE3	UART2_BASE 2	UART2_BASE1	UART2_BASE 0
CB4	**34**	n/a	0	0	0	0	0	0	0	0
CB5	**35**	n/a	0	0	0	0	0	0	0	0
CB6	**36**	Platform	0	0	UART2_485ADC	UART1_485ADC	0	0	UART2_EN	UART1_EN
CB7	**37**	Platform	0	0	0	0	0	0	0	FAST_MODE
CB8	**38**	n/a	0	0	0	0	0	0	0	0
CB9	**39**	n/a	0	0	0	0	0	0	0	0
CBA	**3A**	n/a	0	0	0	0	0	0	0	0
CBB	**3B**	n/a	0	0	0	0	0	0	0	0
CBC	**3C**	Platform	msb	<==========>						lsb
CBD	**3D**	Platform	msb	<==========>						lsb
CBE	**3E**	Platform	msb	<==========>						lsb
CBF	**3F**	Platform	msb	<==========>						lsb
0-C7F	--	--	Unknown							
CC0	**40**	Platform	PLED	PRODUCT_CODE						
CC1	**41**	n/a	REV_LEVEL					EXTEMP	CUSTOM	BETA
CC2	**42**	n/a	WORKVER	0	0	0	0	0	0	0
CC3	**43**	n/a	0	0	0	0	0	0	0	0
CC4	**44**	Platform	UART1_EN	0	0	0	UART1_BASE3	UART1_BASE 2	UART1_BASE1	UART1_BASE 0
CC5	**45**	Platform	UART2_EN	0	0	0	UART2_BASE3	UART2_BASE 2	UART2_BASE1	UART2_BASE 0

I/O Address	Offset	Reset	D7	D6	D5	D4	D3	D2	D1	D0
CC6	**46**	n/a	0	0	0	0	0	0	0	0
CC7	**47**	n/a	0	0	0	0	0	0	0	0
CC8	**48**	Platform	ISA_IRQ11	ISA_IRQ10	ISA_IRQ9	ISA_IRQ7	ISA_IRQ6	ISA_IRQ5	ISA_IRQ4	ISA_IRQ3
CC9	**49**	Platform	ISA_ACCESS	0	0	0	0	ISA_IRQ15	ISA_IRQ14	ISA_IRQ12
CCA	**4A**	Platform	IRQ3MAP3	IRQ3MAP2	IRQ3MAP1	IRQ3MAP0	IRQ4MAP3	IRQ4MAP2	IRQ4MAP1	IRQ4MAP0
CCB	**4B**	n/a	0	0	0	0	0	0	0	0
CCC	**4C**	n/a	0	0	0	0	0	0	0	0
CCD	**4D**	n/a	0	0	0	0	0	0	0	0
CCE	**4E**	n/a	0	0	0	0	0	0	0	0
CCF	**4F**	n/a	0	0	0	0	0	0	0	0
CD0-FFFF	--	--	Unknown							

FPGA Register Descriptions

Register Access Key	
R/W	Read/Write
RO	Read-only (status or reserved)
R/WC	Read-status/Write-1-to-Clear
RSVD	Reserved. Only write 0 to this bit; ignore all read values.

Product Information Registers

This register drives the PLED on the paddleboard. It also provides read access to the product code.

Table 4: PCR – Product Code and LED Register

Bit	Identifier	Access	Default	Description
7	PLED	R/W	0	Drives the programmable LED on the paddleboard. 0 – LED is off (default) 1 – LED is on (can be used by software)
6-0	PRODUCT_CODE	RO	"0010110"	Product Code for the EPM-43 (0x16)

Table 5: PSR – Product Status Register

Bit	Identifier	Access	Default	Description
7:3	REV_LEVEL[4:0]	RO	N/A	Revision level of the PLD (incremented every FPGA release) 0 – Indicates production release revision level when BETA status bit (bit 0) is set to '0' 1 – Indicates development release revision level when BETA status bit (bit 0) is set to '1'
2	EXTEMP	RO	N/A	Extended or Standard Temp Status (set via external resistor): 0 – Standard Temp 1 – Extended Temp (probably always set)
1	CUSTOM	RO	N/A	Custom or Standard Product Status (set in FPGA): 0 – Standard Product 1 – Custom Product or PLD/FPGA
0	BETA	RO	N/A	Beta or Production Status (set in FPGA): 1 – Beta (or Debug) 0 – Production

BIOS and Jumper Status Register

Table 6: SCR –Status/Control Register

Bits	Identifier	Access	Default	Description
7	WORKVER	RO	N/A	Status used to indicate that the FPGA is Not Officially Released and is still in a working state. This bit should only be set on the FPGA developers boards. 0 – FPGA is Released 1 – FPGA is in a Working State (Not Released … Don't trust it) **Note:** This is kind of the usual position this bit lands (no other reason)
6-3	RESERVED	RO	0	Reserved – Writes are ignored. Reads always return 0
2-0	RESERVED	RO	N/A	Reserved – Writes are ignored. Reads normally always return 0 (though they may be used for sub-version indicators if WORKVER is set to a '1')

Timer Registers

The FPGA implements an 8254-compatible timer/counter that includes three 16-bit timers.

