2 Design description

Due to the cross-stack AXI transaction hang issue described in Section 7.6, there are effectively two variants of this example FPGA design: one for production silicon, and another for ES1. These two variants are described separately in the following two sections.

2.1 Design variant for production silicon

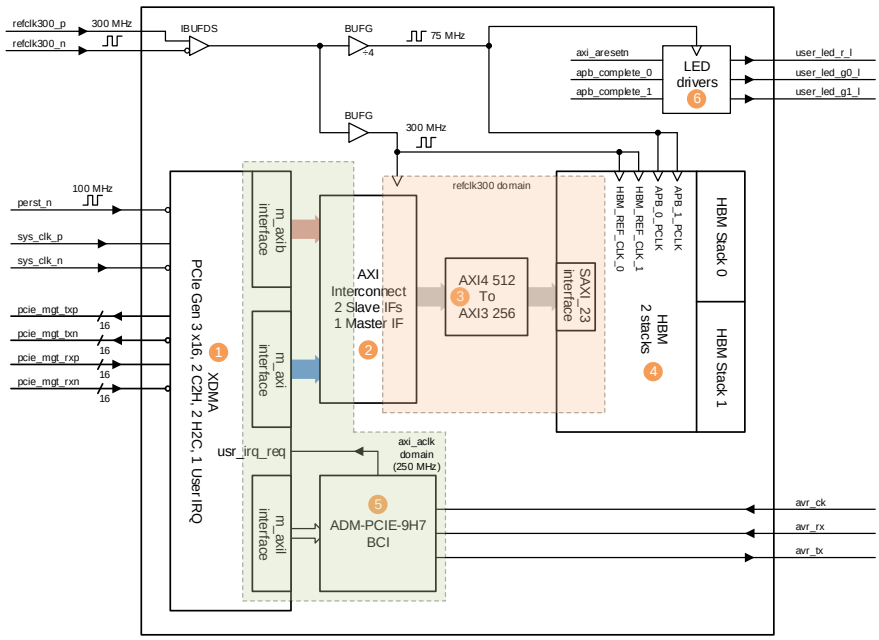

Figure 2 : Block diagram of the Host Interface to HBM FPGA Design

The FPGA design is implemented as a Vivado Block Diagram, consisting of the following elements:

(1) An instance of the Xilinx **XDMA** IP, configured to include a PCI Express to AXI4 Bridge, two C2H DMA engines and two H2C DMA engines.

(2) An instance of the Xilinx **AXI Interconnect** IP, which allows both the DMA engines and the CPU (via the XDMA's PCI Express to AXI4 Bridge) to access the HBM.

(3) An instance of the Xilinx **AXI Width Converter** IP together with an instance of the Xilinx **AXI Protocol Converter** IP, which downsizes the 512-bit wide data from the XDMA instance to 256 bits and converts the protocol from AXI4 to AXI3, to match the requirements of the HBM IP's **SAXI_23** interface.

(4) An instance of the Xilinx **HBM** IP, configured to include both stacks of HBM that are present in the VU35P / VU37P FPGA, for a total of 8 GiB. This is a memory controller with an AXI3 interface.

The HBM IP instance is configured to enable a single slave AXI3 interface, namely **SAXI_23**. This interface has been chosen because its physical location in the FPGA. For further explanation of this choice, see Section 2.1.1.

(5) An instance of the Alpha Data **ADM-PCIE-9H7 BCI** IP (Board Control Interface), which allows the FPGA

on an ADM-PCIE-9H7, when configured with this FPGA design's bitstream, to be compatible with Alpha Data's **avr2util** utility.

Note that the BCI instance is included only for compatibility with **avr2util**, and could be removed without affecting the functionality of the rest of the FPGA design.

(6) Three user-definable LEDs in the ADM-PCIE-9H7 are used to visually indicate (a) whether or not the **axi_aresetn** signal of the XDMA IP instance is asserted and (b) whether or not each HBM stack has completed its calibration. For details, refer to Appendix A.

