XPS Mailbox (v2.00a)

Introduction

In a multiprocessor environment, the processors need to communicate data with each other. The easiest method is to set up inter-processor communication through a mailbox. XPS Mailbox features a bi-directional communication channel between two processors. The XPS Mailbox can be connected to the processor either through PLB or FSL interface. The PLB interface option is available for the MicroBlaze™ processor, PowerPC® processor, or any other PLBv46 master. FSL option is available for direct connection to a MicroBlaze processor or any other FSL capable IP.

Features

- PLB interface is based on PLB v4.6 specification
- FSL interface is based on FSL v2.0 specification
- Configurable depth of mailbox
- Configurable interrupt thresholds and maskable interrupts
- Configurable synchronous or asynchronous operation
- Configurable interface, FSL or PLBv46, on each port
- Bi-directional communication

LogiCORE™ Facts		
Core Specifics		
Supported Device Family	Spartan®-3, Spartan-3E, Spartan-6, Spartan-3A/3A DSP/3AN, Automotive Spartan-3/3A/3A DSP/ 3E, Virtex®-4, Virtex-4Q, Virtex-4QV, Virtex-5, Virtex-6	
Version of core	xps_mailbox	v2.00a
Resources Used[1]		
	Min	Max
Slices	~50	~455
LUTs	~85	~365
FFs	~15	~330
Block RAMs	0	0
Special Features	None	
Provided with Core		
Documentation	Product Specification	
Design File Formats	VHDL	
Constraints File	N/A	
Verification	N/A	
Instantiation Template	N/A	
Reference Designs & application notes	N/A	
Design Tool Requirements		
Xilinx Implementation Tools	ISE® 11.1 or later	
Verification	N/A	
Simulation	ModelSim PE/SE 6.4b or later	
Synthesis	XST	
Support		
Provided by Xilinx, Inc.		

1. Resources for Virtex®-4 implementation. Minimum for 16 word deep and FSL only. Maximum for 16 deep and PLB only

Functional Description

The XPS Mailbox is used for bi-directional inter processor communication. A mailbox is a link between two otherwise separate processor systems. Other multi-port IP blocks, such as a memory controller etc., may also be shared by the two sub systems.

In addition to sending the actual data between processors the mailbox can be used to generate interrupts between the processors.

The XPS Mailbox in a typical PLBv46 system is shown in the top-level block diagram in Figure 1.

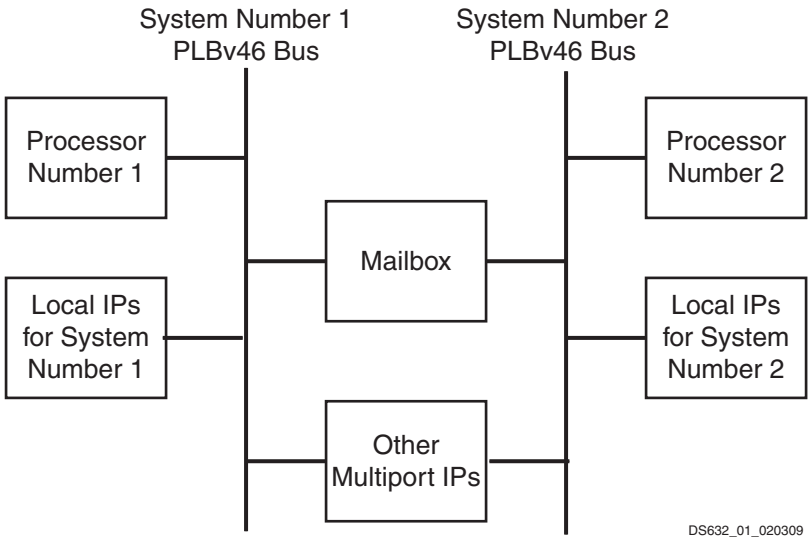

Figure 1: **XPS Mailbox FPGA in a PLBv46 System**

XPS Mailbox I/O Signals

The XPS Mailbox has two interfaces that are used to connect to the rest of the system. Both interfaces can be independently configured to use FSL or PLBv46. The signal descriptions are included in three tables:

1. The PLB signals are described inTable 1.

2. The FSL signals are described in Table 2.

3. The common signals are described in Table 3.

All signals in Table 1 and Table 2 apply to both interface sides; <x> denotes the interface number, which may be 0 or 1.