Table 7: TICR – 8254 Timer Interrupt Control Register

Bit	Identifier	Access	Default	Description
7	IRQEN	R/W	0	8254 Timer interrupt enable/disable: 0 – Interrupts disabled 1 – Interrupts enabled
6-4	IRQSEL(2:0)	R/W	000	8254 Timer interrupt IRQ select in LPC SERIRQ: 000 – IRQ3 001 – IRQ4 010 – IRQ5 011 – IRQ10 100 – IRQ6 101 – IRQ7 110 – IRQ9 111 – IRQ11
3	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
2	IMSK_TC5	R/W	0	8254 timer #5 interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled
1	IMSK_TC4	R/W	0	8254 timer #4 interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled
0	IMSK_TC3	R/W	0	8254 timer #3 interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled

Table 8: TISR – 8254 Timer Interrupt Status Register

Bit	Identifier	Access	Default	Description
7	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
6	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
5	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
4	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
3	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
2	ISTAT_TC5	RW/C	N/A	Status for the 8254 Timer #5 output (terminal count) interrupt when read. This bit is read-status and a write-1-to-clear. 0 – Timer output (terminal count) has not transitioned from 0 to a 1 level 1 – Timer output (terminal count) has transitioned from a 0 to a 1 level
1	ISTAT_TC4	RW/C	N/A	Status for the 8254 Timer #4 output (terminal count) interrupt when read. This bit is read-status and a write-1-to-clear. 0 – Timer output (terminal count) has not transitioned from 0 to a 1 level 1 – Timer output (terminal count) has transitioned from a 0 to a 1 level
0	ISTAT_TC3	RW/C	N/A	Status for the 8254 Timer #3 output (terminal count) interrupt when read. This bit is read-status and a write-1-to-clear. 0 – Timer output (terminal count) has not transitioned from 0 to a 1 level 1 – Timer output (terminal count) has transitioned from a 0 to a 1 level

Table 9: TCR – 8254 Timer Control Register

Bit	Identifier	Access	Default	Description
7	TMR5GATE	R/W	0	Debug/Test Only: Controls the "gate" signal on 8254 timer #5 when not using an external gate signal: 0 – Gate on signal GCTC5 is disabled 1 – Gate on signal GCTC5 is enabled Always set to 0 when configuring timer modes except when TMRFULL is '0' and then it should be set to '1' and not changed unless using internal clocking.
6	TMR4GATE	R/W	0	Controls the "gate" signal on 8254 timer #4 when not using an external gate signal: 0 – Gate on signal GCTC4 is disabled 1 – Gate on signal GCTC4 is enabled Always set to 0 when configuring timer modes except when TMRFULL is '0' and then it should be set to '1' and not changed unless using internal clocking
5	TMR3GATE	R/W	0	Controls the "gate" signal on 8254 timer #3 when not using an external gate signal: 0 – Gate on signal GCTC3 is disabled 1 – Gate on signal GCTC3 is enabled Always set to 0 when configuring timer modes except when TMRFULL is '0' and then it should be set to '1' and not changed unless using internal clocking
4	TM45MODE	R/W	0	Mode to set timers #4 and #5 in: 0 – Timer #4 and #5 form one 32-bit timer controlled by timer #1 signals 1 – Timer #4 and Timer #5 are separate 16-bit timers with their own control signals. Almost always used in 32-bit mode especially when TMRFULL is a '0' (the 16-bit timer #5 if of limited use)
3	TM4CLKSEL	R/W	0	Timer #4 Clock Select: 0 – Use internal 4.125 MHz clock (derived from PCI clock) 1 – Use external ICTC4 Timer #5 is always on internal clock if configured as a 16-bit clock
2	TM3CLKSEL	R/W	0	Timer #3 Clock Select: 0 – Use internal 4.125 MHz clock (derived from PCI clock) 1 – Use external ICTC3 assigned to Digital I/O
1	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
0	TMRFULL	R/W	0	DIOs to use for timer signals (TMREN must be a '1' in the DIOCR register to use the timers). 0 – 4-wire timers and DIO16-DIO13 are external timer control signals 1 – 8-wire and DIO16-DIO9 are external timer control signals Because the gates-controls are not connected to digital I/Os when TMRFULL is a '0', the TIMxGATE gate controls in this register are used so they need to be set to '1' and should not be toggled during operation with external timers (since there is no continuous clock to synchronize them to) but can be toggled if using the internal clock. If you need gating in external modes, set TMRFULL to a '1'.

SPI Control Registers

These are placed at the traditional offset 0x8 location. Only external SPX interface devices use this interface. Because the board uses a 9-pin SPX connector, only two devices are supported.

SPICONTROL

Table 10: SPI Interface Control Register

Bit	Identifier	Access	Default	Description
7	CPOL	R/W	0	SPI clock polarity – Sets the SCLK idle state. 0 – SCLK idles low 1 – SCLK idles high
6	CPHA	R/W	0	SPI clock phase – Sets the SCLK edge on which valid data will be read. 0 – Data is read on rising edge 1 – Data is read on falling edge
5-4	SPILEN(1:0)	R/W	00	Determines the SPI frame length. This selection works in manual and auto slave select modes. 00 – 8-bit 01 – 16-bit 10 – 24-bit 11 – 32-bit
3	MAN_SS	R/W	0	Determines whether the slave select lines are asserted through the user software or are automatically asserted by a write to SPIDATA3. 0 - The slave select operates automatically 1 - The slave select line is controlled manually through SPICONTROL bits SS[2:0]
2-0	SS(2:0)	R/W	000	SPI slave device selection: 000 – None 001 – SS0# 010 – SS1# 011 – Undefined (ignored) 100 – Undefined (ignored) 101 – Undefined (ignored) 110 – Undefined (ignored) 111 – Undefined (ignored)

SPISTATUS

The SPX interrupt is not connected on this product. The control bits and status associated are still defined in the register set but the SPX interrupt will always be de-asserted.