2.1.1 HBM AXI slave interface locations

In the block diagram depicted in Figure 2, the HBM IP instance is configured to enable its **SAXI_23** interface, as opposed to a more obvious choice such as **SAXI_00**.

The reason for this choice is its physical location within the FPGA; Figure 3 below shows the layout of the implemented FPGA design and the locations of some key primitives in Super Logic Region 0 (SLR0):

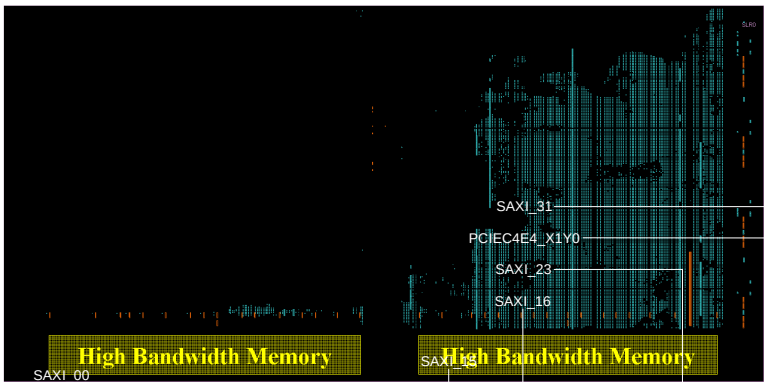

Figure 3 : Locations of key primitives in SLR0

The **SAXI_00** interface is at the extreme left, whereas **SAXI_31** is at the extreme right. **SAXI_23** was chosen because it is neither too close nor too distant from the PCI Express endpoint primitive (site **PCIEC4E4_X1Y0**).

The takeaway point here is that the **SAXI_*** interfaces of the Xilinx HBM IP are at fixed positions in the FPGA, and their positions must be considered if timing closure is to be achieved for a given FPGA design.

2.2 Design variant for ES1

The cross-stack AXI transaction hang issue, described in Section 7.6, means that the block diagram of this FPGA design when targeting an Virtex Ultrascale+ HBM ES1 FPGA is somewhat different, compared to when targeting production silicon.

As shown in Figure 4 below, the ES1 variant of the design uses two slave AXI3 interfaces in the HBM IP in order to avoid the cross-stack AXI transaction hang issue.

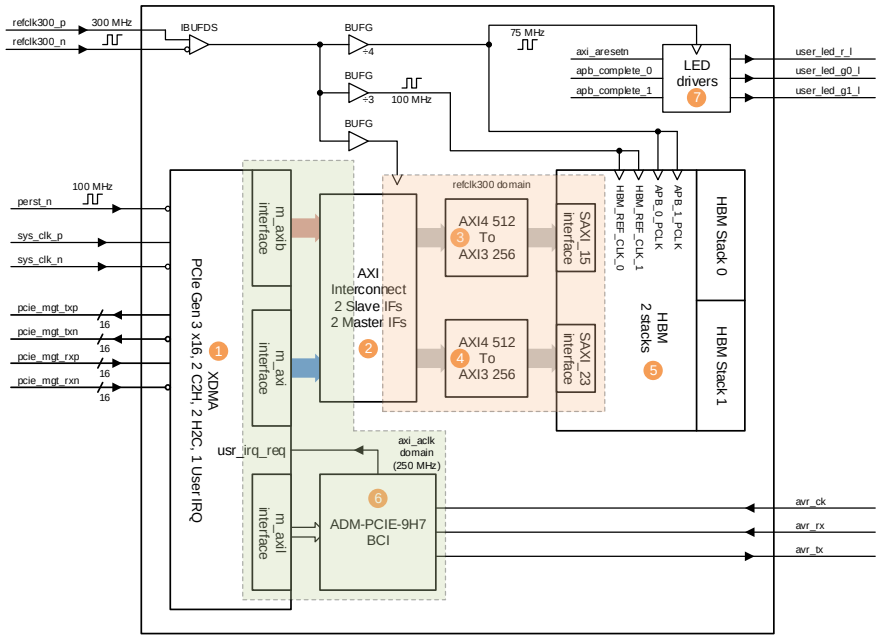

Figure 4 : Block diagram of the Host Interface to HBM FPGA Design (ES1)