Table 1: **XPS Mailbox PLBv46 I/O Signal Description**

Port	Signal Name	Interface	I/O	Initial State	Description
System Signals					
P1	SPLB<x>_Clk	System	I	-	PLB clock
P2	SPLB<x>_Rst	System	I	-	PLB reset, active high
PLB Interface Signals					
P3	PLB<x>_ABus[0:31]	PLB	I	-	PLB address bus
P4	PLB<x>_PAValid	PLB	I	-	PLB primary address valid
P5	PLB<x>_masterID[0: C_SPLB_MID_WIDTH - 1]	PLB	I	-	PLB current master identifier
P6	PLB<x>_RNW	PLB	I	-	PLB read not write
P7	PLB<x>_BE[0: (C_SPLB_DWIDTH/8) - 1]	PLB	I	-	PLB byte enables
P8	PLB<x>_size[0:3]	PLB	I	-	PLB size of requested transfer
P9	PLB<x>_type[0:2]	PLB	I	-	PLB transfer type
P10	PLB<x>_wrDBus[0: C_SPLB_DWIDTH - 1]	PLB	I	-	PLB write data bus
Unused PLB Interface Signals					
P11	PLB<x>_UABus[0:31]	PLB	I	-	PLB upper address bits
P12	PLB<x>_SAValid	PLB	I	-	PLB secondary address valid
P13	PLB<x>_rdPrim	PLB	I	-	PLB secondary to primary read request indicator
P14	PLB<x>_wrPrim	PLB	I	-	PLB secondary to primary write request indicator
P15	PLB<x>_abort	PLB	I	-	PLB abort bus request
P16	PLB<x>_busLock	PLB	I	-	PLB bus lock
P17	PLB<x>_MSize[0:1]	PLB	I	-	PLB data bus width indicator
P18	PLB<x>_lockErr	PLB	I	-	PLB lock error
P19	PLB<x>_wrBurst	PLB	I	-	PLB burst write transfer
P20	PLB<x>_rdBurst	PLB	I	-	PLB burst read transfer
P21	PLB<x>_wrPendReq	PLB	I	-	PLB pending bus write request
P22	PLB<x>_rdPendReq	PLB	I	-	PLB pending bus read request
P23	PLB<x>_wrPendPri[0:1]	PLB	I	-	PLB pending write request priority
P24	PLB<x>_rdPendPri[0:1]	PLB	I	-	PLB pending read request priority
P25	PLB<x>_reqPri[0:1]	PLB	I	-	PLB current request priority
P26	PLB<x>_TAttribute[0:15]	PLB	I	-	PLB transfer attribute

Table 1: **XPS Mailbox PLBv46 I/O Signal Description *(Contd)***

Port	Signal Name	Interface	I/O	Initial State	Description
PLB Slave Interface Signals					
P27	SI<x>_addrAck	PLB	O	0	Slave address acknowledge
P28	SI<x>_SSize[0:1]	PLB	O	0	Slave data bus size
P29	SI<x>_wait	PLB	O	0	Slave wait
P30	SI<x>_rearbitrate	PLB	O	0	Slave bus rearbitrate
P31	SI<x>_wrDAck	PLB	O	0	Slave write data acknowledge
P32	SI<x>_wrComp	PLB	O	0	Slave write transfer complete
P33	SI<x>_rdDBus[0: C_SPLB_DWIDTH - 1]	PLB	O	0	Slave read data bus
P34	SI<x>_rdDAck	PLB	O	0	Slave read data acknowledge
P35	SI<x>_rdComp	PLB	O	0	Slave read transfer complete
P36	SI<x>_MBusy[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Slave busy
P37	SI<x>_MWrErr[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Slave write error
P38	SI<x>_MRdErr[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Slave read error
Unused PLB Slave Interface Signals					
P39	SI<x>_wrBTerm	PLB	O	0	Slave terminate write burst transfer
P40	SI<x>_rdWdAddr[0:3]	PLB	O	0	Slave read word address
P41	SI<x>_rdBTerm	PLB	O	0	Slave terminate read burst transfer
P42	SI<x>_MIRQ[0: C_SPLB_NUM_MASTERS - 1]	PLB	O	0	Master interrupt request

..