Table 11: SPI Interface Status Register

Bits	Identifier	Access	Default	Description
7-6	IRQSEL[1:0]	R/W	00	The SPX interrupt is not connected on this product (always de-asserted). Selects which IRQ will be enabled if HW_IRQ_EN = 1. Interrupts are not used on this board, so this just becomes a read/write non-functional field. 00 – IRQ3 01 – IRQ4 10 – IRQ5 11 – IRQ10 **Note:** These are the first four interrupts in the "usual" LPC SERIRQ group of eight interrupts.
5-4	SPICLK(1:0)	R/W	00	Selects one of four SCLK frequencies. This is based on a 33 MHz LPC clock. 00 – 1.03125 MHz (33 MHz/32) 01 – 2.0625 MHz (33 MHz/16) 10 – 4.125 MHz (33 MHz/8) 11 – 8.25 MHz (33 MHz/4)
3	HW_IRQ_EN	R/W	0	The SPX interrupt is not connected on this product (always de-asserted). This enables the selected IRQ to be activated by a SPI device that is configured to use its interrupt capability. 0 - IRQs are disabled for SPI operations. 1 - The IRQ can be asserted
2	LSBIT_1ST	R/W	0	Controls the SPI shift direction from the SPIDATA(x) registers. 0 - Data is left-shifted (MSB first). 1 - Data is right-shifted (LSB first)
1	HW_INT	RO	0	SPX interrupt is not connected on this product (always de-asserted). Status flag which indicates when the hardware SPX signal SINT# is asserted. 0 - The hardware interrupt SINT# is de-asserted. 1 - An interrupt is present on SINT# If HW_IRQ_EN= 1, the selected IRQ will also be asserted by the hardware interrupt. HW_INT is read-only and is cleared when the external hardware interrupt is no longer present.
0	BUSY	RO	N/A	Status flag which indicates when an SPI transaction is underway. I2C is so slow that there is no reason to ever poll this. 0 - The SPI bus is idle. 1 - SCLK is clocking data in/out of the SPIDATA(x) registers (that is, busy)

SPI Data Registers

There are four data registers used on the SPI interface. How many are used depends on the device being communicated with. SPIDATA0 is typically the least significant byte and SPIDATA3 is the most significant byte. Any write to the most significant byte SPIDATA3 initiates the SCLK and, depending on the MAN_SS state, will assert a slave select to begin an SPI bus transaction.

Data is sent according to the LSBIT_1ST setting. When LSBIT_1ST = 0, the MSbit of SPIDATA3 is sent first and received data will be shifted in the LSbit of the selected frame size determined by SPILEN1 and SPILEN0. When LSBIT_1ST = 1, the LSbit of the selected frame size is sent first and the received data will be shifted in the MSbit of SPIDATA3.

SPIDATA0 (Least Significant Byte)

D7	D6	D5	D4	D3	D2	D1	D0
MSB							LSB

SPIDATA1

D7	D6	D5	D4	D3	D2	D1	D0
MSB							LSB

SPIDATA2

D7	D6	D5	D4	D3	D2	D1	D0
MSB							LSB

SPIDATA3 (Most Significant Byte) [Cycle Trigger Register]

D7	D6	D5	D4	D3	D2	D1	D0
MSB							LSB

SPI Debug Control Register and mSATA/PCIe Select Control Register

This register is only used to set an SPI loopback (debug/test only) but is also used for the mSATA/PCIe Minicard Mux select.

Table 12: SPI – SPI Debug Control Register

Bit	Identifier	Access	Default	Description
7	Reserved	RO	0	Reserved. Writes are ignored; reads always return 0.
6-4	MUXSEL(2:0)	R/W	000	mSATA/PCIe Mux selection for Minicard slot (and 2nd SATA connector): • 000 – Select mSATA using only pin 43 (MSATA_DETECT). This is an Intel-mode that is reliable for PCIe Minicards but not for mSATA modules that inadvertently ground this signal. • 001 – Use only Pin 51 (PRES_DISABLE2#). This is the default method and is defined in the Draft mSATA spec but some Minicards use it as a second wireless disable. • 010 – Use either Pin 43 or Pin 51. This will work just like 001 because Pin 43 is disabled by an FPGA pull-down. • 011 – Force PCIe mode on the Minicard • 100 – Force mSATA mode on the Minicard. • 101 – Undefined (same as 000) • 110 – Undefined (same as 000) • 111 – Undefined (same as 000) **Note:** When the Minicard uses PCIe, the SATA channel automatically switches to the SATA connector.
3	Reserved	RO	0	Reserved. Writes are ignored; reads always return 0.
2	SERIRQEN	R/W	0	When an IRQ is assigned a slot in the SERIRQ, it will drive the slot with the interrupt state, but this bit must be set to a '1' to do that. 0 – Slots assigned to SERIRQ are not driven (available for other devices). 1 – Slots assigned to SERIRQ are driven with their current interrupt state (which is low since interrupts are high-true). This is because the default interrupt settings in this FPGA can conflict with other interrupts if the VersaAPI is not being used (for example, console redirect using IRQ3).
1	SPILB	R/W	0	Debug/Test Only: Used to loop SPI output data back to the input (debug/test mode). 0 – Normal operation 1 – Loop SPI output data back to the SPI input data (data output still active)
0	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.

Miscellaneous FPGA Registers

MISCR1 – Miscellaneous Control Register #1

This is a register in the always-on power well of the FPGA. It holds its state during sleep modes and can only be reset by a power cycle. This is a placeholder register for features like pushing the power-button and also for software initiated resets should those be needed. This register is only reset by the main power-on reset since it must maintain its state in Sleep modes (for example, S3).

Table 13: MISCR1 – Misc. Control Register #1

Bits	Identifier	Access	Default	Description
7-2	Reserved	RO	000000	Reserved. Writes are ignored; reads always return 0.
1	FPGA_PSEN	R/W	0	FPGA digital I/O and AUX GPIO bank I/O power enable 0 – The digital I/O and AUX GPIO bank will be powered down in sleep modes (only power in S0) 1 – The digital I/O and AUX GPIO bank will not be powered down in sleep modes and the configuration will remain. The FPGA 3.3 V I/O bank power switch is controlled by the “OR” of the S0 power control signal and FPGA_PSEN. **Note:** Some register resets are conditional on the state of FPGA_PSEN
0	MINI_PSDIS	R/W	0	Minicard 3.3 V power disable 0 – Minicard 3.3 V power stays on always (this is normally how minicards operate if they support any Wake events) 1 – Minicard 3.3 V power will be turned off when not in S0 (in sleep modes). The Minicard 3.3 V power switch is controlled by the “OR” of the S0 power control signal and the inverse of MINI_PSDIS.

MISCR2 – Miscellaneous Control Register #2

This is a register in the always-on power well of the FPGA. It holds its state during sleep modes and can only be reset by a power cycle. It is primarily used for control signals for the always-powered Ethernet controllers and the USB hubs. This register is only reset by the main power-on reset since it must maintain its state in sleep modes (for example, S3).

