Avalon-MM DMA FIFO Introduction

2014.12.15.

UG-01159 Subscribe Send Feedback

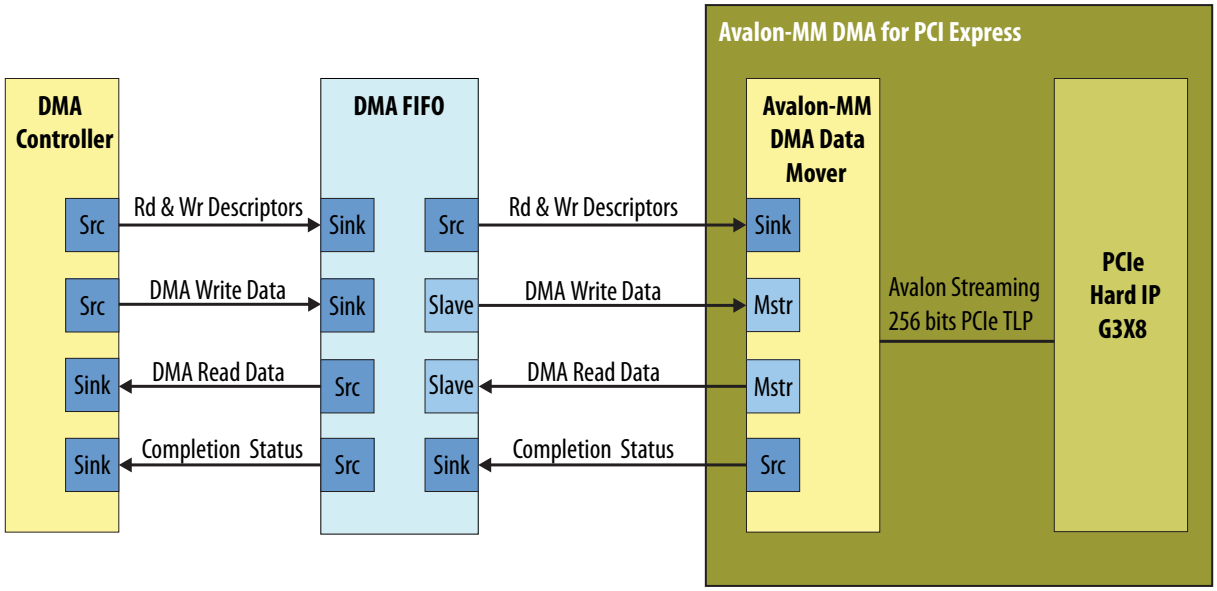
The Avalon® Memory-Mapped (Avalon-MM) DMA FIFO Example Design provides a FIFO interface to the Data Mover in the V-Series Avalon-MM DMA for PCI Express® IP core. This component is part of a system design that includes a custom, third-party, external descriptor controller. It is available for the V-Series Avalon-MM DMA for PCI Express IP core with a 256-bit interface to the Application Layer.

The DMA FIFO passes the descriptors it receives to the Data Mover requires. The DMA FIFO drives completion packets to the DMA Controller on its Avalon-ST source interface. An internal 4 kilobyte (KB) reordering buffer for each descriptor ensures that read completion data is assembled in the correct order. Each

Figure 1: DMA FIFO Example Design System-Level Block Diagram

This figure uses the following abbreviations:

- Src—Avalon-ST source interface
- Sink—Avalon-ST sink interface
- Mstr—Avalon-MM master interface
- Slave—Avalon-MM slave interface



The DMA FIFO supports the following operations:

- Accepts up to four read descriptors
- Transfers write descriptors that write up to 512 KB of data per descriptor
- Transfers read descriptors that read up to 4 KB of data per descriptor
- Drives completion status on the Avalon-ST completion status interface

Note: The DMA FIFO Example Design is preliminary in the Quartus II 14.1 release. Both the programming model and top-level signals may change in subsequent releases.

Device Family Support

Table 1: Device Family Support

Device Family	Support
Stratix V	Final. The IP core is verified with final timing models. The IP core meets all functional and timing requirements for the device family and can be used in production designs.
Other device families	No support.