Table 2: **XPS Mailbox FSL I/O Signal Description**

Port	Signal Name	Interface	I/O	Initial State	Description
FSL Master Interface Signals					
P43	FSL<x>_M_Clk	MFSL	I	N/A	This port provides the input clock to the FSL master interface of the mailbox when used in the asynchronous FIFO mode (C_ASYNC_CLKS = 1). All transactions on the master interface use this clock when implemented in the asynchronous mode
P44	FSL<x>_M_Data	MFSL	I	0	The data input to the FSL master interface of the mailbox
P45	FSL<x>_M_Control	MFSL	I	0	Unused for mailbox
P46	FSL<x>_M_Write	MFSL	I	0	Input signal that controls the write enable signal of the FSL master interface of the FIFO. When set to 1, the value of FSL<x>_M_Data is pushed into the mailbox FIFO on a rising clock edge.
P47	FSL<x>_M_Full	MFSL	O	N/A	Output signal on the FSL master interface of the FIFO indicating that the FIFO is full.
FSL Slave Interface Signals					
P48	FSL<x>_S_Clk	SFSL	I	N/A	This port provides the input clock to the FSL slave interface on the mailbox when used in the asynchronous FIFO mode (C_ASYNC_CLKS = 1). All transactions on the slave interface use this clock when implemented in the asynchronous mode
P49	FSL<x>_S_Data	SFSL	O	N/A	The data output bus onto the FSL slave interface of the mailbox
P50	FSL<x>_S_Control	SFSL	O	N/A	Unused for mailbox
P51	FSL<x>_S_Read	SFSL	I	0	Input signal on the FSL slave interface that controls the read acknowledge signal of the FIFO. When set to 1, the value of FSL<x>_S_Data is popped from the FIFO on a rising clock edge.
P52	FSL<x>_S_Exists	SFSL	O	N/A	Output signal on the FSL slave interface indicating that FIFO contains valid data.

Table 3: **XPS Mailbox Common I/O Signal Description**

Port	Signal Name	Interface	I/O	Initial State	Description
FSL Common Interface Signals					
P53	FSL_Clk		I	N/A	This is the input clock to the mailbox when used in synchronous FIFO mode (C_ASYNC_CLKS = 0) and both interfaces are FSL based (C_INTERFACE_<x>_IS_FSL = 1). The FSL_Clk is in this case used to clock the core, in all other cases are the internal mailbox clock automatically derived from either SPLB<x>_Clk and/or FSL<x>_M_Clk/FSL<x>_S_Clk depending on the settings.
P54	SYS_Rst		I	N/A	External system reset. This signal is only required when both interfaces are configured to be FSL interfaces. If any PLB interface is available this signal is optional.
P55	FSL_Rst		O	0	Output reset signal generated by the FSL reset logic. Any peripherals connected to the FSL bus may use this reset signal to operate the peripheral reset.
Common Signals					
P56	Interrupt_0		O	0	Interrupt signal that data is available at interface 0
P57	Interrupt_1		O	0	Interrupt signal that data is available at interface 1

XPS Mailbox Design Parameters

To allow the user to obtain a XPS Mailbox that is uniquely tailored for the system, certain features can be parameterized in the XPS Mailbox design. This allows the user to configure a design that utilizes the resources required by the system only and that operates with the best possible performance. The features that can be parameterized in the XPS Mailbox design are as shown in Table 4. The PLB related generics, G3 through G10, are separately configured for each interface.