Table 14: MISCSR2 – Misc. Control Register #2

Bit	Identifier	Access	Default	Description
7	USB_HUBMODE	R/W	0	Determines whether the hub resets only once (to support wake-up from sleep modes on USB ports) or resets every time it enters sleep modes using the platform reset: 0 – USB hub is reset once at power on. Use USB_HUBDIS to manually control the reset if necessary. This supports USB wake-up modes 1 – USB hub is reset by platform reset every time (will be reset when entering all sleep modes). USB ports cannot be used to wake-up
6	W_DISABLE	R/W	0	Used to control the W_DISABLE (Wireless Disable) signal going to the PCIe Mincard: 0 – W_DISABLE signal is not asserted (Enabled) 1 – W_DISABLE signal is asserted (Disabled) **Note:** There are other control sources that can be configured to control this signal and if enabled the control becomes the “OR” of all sources
5	ETHOFF1	R/W	0	Used to disable the Ethernet controller #1 (controls the ETH_OFF# input to the I210-IT): 0 – Ethernet controller is enabled (On) 1 – Ethernet controller is disabled (Off)
4	ETHOFF0	R/W	0	Used to disable the Ethernet controller #0 (controls the ETH_OFF# input to the I210-IT): 0 – Ethernet controller is enabled (On) 1 – Ethernet controller is disabled (Off)
3	USB_USBID	R/W	0	Set to use the “ID” signal on the on-board USB 3.0 signal to control the VBUS power. USB OTG (on-the-go) uses this signal to tell whether an “A” or “B” cable is plugged in a micro-USB 3.0 “AB” connector. When USB_USBID is set to a ‘1’, an “A” cable will turn VBUS power on and a “B” will turn it off (because “B” devices are not supported). 0 – Do not use “ID” signal to control VBUS power (VBUS power controlled only by USB_OBDIS) 1 – Use “ID” signal to control VBUS power (USB_OBDIS will still disable VBUS power)
2	USB_HUBDIS	R/W	0	Control the reset on the USB2513B Hub. 0 – USB2513 hub is enabled (reset released) 1 – USB2513 hub is in reset
1	USB_PBDIS	R/W	0	Disable control for the paddleboard USB port VBUS power switches (there are two power-switches but they have a common power enable and overcurrent status) 0 – VBUS power switches are enabled 1 – VBUS power switched are disabled. **Note:** The I2164 power switches latch-off in overcurrent and can only be re-enabled by a power-cycle or by setting this bit to a 1, wait >1 ms, and then a 0
0	USB_OBDIS	R/W	0	Disable control for the on-board USB port VBUS power switches (there are two with a common overcurrent): 0 – VBUS power switches are enabled 1 – VBUS power switched are disabled. **Note:** The I2164 power switches latch-off in overcurrent and can only be re-enabled by a power-cycle of by setting this bit to a 1 and then a 0 with at least 1 ms in between

MISCR3 – Miscellaneous Control Register #3

This register sets the SMBus addresses on the 4-Port PCIe Switch.

Table 15: MISCR3 – Misc. Control Register #3

Bits	Identifier	Access	Default	Description
7-4	Reserved	RO	0000	Reserved. Writes are ignored; reads always return 0.
3	USB_PBOC	RO	N/A	Reads the overcurrent status for the USB paddleboard power switches (there are two power switches for the four ports but they have a common overcurrent status). 0 – Overcurrent is not asserted (power switch is on) 1 – Overcurrent is asserted (power switch is off)
2	PBRESET	R/W	---	When written to, this will do the same thing as pushing the reset button, which could be useful for a software-initiated watchdog. 0 – No action 1 – Activate the reset push-button **Note:** Because this generates a reset that will reset this register, it is not likely a value of a '1' can ever be read-back, so it is somewhat "write-only".
1-0	Reserved	RO	00	Reserved. Writes are ignored; reads always return 0.

AUXDIR – AUX GPIO Direction Control Register

This register controls the direction of the eight AUX GPIO signals.

This reset depends on the state of the FPGA_PSEN signal. If FPGA_PSEN is a '0' then the reset is the power-on and Platform Reset. If FPGA_PSEN is a '1' then this register is only reset at power-on.

Table 16: AUXDIR – AUX GPIO Direction Control Register

Bit	Identifier	Access	Default	Description
7-0	DIR_GPIO[8:1]	R/W	0	Sets the direction of the AUX GPIOx lines. For each bit: 0 – Input 1 – Output

AUXPOL – AUX GPIO Polarity Control Register

This register controls the polarity of the eight AUX GPIO signals.

This reset depends on the state of the FPGA_PSEN signal. If FPGA_PSEN is a '0' then the reset is the power-on and Platform Reset. If FPGA_PSEN is a '1' then this register is only reset at power-on.

Table 17: AUXPOL – AUX GPIO Polarity Control Register

Bits	Identifier	Access	Default	Description
7-0	POL_GPIO[8:1]	R/W	0	Sets the polarity of the AUX GPIOx lines. For each bit: 0 – No inversion 1 – Invert **Note:** This impacts the polarity as well as the interrupt status edge used.

AUXOUT – AUX GPIO Output Control Register

This register sets the AUX GPIO output value. This value will only set the actual output if the GPIO direction is set as an output. Reading this register does not return the actual input value of the GPIO (use the AUXIN register for that). As such, this register can actually be used to detect input/output conflicts.

This reset depends on the state of the FPGA_PSEN signal. If FPGA_PSEN is a '0' then the reset is the power-on and Platform Reset. If FPGA_PSEN is a '1' then this register is only reset at power-on.

Table 18: AUXOUT – AUX GPIO Output Control Register

Bits	Identifier	Access	Default	Description
7-0	OUT_GPIO[8:1]	R/W	0	Sets the AUX GPIOx output values. For each bit: 0 – De-asserts the output (0 if polarity not-inverted, 1 if inverted) 1 – Asserts the output (1 if polarity not-inverted, 0 if inverted)

AUXIN – AUX GPIO I/O Input Status Register

This registers sets the AUX GPIO input value. It will read the input value regardless of the setting on the direction (that is, it always reads the input). This reads the actual state of the GPIO pin into the part.