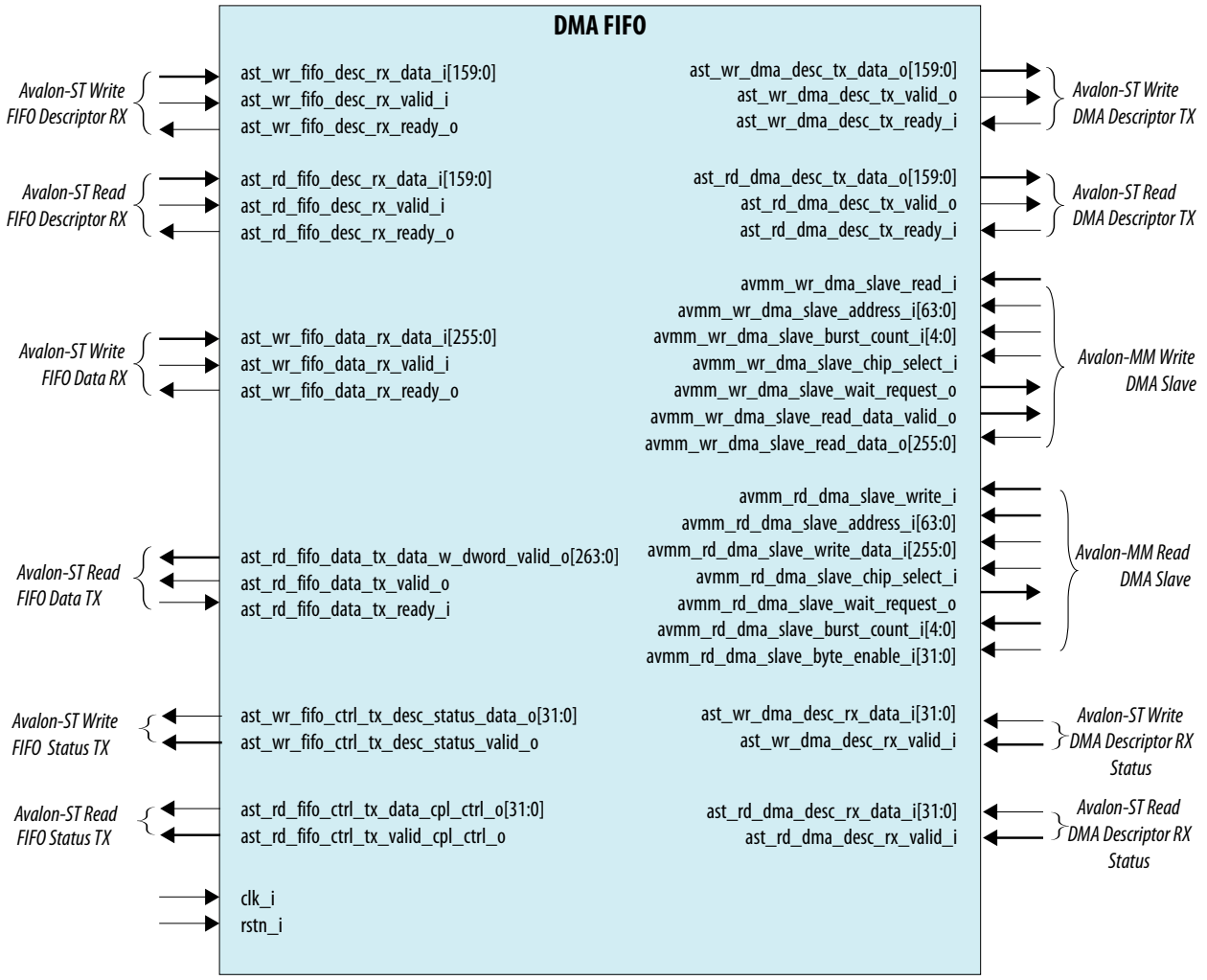
DMA FIFO Parameters

The DMA FIFO has no parameters.

DMA FIFO Interfaces

The DMA FIFO connection to a DMA controller comprises six Avalon-ST interfaces to transfer read and write descriptors, data, and status. The DMA FIFO connection to the Data Mover also comprises six interfaces. Four Avalon-ST interfaces transfer read and write descriptors and status. Two Avalon-MM interfaces transfer DMA data.

Figure 2: DMA FIFO Top-Level Interfaces



Avalon-ST Write and Read FIFO Descriptors

The DMA FIFO receives write and read FIFO descriptors from the DMA controller on its Avalon-ST sink interfaces. The DMA FIFO transmits the descriptors to the Data Mover on its Avalon-ST source interfaces. All interfaces are synchronous to the input clock, `clk_i`.