Table 4: **XPS Mailbox Design Parameters**

Generic	Feature/Description	Parameter Name	Allowable Values	Default Value	VHDL Type
System Parameter					
G1	Target FPGA family	C_FAMILY	spartan3, aspartan3, spartan3e, aspartan3e, spartan3a, aspartan3a, spartan3adsp, aspartan3adsp, spartan6, virtex4, qrvirtex4, qvirtex4, virtex5, virtex6	virtex-5	string
G2	Level of external reset	C_EXT_RESET_HIGH	0 or 1	1	integer
PLB Parameters					
G3	PLB Base Address	C_SPLB<x>_BASEADDR	Valid Address[1]	None[3]	std_logic_vector
G4	PLB High Address	C_SPLB<x>_HIGHADDR	Valid Address[2]	None[3]	std_logic_vector
G5	PLB least significant address bus width	C_SPLB<x>_AWIDTH	32	32	integer
G6	PLB data width	C_SPLB<x>_DWIDTH	32, 64, 128	32	integer
G7	Selects point-to-point or shared bus topology	C_SPLB<x>_P2P	0 = Shared Bus Topology 1 = Point-to-Point Bus Topology[4]	0	integer
G8	PLB Master ID Bus Width	C_SPLB<x>_MID_WIDTH	$\log_2$(C_SPLB_NUM_MASTERS) with a minimum value of 1	1	integer
G9	Number of PLB Masters	C_SPLB<x>_NUM_MASTERS	1 - 16	1	integer
G10	Support Bursts	C_SPLB<x>_SUPPORT_BURSTS	0	0	integer
G11	Width of the Slave Data Bus	C_SPLB_NATIVE_DWIDTH	32	32	integer
XPS Mailbox Parameters					
G12	Specify clocking modes of FIFO as synchronous or asynchronous	C_ASYNC_CLKS	0 - 1	0	Integer
G13	Use BRAMs to implement FIFO	C_IMPL_STYLE	0 - 1	1	Integer
G14	FSL bus width	C_FSL_DWIDTH	32	32	Integer

Table 4: **XPS Mailbox Design Parameters *(Contd)***

Generic	Feature/Description	Parameter Name	Allowable Values	Default Value	VHDL Type
G15	If interface 0 shall use FSL instead of PLB	C_INTERFACE_0_IS_FSL	0 - 1	0	Integer
G16	If interface 1shall use FSL instead of PLB	C_INTERFACE_1_IS_FSL	0 - 1	0	Integer
G17	FIFO depth of mailbox	C_MAILBOX_DEPTH	16 - 8192	16	Integer
G18	Read Clock period for interface 0 when asynchronous LUTRAM is used (in ps)	C_READ_CLOCK_PERIOD_0	> 0 when enabled	0	Integer
G19	Read Clock period for interface 1when asynchronous LUTRAM is used (in ps)	C_READ_CLOCK_PERIOD_0	> 0 when enabled	0	Integer

Notes:

1. The user must set the values. The C_BASEADDR must be a multiple of the range, where the range is C_HIGHADDR - C_BASEADDR + 1.
2. C_HIGHADDR - C_BASEADDR must be a power of 2 greater than equal to C_BASEADDR + 0x3F.
3. No default value will be specified to insure that the actual value is set, i.e., if the value is not set, a compiler error will be generated.
4. Value of '1' is not supported in this core.

Allowable Parameter Combinations

The address range specified by C_BASEADDR and C_HIGHADDR must be a power of 2, and must be at least 0x3F.

For example, if C_BASEADDR = 0xE0000000, C_HIGHADDR must be at least = 0xE000003F.

XPS Mailbox Parameter - Port Dependencies

The dependencies between the XPS Mailbox core design parameters and I/O signals are described in Table 5. In addition, when certain features is deselected, the related logic will no longer be a part of the design. The unused input and output signals are set to a specified value.