Table 19: AUXIN – AUX GPIO Input Status Register

Bits	Identifier	Access	Default	Description
7-0	IN_GPIOIO[8:1]	RO	N/A	Reads the GPIOx input status. For each bit: 0 – Input de-asserted if polarity not-inverted; asserted if polarity inverted 1 Input asserted if polarity not-inverted; de-asserted if polarity inverted

AUXIMASK – AUX GPIO Interrupt Mask Register

This is the interrupt mask registers for the AUX GPIOs and the interrupt enable selection. The reset type is Platform Reset because interrupts always have to be setup after exiting sleep states.

Table 20: AUXICR – AUX GPIO Interrupt Mask Register

Bits	Identifier	Access	Default	Description
7-0	IMASK_GPIO[8:1]	R/W	0	GPIOx interrupt mask. For each bit: 0 – Interrupt disabled 1 – Interrupt enabled

AUXISTAT – AUX GPIO I/O Interrupt Status Register

Table 21: AUXISTAT – AUX GPIO Interrupt Status Register

Bits	Identifier	Access	Default	Description
7-0	ISTAT_GPIO[8:1]	RW/C	N/A	GPIOx interrupt status. A read returns the interrupt status. Writing a '1' clears the interrupt status. This bit is set to a '1' on a transition from low-to-high (POL_DIOx=0) or high-to-low (POL_DIOx=1).

AUXMODE1– AUX I/O Mode Register #1

These two registers selected the mode on each AUX GPIO. This reset depends on the state of the FPGA_PSEN signal. If FPGA_PSEN is a '0' then the reset is the power-on and Platform Reset. If FPGA_PSEN is a '1' then this register is only reset at power-on.

Table 22: AUXMODE1 – AUX I/O Mode Register

Bit	Identifier	Access	Default	Description
7	MODE_GPIO8	R/W	0	GPIO8 mode. 0 – GPIO (I/O) 1 – W_DISABLE (input). In this mode, the GPIO is passed through to the W_DISABLE# signal. The GPIO polarity control applies. The GPIO input status can still be read.
6	MODE_GPIO7	R/W	0	GPIO7 mode. 0 – GPIO (I/O) 1 – SLEEP (output). In this mode, the GPIO indicates that the CPU is in a sleep mode (S3 or higher). The GPIO polarity control applies. The GPIO input status can still be read.
5	MODE_GPIO6	R/W	0	GPIO6 mode. 0 – GPIO (I/O) 1 – PWM0 from BayTrail SoC (output). In this mode, the GPIO tracks the PWM0 signal from the BayTrail SoC. The GPIO polarity control applies. The GPIO input status can still be read.
4	MODE_GPIO5	R/W	0	GPIO5 mode. 0 – GPIO (I/O) 1 – Power Good (output). In this mode, the GPIO indicates that the board is in the S0 state and all power is good. The GPIO polarity control applies. The GPIO input status can still be read.
3	MODE_GPIO4	R/W	0	GPIO4 mode. 0 – GPIO (I/O) 1 – Watchdog Reset (output). In this mode, the GPIO will be the watchdog timer trigger output that signals external equipment that the watchdog fired. The GPIO polarity control applies. The GPIO input status can still be read.
2	MODE_GPIO3	R/W	0	GPIO3 mode. 0 – GPIO (I/O) 1 – Reserved for future secondary mode (do not set)
1	MODE_GPIO2	R/W	0	GPIO2 mode. 0 – GPIO (I/O) 1 – Reserved for future secondary mode (do not set)
0	MODE_GPIO1	R/W	0	GPIO1 mode. 0 – GPIO (I/O) 1 – Reserved (same as GPIO)

WDT_CTL – Watchdog Control Register

Reset type is Platform.

Table 23: WDT_CTL – Watchdog Control Register

Bits	Identifier	Access	Default	Description
7	IRQEN	R/W	0	Watchdog interrupt enable/disable: 0 – Interrupts disabled 1 – Interrupts enabled
6-4	IRQSEL(2:0)	R/W	000	Watchdog interrupt IRQ select in LPC SERIRQ: 000 – IRQ3 001 – IRQ4 010 – IRQ5 011 – IRQ10 100 – IRQ6 101 – IRQ7 110 – IRQ9 111 – IRQ11
3	Reserved	RO	0	Reserved. Writes are ignored; reads always return 0.
2	RESET_EN	R/W	0	Enable the Watchdog to assert the push-button reset if it “fires”. 0 – Watchdog will not reset the board 1 – Board will be reset if the Watchdog “fires”
1	WDT_EN	R/W	0	Watchdog Enable: 0 – Watchdog is disabled 1 – Watchdog is enabled **Note:** The WDT_VAL register must be set before enabling.
0	WDT_STAT	RO	0	Watchdog Status: 0 – Watchdog disabled or watchdog has not “fired” 1 – Watchdog fired. **Note:** Once set to a ‘1’, it will remain so until any of the following occurs: • the WDT_VAL register is written to • the WDT_EN is disabled • a reset occurs

WDT_VAL – Watchdog Value Register

This register sets the number of seconds for a Watchdog prior to enabling the watchdog. By writing this value, the watchdog can be prevented from "firing". A watchdog fires whenever this registers value is all 0s, so it must be set to a non-zero value before enabling the watchdog to prevent an immediate "firing". Reset type is Platform.

The value written should always be 1 greater than the desired timeout value due to a 0-1 second "tick" error band (values written should range from 2-255 because a 1 could cause an immediate trigger); that is, the actual timeout is WDT_VAL seconds with a -1 second to 0 second error band.

Table 24: WDT_VAL – Watchdog Control Register

Bits	Identifier	Access	Default	Description
7-0	WDT_VAL(7:0)	R/W	0x00	Number of seconds before the Watchdog fires. By default, it is set to zero which results in an immediate watchdog if WDT_EN is set to a '1'.