Table 2: Avalon-ST Write FIFO Descriptor RX

Signal Name	Direction	Definition
`ast_wr_fifo_desc_rx_data_i[159:0]`	Input	Specifies the write descriptor data. Refer **Table 6** *DMA Descriptor Format* for the format of this data.
`ast_wr_fifo_desc_rx_valid_i`	Input	When asserted, `ast_wr_fifo_desc_rx_data_i[159:0]` is valid. Readt latency on this interface is 0 cycles. The DMA FIFO stores data on the rising edge of `clk_i` when both `ast_wr_fifo_desc_rx_valid_i` and `ast_wr_fifo_desc_rx_ready_o` are asserted.

Signal Name	Direction	Definition
ast_wr_fifo_desc_rx_ready_o	Output	When asserted, the FIFO can accept the write descriptor data. The ready latency is 0 cycles. Consequently, the DMA FIFO stores data on the rising edge of clk_i when both the ast_wr_fifo_desc_rx_valid_i and ast_wr_fifo_desc_rx_read_o are asserted.

Table 3: Avalon-ST Read FIFO Descriptor RX

Signal Name	Direction	Definition
ast_rd_fifo_desc_rx_data_i[159:0]	Input	Specifies the read descriptor data. Refer to **Table 6** *DMA Descriptor Format* for the format of this data.
ast_rd_fifo_desc_rx_valid_i	Input	When asserted, ast_rd_fifo_desc_rx_data_i[159:0] is valid. Ready latency on this interface is 0 cycles. The DMA FIFO stores data on the rising edge of clk_i when both ast_rd_fifo_desc_rx_valid_i and ast_rd_fifo_desc_rx_ready_o are asserted.
ast_rd_fifo_desc_rx_ready_o	Output	When asserted, the FIFO stores the read descriptor data. The ready latency is 0 cycles. The DMA FIFO stores data on the rising edge of clk_i whenever both the ast_rd_fifo_desc_rx_valid_i and ast_rd_fifo_desc_rx_ready_o are asserted.

Table 4: Avalon-ST Write DMA Descriptor TX

Signal Name	Direction	Definition
ast_wr_dma_desc_tx_data_o[159:0]	Output	Specifies the write descriptor data. Refer **Table 6** *DMA Descriptor Format* for the format of this data.
ast_wr_dma_desc_tx_valid_o	Output	When asserted, ast_wr_dma_desc_tx_data_o[159:0] is valid.
ast_wr_dma_desc_tx_ready_i	Input	When asserted, the Data Mover stores the write descriptor data. The ready latency is 3 cycles. Consequently, the DMA FIFO transmits valid data 3 cycles after ast_wr_dma_desc_tx_ready_i is asserted.

Table 5: Avalon-ST Read DMA Descriptor TX

Signal Name	Direction	Definition
ast_rd_dma_desc_tx_data_o[159:0]	Input	Specifies the read descriptor data. Refer **Table 6** *DMA Descriptor Format* for the format of this data.
ast_rd_dma_desc_tx_valid_o	Input	When asserted, ast_rd_dma_desc_tx_data_o[159:0] is valid.

Signal Name	Direction	Definition
ast_rd_dma_desc_tx_ready_i	Output	When asserted, the Data Mover stores the read descriptor data. The ready latency is 3 cycles. Consequently, the DMA FIFO transmits valid data 3 cycles after ast_rd_dma_desc_tx_ready_i is asserted.

Table 6: DMA Descriptor Format

Bits	Name	Description
[31:0]	Source Low Address	Low-order 32 bits of the DMA source address. The address boundary must align to the 32 bits so that the 2 least significant bits are 2'b00. For reads, the source address is the PCIe domain address. For writes, this is the Avalon-MM domain address as seen by the Data Mover. Refer to **Figure 1** *DMA FIFO Example Design System-Level Block Diagram*.
[63:32]	Source High Address	High-order 32 bits of the source address.
[95:64]	Destination Low Address	Low-order 32 bits of the DMA destination address. The address boundary must align to 32 bits so that the 2 least significant bits have the value of 2'b00. For reads, the destination address is the Avalon-MM domain address as seen by the Data Mover Avalon-MM read master to the DMA FIFO Avalon-MM read slave . For writes, the destination address is the PCIe domain address. Refer to **Figure 1** *DMA FIFO Example Design System-Level Block Diagram*.
[127:96]	Destination High Address	High-order 32 bits of the destination address.
[145:128]	DMA Length	Specifies data length in dwords. The length must be greater than 0. The maximum length for write data is 512 KB. The maximum length for read data in 8 KB.
[153:146]	DMA Descriptor ID	Specifies the descriptor ID. The DMA FIFO can process up to 4 outstanding descriptors.
[159:154]	Reserved	N/A

