Xilinx Standalone Library Documentation

XilMailbox Library v1.0

UG1367 (2019.1) May 15, 2019

Table of Contents

Chapter 1

XilMailbox

Overview

The XilMailbox library provides the top-level hooks for sending or receiving an inter-processor interrupt (IPI) message using the Zynq® UltraScale+™ MPSoC IPI hardware.

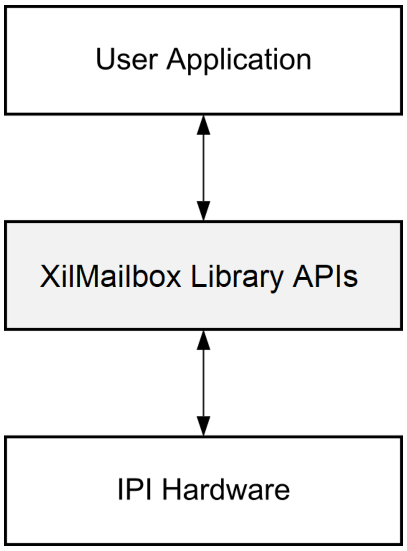

Figure 1.1: Overview

For more details on the IPI interrupts, see the Zynq UltraScale+ MPSoC Technical Reference Manual (UG1085). This library supports the following features:

- Triggering an IPI to a remote agent.
- Sending an IPI message to a remote agent.
- Callbacks for error and recv IPI events.
- Reading an IPI message.

Software Initialization

1. XMailbox_Initialize() function initializes a library instance for the given IPI channel.
2. XMailbox_Send() function triggers an IPI to a remote agent.
3.

Parameters

XMbox_IPI_Send	Triggers an IPI to a destination CPU
XMbox_IPI_SendData	Sends an IPI message to a destination CPU
XMbox_IPI_Recv	Reads an IPI message
RecvHandler	Callback for receive IPI event
ErrorHandler	Callback for error event
ErroRef	To be passed to the error interrupt callback
RecvRef	To be passed to the receive interrupt callback.
Agent	Used to store IPI Channel information.

Enumeration Type Documentation

enum XMailbox_Handler

This typedef contains XMAILBOX Handler Types.

Enumerator

XMAILBOX_RECV_HANDLER	For Recv Handler.
XMAILBOX_ERROR_HANDLER	For Error Handler.

Function Documentation

u32 XMailbox_Send (XMailbox ∗ *InstancePtr,* u32 *RemoteId,* u8 *Is_Blocking*)

This function triggers an IPI to a destination CPU.

Parameters

InstancePtr	Pointer to the XMailbox instance
RemoteId	Mask of the CPU to which IPI is to be triggered
Is_Blocking	If set, triggers notification in the blocking mode

Returns

- XST_SUCCESS if successful
- XST_FAILURE if unsuccessful

u32 XMailbox_SendData (XMailbox ∗ *InstancePtr*, u32 *RemoteId*, void ∗ *BufferPtr*, u32 *MsgLen*, u8 *BufferType*, u8 *Is_Blocking*)

This function sends an IPI message to a destination CPU.

Parameters

InstancePtr	Pointer to the XMailbox instance
RemoteId	Mask of the CPU to which IPI is to be triggered
BufferPtr	Pointer to Buffer which contains the message to be sent
MsgLen	Length of the buffer/message
BufferType	Type of buffer (XILMBOX_MSG_TYPE_REQ (OR) XILMBOX_MSG_TYPE_RESP)
Is_Blocking	If set, triggers the notification in blocking mode

Returns

- XST_SUCCESS if successful
- XST_FAILURE if unsuccessful

u32 XMailbox_Recv (XMailbox ∗ *InstancePtr*, u32 *SourceId*, void ∗ *BufferPtr*, u32 *MsgLen*, u8 *BufferType*)

This function reads an IPI message.

Parameters

InstancePtr	Pointer to the XMailbox instance
SourceId	Mask for the CPU which has sent the message
BufferPtr	Pointer to Buffer to which the read message needs to be stored
MsgLen	Length of the buffer/message
BufferType	Type of buffer (XILMBOX_MSG_TYPE_REQ or XILMBOX_MSG_TYPE_RESP)

Returns

- XST_SUCCESS if successful
- XST_FAILURE if unsuccessful

s32 XMailbox_SetCallBack (XMailbox ∗ *InstancePtr*, XMailbox_Handler *HandlerType*, void ∗ *CallBackFuncPtr*, void ∗ *CallBackRefPtr*)

This routine installs an asynchronous callback function for the given HandlerType.

HandlerType	Callback Function Type
XMAILBOX_RECV_HANDLER	Recv handler
XMAILBOX_ERROR_HANDLER	Error handler

Parameters

InstancePtr	Pointer to the XMailbox instance
HandlerType	Specifies which callback is to be attached
CallBackFunc	Address of the callback function
CallBackRef	User data item that will be passed to the callback function when it is invoked

Returns

- XST_SUCCESS when handler is installed.
- XST_INVALID_PARAM when HandlerType is invalid.

Note

Invoking this function for a handler that already has been installed replaces it with the new handler.

u32 XMailbox_Initialize (XMailbox ∗ *InstancePtr*, u8 *DeviceId*)

Initialize the XMailbox Instance.

Parameters

InstancePtr	is a pointer to the instance to be worked on
DeviceId	is the IPI Instance to be worked on

Returns

XST_SUCCESS if initialization was successful XST_FAILURE in case of failure

Appendix A

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support .

Solution Centers

See the Xilinx Solution Centers for support on devices, software tools, and intellectual property at all stages of the design cycle. Topics include design assistance, advisories, and troubleshooting tips.

Please Read: Important Legal Notices