XCVRMODE – COM Transceiver Mode Register

Sets the RS232 vs RS422/485 mode on the COM port Transceivers. These drive the UART_SEL signals from the FPGA to the Transceivers.

Reset type is Platform.

Note: The values shown are for the default BIOS configuration.

Table 25: XCVRMODE – COM Transceiver Mode Register

Bits	Identifier	Access	Default	Description
7-4	Reserved	RO	0000	Reserved. Writes are ignored; reads always return 0.
3-2	Reserved	RO	00	Reserved. Writes are ignored; reads always return 0.
1	COM2_MODE	R/W	0	COM2 Transceiver mode: 0 – RS232 1 – RS422/485
0	COM1_MODE	R/W	0	COM1 Transceiver mode: 0 – RS232 1 – RS422/485

AUXMODE2– AUX I/O Mode Register #2

This register defines the interrupt mapping for the AUX GPIOs. Reset type is Platform.

Table 26: AUXMODE2 - AUX I/O Mode Register #2

Bits	Identifier	Access	Default	Description
7	IRQEN	R/W	0	AUX GPIO interrupt enable/disable: 0 – Interrupts disabled 1 – Interrupts enabled
6-4	IRQSEL(2:0)	R/W	000	AUX GPIO interrupt IRQ select in LPC SERIRQ: 000 – IRQ3 001 – IRQ4 010 – IRQ5 011 – IRQ10 100 – IRQ6 101 – IRQ7 110 – IRQ9 111 – IRQ11
3-0	Reserved	RO	0000	Reserved. Writes are ignored; reads always return 0.

FANCON – Fan Control Register

The fan is always off in any sleep mode. When the processor comes out of a sleep state, this register must be setup again since it will be reset to default by the platform reset signal. The fan is always turned “off” in sleep modes. No PWM fan control is supported on the EPM-43.

Reset type is Platform.

Note: The BIOS (via ACPI) may modify this register when in an ACPI-capable operating system. The register can be read for status purposes but do not write to it unless you are using a non-ACPI operating system.

Table 27: FANCON – Fan Control Register

Bits	Identifier	Access	Default	Description
7-1	Reserved	RO	0000000	Reserved. Writes are ignored; reads always return 0.
0	FAN_OFF	R/W	0	Fan enable: 0 – Fan is on 1 – Fan is off

FANTACHLS, FANTACHMS – Fan Tach Status Registers

These registers contain the number of fan tach output samples over a one-second sampling period. The value is always valid after the fan speed stabilizes and is updated every 1 second (after a delay of 1 second). Currently, only the lower 10-bits have a valid tach reading (that is, the upper 6 bits will always be zero). The fan tach count should never overflow in the one second period, but it if does, the value will "stick" at 0x03FF.

The board can handle up at least a 10,000 RPM fan with a fan tach output of up to four uniform pulses per revolution. The duty cycle of the fan tach output pulse can be as low as 25% (typically they are very close to 50%). The conversion to RPM is as follows:

RPM = (FANTACH x 60) / PPR

Where…

- FANTACH - the 16-bit register reading
- PPR – fan tach pulses per revolution (typically either 1, 2, or 4)

Reset type is N/A.

Table 28: FANTACHLS – FANTACH Status Register Least Significant Bits

Bits	Identifier	Access	Default	Description
7-0	FANTACH[7:0]	RO	N/A	Least significant eight bits of FANTACH. Read this register first since it latches the value for the most significant eight bits.

Table 29: FANTACHMS – FANTACH Status Register Most Significant Bits

Bits	Identifier	Access	Default	Description
7-0	FANTACH[15:8]	RO	N/A	Most significant eight bits of FANTACH. Read this register after reading FANTACHLS.

Integrator's Note:

The FANTACHLS register must be read first. It will latch a copy of the MS bits so that when FANTACHMS is read, it is based on the same 16-bit value. This assumes that a 16-bit word read on the LPC bus reads the even (LS) address before the odd (MS) address.

TEMPICR – Temperature Interrupt Control Register

This is the interrupt mask register for the temperature sensor thermal alerts and the DDR3 SODIMM EVENT signals and the interrupt enable and selection. The SODIMM may not have any temperature event capability.

Reset type is Platform.

Table 30: TEMPICR – Temperature Interrupt Control Register

Bits	Identifier	Access	Default	Description
7	IRQEN	R/W	0	Temperature interrupt enable/disable: 0 – Interrupts disabled 1 – Interrupts enabled
6-4	IRQSEL(2:0)	R/W	000	Temperature interrupt IRQ select in LPC SERIRQ: 000 – IRQ3 001 – IRQ4 010 – IRQ5 011 – IRQ10 100 – IRQ6 101 – IRQ7 110 – IRQ9 111 – IRQ11
3	IMASK_BATTLOW	R/W	0	Battery-low interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled.
2	IMASK_EVENT	R/W	0	SODIMM EVENT output interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled.
1	IMASK_THERM	R/W	0	Temperature Sensor THERM output interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled.
0	IMASK_ALERT	R/W	0	Temperature Sensor ALERT output interrupt mask: 0 – Interrupt disabled 1 – Interrupt enabled.

TEMPISTAT – Temperature Interrupt Status Register

Reset type: N/A.

Table 31: TEMPISTAT – Temperature Interrupt Status Register

Bits	Identifier	Access	Default	Description
7	IN_BATTLOW	RO	N/A	Reads the battery low input status. 0 – battery-low is de-asserted (battery is OK) 1 – battery-low is asserted (battery is low)
6-4	Reserved	RO	000	Reserved. Writes are ignored; reads always return 0.
3	ISTAT_BATTLOW	RW/C	N/A	Battery-Low interrupt status. A read returns the interrupt status. Writing a '1' clears the interrupt status. This bit is set to a '1' on a transition from de-asserted-to-asserted
2	ISTAT_EVENT	RW/C	N/A	SODIMM EVENT interrupt status. A read returns the interrupt status. Writing a '1' will clear the interrupt status
1	ISTAT_THERM	RW/C	N/A	Temperature Sensor THERM interrupt status. A read returns the interrupt status. Writing a '1' clears the interrupt status
0	ISTAT_ALERT	RW/C	N/A	Temperature Sensor ALERT interrupt status. A read returns the interrupt status. Writing a '1' clears the interrupt status

UART1CR – UART1 Control Register (COM1)

Reset type is Platform.