Table 5: **XPS Mailbox Parameter-Port Dependencies**

Generic or Port	Name	Affects	Depends	Relationship Description
Design Parameters				
G6	C_SPLB<x>_DWIDTH	P7, P10, P33	-	Affects the number of bits in data bus
G8	C_SPLB<x>_MID_WIDTH	P5	G9	This value is calculated as: $\log_2$(C_SPLB<x>_NUM_MASTERS) with a minimum value of 1
G9	C_SPLB<x>_NUM_MASTERS	P36, P37, P38, P42	-	Affects the number of PLB masters
G14	C_FSL_DWIDTH	P44, P49	-	Affects the number of bits in data bus
I/O Signals				
P5	PLB<x>_masterID[0: C_SPLB_MID_WIDTH - 1]	-	G8	Width of the PLB<x>_masterID varies according to C_SPLB<x>_MID_WIDTH
P7	PLB<x>_BE[0: (C_SPLB_DWIDTH/8) -1]	-	G6	Width of the PLB<x>_BE varies according to C_SPLB<x>_DWIDTH
P10	PLB<x>_wrDBus[0: C_SPLB_DWIDTH - 1]	-	G6	Width of the PLB<x>_wrDBus varies according to C_SPLB<x>_DWIDTH
P33	SI<x>_rdDBus[0: C_SPLB_DWIDTH - 1]	-	G6	Width of the SI<x>_rdDBus varies according to C_SPLB<x>_DWIDTH
P36	SI<x>_MBusy[0: C_SPLB_NUM_MASTERS - 1]	-	G9	Width of the SI<x>_MBusy varies according to C_SPLB<x>_NUM_MASTERS
P37	SI<x>_MWrErr[0: C_SPLB_NUM_MASTERS - 1]	-	G9	Width of the SI<x>_MWrErr varies according to C_SPLB<x>_NUM_MASTERS
P38	SI<x>_MRdErr[0: C_SPLB_NUM_MASTERS - 1]	-	G9	Width of the SI<x>_MRdErr varies according to C_SPLB<x>_NUM_MASTERS
P42	SI<x>_MIRQ[0: C_SPLB_NUM_MASTERS - 1]	-	G9	Width of the SI<x>_MIRQ varies according to C_SPLB<x>_NUM_MASTERS
P44	FSL<x>_M_Data		G14	Width of the FSL<x>_M_Data varies according to C_FSL_DWIDTH
P49	FSL<x>_S_Data		G14	Width of the FSL<x>_S_Data varies according to C_FSL_DWIDTH

XPS Mailbox Register Descriptions

Each interface of the XPS Mailbox core has the same set of information registers. The information at each interface is not identical but rather localized for that interface since the communication is bi-directional.

Table 6 shows all the XPS Mailbox registers and their addresses for the PLB case. Much of the information can be acquired for the FSL case with the FSL<x>_M_Full and FSL<x>_S_Exists flags.

Table 6: **XPS Mailbox Registers**

Base Address + Offset (hex)	Register Name	Access Type	Default Value (hex)	Description
C_BASEADDR + 0x0	WRDATA	Write	N/A	Write Data address. Write only.
C_BASEADDR + 0x4	Reserved	N/A	N/A	Reserved for future use
C_BASEADDR + 0x8	RDDATA	Read	N/A	Read Data address. Read only
C_BASEADDR + 0xC	Reserved	N/A	N/A	Reserved for future use
C_BASEADDR + 0x10	STATUS	Read	0x1	Status flags for mailbox. Read only.
C_BASEADDR + 0x14	ERROR	Read	0x0	Error flags, clear on read. Read only.
C_BASEADDR + 0x18	SIT	-	-	Send Interrupt Threshold. Read/Write
C_BASEADDR + 0x1C	RIT	-	-	Receive Interrupt Threshold. Read/Write
C_BASEADDR + 0x20	IS	-	-	Interrupt Status register. Read/Write
C_BASEADDR + 0x24	IE	-	-	Interrupt Enable register. Read/Write
C_BASEADDR + 0x28	IP	-	-	Interrupt Pending register. Read only
C_BASEADDR + 0x2C	Reserved	-	-	Reserved for future use
C_BASEADDR + 0x30	Reserved	-	-	Reserved for future use
C_BASEADDR + 0x34	Reserved	-	-	Reserved for future use
C_BASEADDR + 0x38	Reserved	-	-	Reserved for future use
C_BASEADDR + 0x3C	Reserved	-	-	Reserved for future use

XPS Mailbox Write Data Register (WRDATA)

Writing to this register will result in the data transferred to the RDDATA register at the other interface. Trying to write while the full flag is set will result in an error and the FULL_ERROR bit will be set. The register is write only and a read request issued to WRDATA will be ignored. Bit assignment in the WRDATA register is described in Table 8.

Table 7: **Write Data Register**

WRDATA	
0	C_FSL_DWIDTH-1

Table 8: **XPS Mailbox Write Data Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - C_FSL_DWIDTH - 1	WRDATA	Write	-	Write register to send data to the other interface

XPS Mailbox Read Data Register (RDDATA)

Reading from this register will pop one value from the mail FIFO. Trying to read while the empty flag is set will result in an error and the EMPTY_ERROR bit will be set. The register is read only and a write request issued to RDDATA will be ignored. Bit assignment in the RDDATA register is described in Table 10.