The FPGA design is implemented as a Vivado Block Diagram, consisting of the following elements:

(1) An instance of the Xilinx **XDMA** IP, configured to include a PCI Express to AXI4 Bridge, two C2H DMA engines and two H2C DMA engines.

(2) An instance of the Xilinx **AXI Interconnect** IP, which allows both the DMA engines and the CPU (via the XDMA's PCI Express to AXI4 Bridge) to access the HBM.

In the ES1 variant of the design, the AXI Interconnect instance also peforms an address decoding function. As mentioned above, this example FPGA design is implemented as a Vivado Block Diagram. The Block Diagram's address map is used to ensure that, depending on whether an AXI transaction addresses HBM stack 0 (lower 4 GiB) or 1 (upper 4 GiB), the AXI transaction is routed to the **SAXI_15** or **SAXI_23** interface (respectively) of the HBM IP. This ensures that no cross-stack AXI transactions are delivered to the HBM IP, avoiding the issue described in Section 7.6.

(3) An instance of the Xilinx **AXI Width Converter** IP together with an instance of the Xilinx **AXI Protocol Converter** IP, which downsizes the 512-bit wide data from the XDMA instance to 256 bits and converts the protocol from AXI4 to AXI3, to match the requirements of the HBM IP's **SAXI_15** interface.

(4) Another instance of the Xilinx **AXI Width Converter** IP together with another instance of the Xilinx **AXI Protocol Converter** IP, which downsizes the 512-bit wide data from the XDMA instance to 256 bits and converts the protocol from AXI4 to AXI3, to match the requirements of the HBM IP's **SAXI_23** interface.

(5) An instance of the Xilinx **HBM** IP, configured to include both stacks of HBM that are present in the VU35P / VU37P FPGA, for a total of 8 GiB. This is a memory controller with an AXI3 interface.

The HBM IP instance is configured to enable two slave AXI3 interfaces: **SAXI_15** and **SAXI_23**. These interfaces were chosen because of their physical location in the FPGA and their associations with the two HBM stacks; **SAXI_15** is associated with HBM stack 0 whereas **SAXI_23** is associated with HBM stack 1. For further explanation of these choices, see Section 2.2.1.

(6) An instance of the Alpha Data **ADM-PCIE-9H7 BCI** IP (Board Control Interface), which allows the FPGA on an ADM-PCIE-9H7, when configured with this FPGA design's bitstream, to be compatible with Alpha Data's **avr2util** utility.

Note that the BCI instance is included only for compatibility with **avr2util**, and could be removed without affecting the functionality of the rest of the FPGA design.

(7) Three user-definable LEDs in the ADM-PCIE-9H7 are used to visually indicate (a) whether or not the **axi_aresetn** signal of the XDMA IP instance is asserted and (b) whether or not each HBM stack has completed its calibration. For details, refer to Appendix A.

2.2.1 HBM AXI slave interface locations (ES1)

In the block diagram depicted in Figure 4, the HBM IP instance is configured to enable its **SAXI_15** and **SAXI_23** interfaces, as opposed to the more obvious choices of **SAXI_00** and **SAXI_16**.