Note: The BIOS (via ACPI) may modify this register when in an ACPI-capable operating system. The register can be read for status purposes but do not write to it unless you are using a non-ACPI operating system.

Table 32: UART1CR – UART1 Control Register (COM1)

Bits	Identifier	Access	Default	Description
7	UART1_EN	R/W	1	UART on the EPMe-42 is Enabled/Disabled: 0 – UART is disabled 1 – UART is enabled. **Note:** Bit was in the UARTMODE1 register on the EPMe-42.
6-4	RESERVED	RO	0	Reserved – Writes are ignored. Reads always return 0
3-0	UART1_BASE(3:0)	R/W	0000	UART Base Address: 0000 - 3F8h **[← COM1 Default]** 0001 - 2F8h 0010 - 3E8h 0011 - 2E8h 0100 - 200h 0101 - 208h 0110 - 220h 0111 - 228h 1000 - 238h 1001 - 338h 1010-1111 **[← These values are reserved; do not use.]**

UART2CR – UART2 Control Register (COM2)

Reset type is Platform.

Note: The BIOS (via ACPI) may modify this register when in an ACPI-capable operating system. The register can be read for status purposes but do not write to it unless you are using a non-ACPI operating system.

Table 33: UART2CR – UART2 Control Register (COM2)

Bits	Identifier	Access	Default	Description
7	UART2_EN	R/W	1	UART on the EPMe-42 is Enabled/Disabled: 0 – UART is disabled 1 – UART is enabled. **Note:** Bit was in the UARTMODE1 register on the EPMe-42.
6-4	RESERVED	RO	0	Reserved – Writes are ignored. Reads always return 0
3-0	UART2_BASE(3:0)	R/W	0001	UART Base Address: 0000 - 3F8h 0001 - 2F8h **[← COM2 Default]** 0010 - 3E8h 0011 - 2E8h 0100 - 200h 0101 - 208h 0110 - 220h 0111 - 228h 1000 - 238h 1001 - 338h 1010-1111 **[← These values are reserved; do not use.]**

UARTMODE1 – UART MODE REGISTER #1

When the COM Transceiver Mode is set to RS422/485 (in the XCVRMODE register) and the RS-485 Automatic Direction Control is enabled (e.g., UART1_485ADC set to ‘1’) then the transceiver Tx output is enabled. When there are bytes to transmit and the transceiver Tx output is disabled (i.e., tri-stated) when there are no bytes to transmit.

When the COM Transceiver Mode is set to RS422/485 and Automatic Direction Control is disabled (e.g., UART1_485ADC set to ‘0’) then the UART is in Manual Direction Control mode and the transceiver Tx output enable is controlled by software using the RTS bit in the UART Modem Control Register.

RTS = '0' - Transceiver Tx output is enabled.

RTS = '1' - Transceiver Tx output is disabled (i.e., tri-stated).

Warning: Terminal software, expecting an RS-232 port, may set RTS to '1' and disable the transmitter when initializing an RS-422/485 port in Manual Direction Control mode. Application software that handles the RS-422/485 port should set RTS to '0' to enable transmitting when in Manual Direction Control mode.

Reset type is Platform.

Note: The values shown are for the default BIOS configuration.

Table 34: UARTMODE1 – UART MODE Register #1

Bits	Identifier	Access	Default	Description
7-6	Reserved	RO	00	Reserved. Writes are ignored; reads always return 0.
5	UART2_485ADC	R/W	0	COM2 RS-485 Automatic Direction Control: 0 – Disabled 1 – Enabled **Note:** Only enable in RS-485 mode. The COM2_MODE in XCVRMODE register must also be set to a ’1’
4	UART1_485ADC	R/W	0	COM1 RS-485 Automatic Direction Control: 0 – Disabled 1 – Enabled **Note:** Only enable in RS-485 mode. . The COM1_MODE in XCVRMODE register must also be set to a ’1’
3-2	Reserved	RO	00	Reserved. Writes are ignored; reads always return 0.
1	UART2_EN	R/W	1	UART #2 Output Enable: 0 – Tx and RTS outputs are disabled and UART I/O accesses are passed to the ISA bus. 1 – Tx and RTS outputs are enabled
0	UART1_EN	R/W	1	UART #1 Output Enable: 0 – Tx and RTS outputs are disabled and UART I/O accesses are passed to the ISA bus 1 – Tx and RTS outputs are enabled

Note: When a UART is disabled, there will be no decoding of the UART I/O address range and any I/O activity will be passed to the ISA bus (this allows COM ports to be on the ISA bus if the FPGA UARTs are not used). This means that the UART must be enabled before accessing registers. If this is an issue, separate control bits can be added in UARTMODE2 or make one of the unused UART Base addresses a no-decode.

UARTMODE2 – UART MODE REGISTER #2

Standard software (the BIOS and the operating system) assumes the baud-rate clock is 1.8432 MHz and programs the divisors accordingly; however, a faster oscillator is needed for baud rates higher than 115,200.

The FAST_MODE bit in this register is used to shift the divisor by 4 bits (multiply by 16) so that the legacy baud rate comes out correctly for the 16x UART clock. This bit must be set to use rates above 115,200 and may require custom software.

Reset type is Platform.

Note: The values shown are for the default BIOS configuration.

Table 35: UARTMODE2 – UART MODE Register #2

Bits	Identifier	Access	Default	Description
7-1	Reserved	RO	0000000	Reserved. Writes are ignored; reads always return 0.
0	FAST_MODE	R/W	0	Sets how the baud-rate divisor for the 16550 UARTs are interpreted (applies to all ports): 0 – Divisor is multiplied by 16 (legacy mode for 1.8432 MHz clock) 1 – Divisor is not modified (fast mode for 16x 1.8432 MHz clock) **Note:** This must be set to '1' to use baud rates above 115,200.