DMA Write and Read Operation

The DMA FIFO receives write FIFO data on its Avalon-ST sink interface. The Data Mover Avalon-MM master interface reads this data from the FIFO's Avalon-MM Write DMA Slave interface using the source address specified in ast_wr_dma_desc_tx_data_o.

Table 7: Avalon-ST Write FIFO Data RX

Signal Name	Direction	Definition
ast_wr_dma_fifo_data_rx_data[255:0]	Input	The DMA FIFO receives data on this bus.
avst_wr_fifo_data_rx_valid_i	Input	Asserted when the ast_wr_dma_fifo_data_rx_data[255:0] is valid. The ready latency on this interface is 0 cycles. The DMA FIFO stores data on the rising edge of clock when ast_wr_fifo_data_rx_ready_o and avst_wr_fifo_data_rx_valid_i are both asserted.
ast_wr_fifo_data_rx_ready_o	Output	Asserted when the DMA FIFO write queue can accept write data.

Table 8: Avalon-MM Write DMA Slave

Signal Name	Direction	Definition
avmm_wr_dma_slave_read_i	Input	The Data Mover asserts avmm_wr_dma_slave_read_i to read data from the DMA FIFO. It transmits this data to the source address specified in the write descriptor.
avmm_wr_dma_slave_address_i[63:0]	Input	The Data Mover specifies avmm_wr_dma_slave_address_i[63:0]. This is the address offset of the DMA FIFO's write FIFO in the Avalon address domain.
avmm_wr_dma_slave_burst_count_i[4:0]	Input	Specifies the size of the burst per write operation. DMA FIFO supports a write of up to 512 Kbytes.
avmm_wr_dma_slave_chip_select_i[4:0]	Input	When asserted, the Data Mover is accessing the Avalon-MM Write DMA Slave interface.
avmm_wr_dma_slave_wait_request_o	Output	When asserted, this slave interface is not ready to respond. If the Data Mover is accessing this interface, it must hold the signals constant until avmm_wr_dma_slave_wait_request_o deasserts.
avmm_wr_dma_slave_read_data_valid_o	Output	When asserted, avmm_wr_dma_slave_read_data_o[255:0] is valid.
avmm_wr_dma_slave_read_data_o[255:0]	Output	Drives FIFO data to the Data Mover.

Table 9: Avalon-ST Read FIFO Data TX

Signal Name	Direction	Definition
ast_rd_dma_fifo_data_tx_data_w_dword_valid_o[263:0]	Output	The DMA FIFO drives data on this bus.The low-order 8 bits specify valid dwords. The DMA Controller can use ast_rd_dma_fifo_data_tx_data_w_dword_valid_o[7:0] to align data based of the first and last valid dwords of the 256-bit data.

Signal Name	Direction	Definition
ast_rd_fifo_data_tx_valid_o	Output	Asserted when the ast_rd_dma_fifo_data_tx_data_w_dword_valid_o[263:0] is valid. The ready latency on this interface is 0 cycles. The DMA FIFO stores data on the rising edge clk_i when ast_rd_fifo_data_tx_ready_i and ast_rd_dma_fifo_data_tx_data_w_dword_valid_o[263:0] are both asserted.
ast_rd_fifo_data_tx_ready_i	Input	Asserted when the DMA Controller can accept read data.

Table 10: Avalon-MM Read DMA Slave

The DMA FIFO receives read data from the Data Mover on its Avalon-MM slave read interface. It drives this data to the DMA Controller on its DMA read Avalon-ST source interface.