The reason for these choices is the physical locations of the **SAXI_*** interfaces within the FPGA; Figure 5 below shows the layout of the implemented FPGA design and the locations of some key primitives in the Super Logic Region 0 (SLR0):

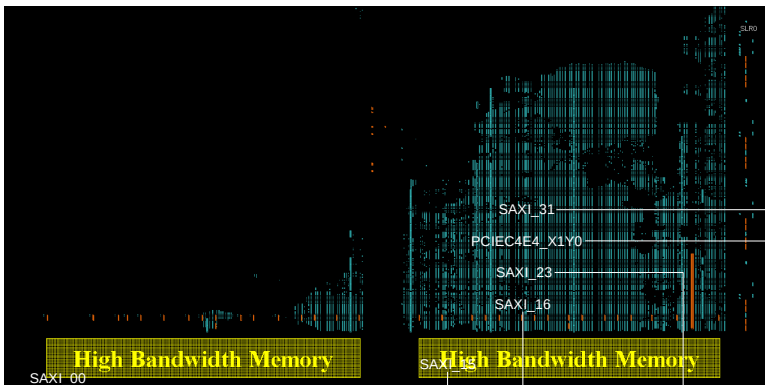

Figure 5 : Locations of key primitives in SLR0 (ES1)

The **SAXI_00** interface is at the extreme left, whereas **SAXI_31** is at the extreme right. **SAXI_15** was chosen because it is the nearest interface to the PCI Express endpoint primitive (site **PCIEC4E4_X1Y0**) that is associated with HBM stack 0. **SAXI_23** was chosen because it associated with HBM stack 1 and is neither too close nor too distant from the PCI Express endpoint primitive (site **PCIEC4E4_X1Y0**).

The takeaway point here is that the **SAXI_*** interfaces of the Xilinx HBM IP are at fixed positions in the FPGA, and their positions must be considered if timing closure is to be achieved for a given FPGA design.

2.3 Address Maps

Figure 6 below illustrates the addressing scheme as seen by the **m_axib** and **m_axi** interfaces of the XDMA IP instance.

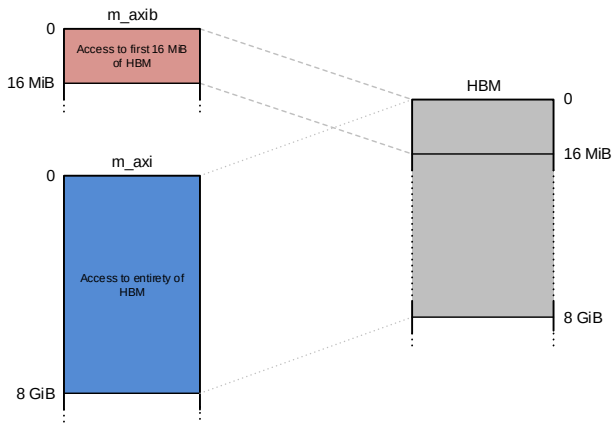

Figure 6 : Addressing scheme of the Host Interface to HBM FPGA Design

The address map of the Bypass interface (**m_axib** interface) consists of a single region which permits access by the host CPU to the first 16 MiB of the HBM:

Address range	Size	Purpose
0x000000 - 0xFFFFFF	16 MiB	Permits the host CPU to read and write the first 16 MiB of the HBM.
Others		Reserved; must not be accessed.

Table 1 : Bypass interface address map

Bypass Window size limitations
Although the Xilinx XDMA IP permits the Bypass Window size to be considerably larger than 16 MiB, in this example, 16 MiB has been chosen as a value which is unlikely to be problematic for the host machine's firmware (BIOS). Behavior is highly platform-specific, but some hardware platforms are unable to configure a PCI Express BAR whose size is larger a certain threshold.

In a more complex FPGA design, a page register could be implemented to enable all of the HBM in the FPGA to be accessible by the CPU, rather than the first 16 MiB. This is outside the scope of this example, however.

The C2H and H2C DMA engines (**m_axi** interface) each see an address map that permits access to the entirety of the HBM:

Address range	Size	Purpose
0x000000000 - 0x1FFFFFFFF	8 GiB	Permits any C2H DMA engine to read from the HBM. Permits any H2C DMA engine to write to the HBM.
Others		Reserved; must not be accessed.

Table 2 : DMA engine address map