ISACONx (x = 1,2) – ISA Control Registers

These register are used to enable ISA interrupts on the LPC SERIRQ. ISA interrupts simply pass through to SERIRQ and - per the ISA bus standard - are always high-true. The SERIRQEN control bit is not used for the ISA interrupt mask and should not be set until the interrupt processing is ready.

Note: The values shown are for the default BIOS configuration.

Table 36: ISACON1 – ISA Control Register #1

Bits	Identifier	Access	Default	Description
7	ISA_IRQ11	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
6	ISA_IRQ10	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
5	ISA_IRQ9	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
4	ISA_IRQ7	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
3	ISA_IRQ6	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
2	ISA_IRQ5	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
1	ISA_IRQ4	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
0	ISA_IRQ3	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ

Table 37: ISACON2 – ISA Control Register #2

Bits	Identifier	Access	Default	Description
7	ISA_ACCESS	ROC	0	Status bit that is set and held when a prior LPC access was decoded to go to the ISA bus. Writes to the register bit are ignored. 0 – No ISA Access has been made since last time this register was read 1 – ISA Access was made. This bit is cleared-to-zero after this register is read. **Note:** This is an alternative to using the ISASCAN method and is easier (no shorting of ISA data lines) and more reliable. This is not intended to be used for normal operation … just debug/test. This is not very useful for an ISASCAN if there are any ISRs running that access the ISA bus
6-5	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
4	ISA_16MODE	R/W	0	ISA 16-Bit Mode (applies to cards that assert IOCS16#, MEMCS16#). 0 – 16-bits cards support both 8-bit (using SBHE) and 16-bit accesses (this is the more standard 16-bit mode) 1 – 16-bit cards support only 16-bit accesses (not 8-bit). Sometimes referred to as “Optia 16-Bit ISA Mode”. Defaults to 8/16 mode. BIOS config required to use the custom “Optia” mode
3	RESERVED	RO	0	Reserved. Writes are ignored; reads always return 0.
2	ISA_IRQ15	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
1	ISA_IRQ14	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ
0	ISA_IRQ12	R/W	0	Interrupt enable for ISA interrupt 0 – Do not pass ISA interrupt to SERIRQ 1 – Pass ISA interrupt to SERIRQ

Programming Information for Hardware Interfaces

4

Processor WAKE# Capabilities

The following devices can wake up the processor using the PCIE_WAKE# signal to the SoC:

- Ethernet I210 Controller for Port 1
- Minicard (when always-powered)
- FPGA via either a Digital I/O or AUX connector GPIO

The following nine USB devices can wake up the processor using the in-band SUSPEND protocol:

- The 2 On-Board USB 3.0 Ports
- Any of the 4 CBR-4005 Paddleboard USB Ports
- Either of the 2 PCI104/Express “A” Connector (1-Blade) USB Ports
- Minicard (when always-powered)

Watchdog Timer

A Watchdog timer is implemented within the FPGA. When triggered, the Watchdog timer can set a status bit, generate an interrupt and/or hit the push-button-reset. The Watchdog timer implements a 1-255 second timeout.

The Watchdog time out is set in an 8-bit register (WDT_VAL). When the Watchdog is enabled, the WDT_VAL will start to count down. If the Watchdog is enabled and whenever WDT_VAL is zero, the Watchdog is triggered (so a non-zero value must be written before enabling the watchdog). Software must periodically write a non-zero value to WDT_VAL to prevent this trigger. The value written should always be 1 greater than the desired timeout value due to a 0-1 second error band. Values written should be from 2-255 because a 1 could cause an immediate trigger); that is, the actual timeout is WDT_VAL seconds with a -1 second to 0 second error band.

The Watchdog control/status register(s) have bits for the following:

- Watchdog enable/disable (disabled by default)
- Watchdog timeout status (This is cleared when the Watchdog is disabled or when a new value is written to WDT_VAL. Writing WDT_VAL would be the interrupt-acknowledge.)
- Watchdog interrupt IRQ select (from the same list of eight interrupts supported on the LPC SERIRQ)
- Interrupt enable
- Board reset enable (when set, the board will be reset when the Watchdog timer expires).

Industrial I/O Functions and SPI Interface

The EPM-43 employs a set of I/O registers for controlling external serial peripheral interface (SPI) devices. Refer to the descriptions of the SPICONTROL (page 15), SPISTATUS (page 16), and SPIDATA[0:3] (page 17) registers for more information.

The SPI bus specifies four logic signals:

- SCLK – Serial clock (output from master)
- MOSI – Master output, slave input (output from master)
- MISO – Master input, slave output (output from slave)
- SS – Slave select (output from master)

The EPM-43 SPI implementation adds additional features, such as hardware interrupt input to the master. The master initiates all SPI transactions. A slave device responds when its slave select is asserted and it receives clock pulses from the master.

Slave selects are controlled in one of two modes: manual or automatic. In automatic mode, the slave select is asserted by the SPI controller when the most significant data byte is written. This initiates a transaction to the specified slave device. In manual mode, the slave select is controlled by the user and any number of data frames can be sent. The user must command the slave select high to complete the transaction.

The SPI clock rate can be software configured to operate at speeds between 1 MHz and 8 MHz. All four common SPI modes are supported through the use of clock polarity and clock phase controls.

To initiate an SPI transaction, configure SPI registers SPICONTROL and SPISTATUS as shown in Table 12 and Table 13 for the desired I/O device. For additional information on communicating with specific SPI devices, refer to their respective manufacturer's datasheets.

Programmable LED

User I/O connector J4 includes an output signal for attaching a software controlled LED. Connect the cathode of the LED to J4, pin 16; connect the anode to +3.3 V. An on-board resistor limits the current when the circuit is turned on. A programmable LED is provided on the CBR-4005B paddleboard. Refer to the EPM-43 Hardware Reference Manual for the location of the Programmable LED on the CBR-4005B paddleboard.

To switch the PLED on and off, refer to Table 4: PCR – Product Code and LED Register, on page 10.

*** *End of document* ***