Table 9: **Read Data Register**

RDDATA	
0	C_FSL_DWIDTH-1

Table 10: **XPS Mailbox Read Data Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - C_FSL_DWIDTH - 1	RDDATA	Read	-	Read register to get data word sent from the other interface

XPS Mailbox Status Register (STATUS)

The XPS Mailbox Status Register contains the current status of the mailbox. The register is read only and a write request issued to STATUS will be ignored. Bit assignment in the STATUS register is described in Table 12

Table 11: **Status Register**

Reserved		RTA	STA	Full	Empty
0	27	28	29	30	31

Table 12: **XPS Mailbox Status Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - 27	Reserved			Reserved for future use
28	RTA	Read	0	Receive Threshold Active indicates the current FIFO status of this interface in the receive direction '0' = The receive FIFO level is less than or equal to the RIT threshold '1' = The receive FIFO level is greater than the RIT threshold
29	STA	Read	0	Send Threshold Active indicates the current FIFO status of this interface in the send direction '0' = The send FIFO level is greater than the SIT threshold '1' = The send FIFO level is less than or equal to the SIT threshold
30	Full	Read	'0'	Indicates the current status of this interface in the send direction '0' = There is room for more data '1' = The FIFO is full, any attempts to write data will be ignored and generate an error
31	Empty	Read	'1'	Indicates the current status of this interface in the receive direction '0' = There is data available '1' = The FIFO is empty, any attempts to read data will be ignored and generate an error

XPS Mailbox Error Register (ERROR)

The XPS Mailbox Error Register contains the error flags for PLB accesses from this interface. The error register will be cleared at read, this means that all bits are sticky and that they indicate any errors that occurred since last time the error register was read. The register is read only and a write request issued to ERROR will be ignored. Bit assignment in the ERROR register is described in Table 14.

Table 13: **Error Register**

Reserved		Full Error	Empty Error
0	29	30	31

Table 14: **XPS Mailbox Error Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - 29	Reserved			Reserved for future use
30	Full Error	Read	'0'	Indicates if there has been any attempts to write to the WRDATA register while the Full flag was asserted since the error register was last read '0' = No error has occurred '1' = One or more attempts to write while FSL link was full
31	Empty Error	Read	'0'	Indicates if there has been any attempts to read from the RDDATA register while the Empty flag was asserted since the error register was last read '0' = No error has occurred '1' = One or more attempts to read while FSL link was empty

XPS Mailbox Send Interrupt Threshold Register (SIT)

The XPS Mailbox Send Interrupt Threshold Register contains the interrupt threshold for this interface in the send direction. Depending on the send FIFO data level writing a new SIT can cause a rising edge on STA that can generate a STI interrupt if it is enabled in the IE register. Bit assignment in the SIT register is described in Table 16.

Table 15: **SIT Register**

	SIT	
0	32-Log2(C_MAILBOX_DEPTH)	31

Table 16: **XPS Mailbox SIT Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
Log2(C_MAILBOX_DEPTH)	SIT	Read/Write	0	Lower Log2(C_MAILBOX_DEPTH) bits used, right justified to bit 31

XPS Mailbox Receive Interrupt Threshold Register (RIT)

The XPS Mailbox Receive Interrupt Threshold Register contains the interrupt threshold for this interface in the receive direction. Depending on the receive FIFO data level writing a new RIT can cause a rising edge on RTA that can generate a RTI interrupt if it is enabled in the IE register. Bit assignment in the RIT register is described in Table 18.

Table 17: **RIT Register**

	RIT	
0	32-Log2(C_MAILBOX_DEPTH)	31

Table 18: **XPS Mailbox RIT Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
Log2(C_MAILBOX_DEPTH)	RIT	Read/Write	0	Lower Log2(C_MAILBOX_DEPTH) bits used, right justified to bit 31

XPS Mailbox Interrupt Status Register (IS)

The XPS Mailbox Interrupt Status Register contains the current interrupt status for this interface. There are three types of interrupts that can be generated. Mailbox Error interrupt are generated when any of the bits in the ERROR register is set. The other two interrupts are FIFO related: RTI is generated for a rising edge on the RTA bit in the STATUS register and STI that is generated for a rising edge on the STA STATUS register bit. RTI and STI are used to indicate that it is time to read from or write to the FIFOs to avoid any stalls in the data flow. Bit assignment in the IS register is described in Table 20.