Signal Name	Direction	Definition
avmm_rd_dma_slave_write_i	Input	The Data Mover asserts avmm_rd_dma_slave_write_i to write data to the DMA FIFO.
avmm_rd_dma_slave_address_i[63:0]	Input	The Data Mover specifies avmm_rd_dma_slave_address_i[63:0]. This is the address offset of the DMA FIFO's read FIFO in the Avalon address domain.
avmm_rd_dma_slave_read_data_o[255:0]	Input	Receives data from the PCIe address domain.
avmm_rd_dma_slave_burst_count_i[4:0]	Input	Specifies the size of the read data.
avmm_rd_dma_slave_chip_select_i[4:0]	Input	When asserted, the Data Mover is accessing the Avalon-MM Read DMA Slave interface.
avmm_rd_dma_slave_wait_request_o	Output	When asserted, this DMA FIFO read data FIFO is full. The FIFO stores completion data for up to 4, 4 Kbyte read descriptors. Internal logic restores the correct order for out-of-order read completion data.
avmm_rd_dma_slave_read_burst_count[4:0]	Input	Specifies the size of the read burst count.
avmm_rd_dma_slave_byte_enable[31:0]	Input	Specifies the bytes of avmm_rd_dma_slave_read_data_o[255:0] that are valid.

Write DMA and Read DMA Completion Status

The DMA FIFO receives the completion status from the Data Mover on the Descriptor RX Status interfaces. It drives this information to the DMA controller on the FIFO Status TX interfaces.

Table 11: Avalon-ST Write FIFO Status TX

Signal Name	Direction	Definition
ast_wr_fifo_ctrl_tx_desc_status_data_o[31:0]	Output	Indicates the status for the descriptor ID specified. The following fields are defined: • [31:9]: Reserved. • [8]: Done. When 1'b1, indicates successful completion of when Descriptor ID when ast_wr_fifo_ctrl_tx_desc_status_valid_o is asserted. • [7:0]: Descriptor ID.
ast_wr_fifo_ctrl_tx_desc_status_valid_o	Output	When asserted, the data on ast_wr_fifo_ctrl_tx_desc_status_data_o[31:0] is valid.

Table 12: Avalon-ST Read FIFO Status TX

Signal Name	Direction	Definition
ast_rd_fifo_ctrl_tx_cpl_ctrl_o[31:0]	Output	Indicates status of the descriptor ID specified. The following fields are defined: • [31:9]: Reserved. Must be all 0s. • [8]: Done. When 1'b1, indicates successful completion of Descriptor ID when ast_rd_fifo_ctrl_tx_valid_cpl_ctrl_o is asserted. • [7:0]: Descriptor ID.
ast_rd_fifo_ctrl_tx_valid_cpl_ctrl_o	Output	When asserted, ast_rd_fifo_ctrl_tx_cpl_ctrl_o[31:0] is valid.

Table 13: Avalon-ST Write DMA Descriptor RX Status

Signal Name	Direction	Definition
ast_wr_dma_desc_rx_data_i[31:0]	Input	Specifies the status for the descriptor ID specified. The following fields are defined: • [31:9]: Reserved. Must be all 0s. • [8]: Done. When 1'b1, indicates successful completion of the Descriptor ID when ast_wr_dma_desc_rx_valid_i is asserted. • [7:0]: Descriptor ID
ast_wr_dma_desc_rx_valid_i	Input	When asserted, the data on ast_wr_dma_desc_rx_data_i[31:0] is valid.

Table 14: Avalon-ST Read DMA Descriptor RX Status

Signal Name	Direction	Definition
ast_rd_dma_desc_rx_data_i[31:0]	Input	Specifies the status for the descriptor ID specified. The following fields are defined: • [31:9]: Reserved. Must be all 0s. • [8]: Done. When 1'b1, indicates successful completion of the Descriptor ID when ast_rd_dma_desc_rx_valid_i is asserted. • [7:0]: Descriptor ID.
ast_rd_dma_desc_rx_valid_i	Input	When asserted, the data on ast_rd_dma_desc_rx_data_i[31:0] is valid.

Clocks and Reset Interfaces

Table 15: Clock and Reset

	Direction	
clk_i	Input	Input clock. Connect the same clock source that drives the input clock of the Avalon-MM DMA for PCI Express IP core. Its frequency is either 125 MHz or 250 MHz.
rstn_i	Input	Active low reset signal. Connect the same reset signal that drives the reset input of the Avalon-MM DMA for PCI Express IP core.

Revision History

Date	Version	Changes
2014.12.15	14.1	• Initial Release.