Table 19: **IS Register**

Reserved		ERR	RTI	STI
0	28	29	30	31

Table 20: **XPS Mailbox IS Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - 28	Reserved			Reserved for future use
29	ERR	Read/Write	'0'	Mailbox Error Interrupt Status for this interface. Values for read: '0' = No interrupt event has occurred. '1' = A Mailbox error has occurred. Values for write: '0' = No change '1' = Acknowledge and clear the interrupt if it is active
30	RTI	Read/Write	'0'	Mailbox Receive Threshold Interrupt pending status for this interface. Values for read: '0' = No interrupt event has occurred. '1' = Data level in the receive FIFO has caused a RTI. Values for write: '0' = No change '1' = Acknowledge and clear the interrupt if it is active
31	STI	Read/Write	'0'	Mailbox Send Threshold Interrupt penging status for this interface. Values for read: '0' = No interrupt event has occurred. '1' = Data level in the send FIFO has caused a STI. Values for write: '0' = No change '1' = Acknowledge and clear the interrupt if it is active

XPS Mailbox Interrupt Enable Register (IE)

The XPS Mailbox Interrupt Enable Register contains the mask for the allowed interrupts on this interface. Bit assignment in the IE register is described in Table 22.

Table 21: **IE Register**

Reserved		ERR	RTI	STI
0	28	29	30	31

Table 22: **XPS Mailbox IE Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - 28	Reserved			Reserved for future use
29	ERR	Read/Write	'0'	Mailbox Error Interrupt Enable for this interface '0' = ERR interrupt is disabled '1' = ERR interrupt is enabled
30	RTI	Read/Write	'0'	Mailbox Receive Threshold Interrupt Enable for this interface '0' = RTI interrupt is disabled '1' = RTI interrupt is enabled
31	STI	Read/Write	'0'	Mailbox Send Threshold Interrupt Enable for this interface '0' = STI interrupt is disabled '1' = STI interrupt is enabled

XPS Mailbox Interrupt Pending Register (IP)

The XPS Mailbox Interrupt Pending Register contains the currently pending interrupts from this interface. It is a read only register generated by performing bit-wise AND between the IS and IE registers. A write request issued to IP will be ignored. Bit assignment in the IP register is described in Table 24. All the bits in this register are ORed together to generate the interrupt output signal for this interface. When an interrupt has been serviced it is acknowledge by writing the corresponding bit to the IS Register.

Table 23: **IP Register**

Reserved		ERR	RTI	STI
0	28	29	30	31

Table 24: **XPS Mailbox IP Register Bit Definitions**

Bit(s)	Name	Core Access	Reset Value	Description
0 - 28	Reserved			Reserved for future use
29	ERR	Read	'0'	Mailbox Error Interrupt Pending status for this interface '0' = No pending interrupt '1' = Pending interrupt for Mailbox errors
30	RTI	Read	'0'	Mailbox Receive Threshold Interrupt Pending status for this interface '0' = No pending interrupt '1' = Pending interrupt for data level in receive FIFO
31	STI	Read	'0'	Mailbox Send Threshold Interrupt Pending status for this interface '0' = No pending interrupt '1' = Pending interrupt for data level in send FIFO

Design Implementation

Target Technology

The intended target technology is an FPGA in one of the following families: Spartan-3, Spartan-3E, Spartan-3A, Spartan-3A DSP, Automotive Spartan-3/3A/3E/3A DSP, Virtex-4, Virtex-4QV, Virtex-4Q, Virtex-5, Virtex-6.

Support

Xilinx provides technical support for this LogiCORE product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

Reference Documents

1. IBM CoreConnect 128-Bit Processor Local Bus, Architectural Specification (v4.6).

Revision History

Date	Version	Revision
08/31/2007	1.0	Initial Xilinx release.
4/24/09	1.1	Replaced references to supported device families and tool name(s) with hyperlink to PDF file.
6/26/09	1.2	Updated for EDK_L 11.2; created v2.00a.

Notice of Disclaimer