## Revision History

The following table shows the revision history for this document.

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>02/27/2020</td>
<td>Version 1.3</td>
<td>Revised to indicate changes from ES to production release.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Removed sections.</td>
</tr>
<tr>
<td>11/20/2019</td>
<td>Version 1.2.1</td>
<td>Added information for generating the MCS file.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added reference clock information.</td>
</tr>
<tr>
<td>10/31/2019</td>
<td>Version 1.2</td>
<td>Updated to Vitis™ taxonomy.</td>
</tr>
<tr>
<td>06/28/2019</td>
<td>Version 1.1</td>
<td>New section.</td>
</tr>
<tr>
<td>02/11/2019</td>
<td>Version 1.0</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### 02/27/2020 Version 1.3

- **Block Diagram**
  - Revised to indicate changes from ES to production release.
- **Target Platform Floorplan, Memory Resource Availability per SLR, Basic Vitis Compiler Options, Kernel Insertion, -slr Example, and HMSS RAMA IP Insertion**
  - Removed sections.
- **Creating an MCS File and Programming the Alveo Card**
  - Added information for generating the MCS file.
- **DDR4 DIMM Memory**
  - Added reference clock information.
- **USB JTAG Interface**
  - Removed section.
- **FT4232HQ USB-JTAG/UART Interface**
  - Added connectivity information for the FTDI FT4232HQ device.
- **QSFP28 Module Connectors and Front Panel Indicators**
  - Added high-speed connection tables.
- **I2C Bus**
  - Provided additional information on the I2C bus.
- **Card Power System**
  - Added information on the card management system.
- **Clocks**
  - Added new section.
- **Miscellaneous I/O**
  - Added new section.

### 11/20/2019 Version 1.2.1

- General updates.
- Editorial updates only. No technical content updates.

### 10/31/2019 Version 1.2

- All sections.
- Updated to Vitis™ taxonomy.

### 06/28/2019 Version 1.1

- Chapter 2: Vitis Design Flow
- New section.

### 02/11/2019 Version 1.0

- Initial Xilinx release.
# Table of Contents

**Revision History** .......................................................................................................................... 2

**Chapter 1: Introduction** .............................................................................................................. 5  
  Block Diagram.............................................................................................................................. 7  
  Card Features............................................................................................................................... 7  
  Design Flows................................................................................................................................ 9  

**Chapter 2: Vitis Design Flow** .................................................................................................... 10  
  Target Platform Overview........................................................................................................ 10  
  High Bandwidth Memory Overview........................................................................................11  
  HBM Performance..................................................................................................................... 13  

**Chapter 3: Vivado Design Flow** .................................................................................................. 19  
  Downloading the Alveo U280 Card......................................................................................... 19  
  Creating an MCS File and Programming the Alveo Card..................................................... 22  
  Xilinx Design Constraints (XDC) File........................................................................................24  

**Chapter 4: Card Installation and Configuration** ........................................................................... 25  
  Electrostatic Discharge Caution............................................................................................... 25  
  Installing Alveo Data Center Accelerator Cards in Server Chassis......................................25  
  UltraScale+ Device Configuration............................................................................................ 26  

**Chapter 5: Card Component Description** .................................................................................. 27  
  UltraScale+ Device..................................................................................................................... 27  
  DDR4 DIMM Memory................................................................................................................ 27  
  Quad SPI Flash Memory........................................................................................................... 28  
  FT4232HQ USB-JTAG/UART Interface......................................................................................28  
  PCI Express Endpoint................................................................................................................29  
  QSFP28 Module Connectors and Front Panel Indicators..................................................... 29  
  I2C Bus........................................................................................................................................32  
  Card Power System..................................................................................................................... 32  
  Clocks.......................................................................................................................................... 32  
  Miscellaneous I/O..................................................................................................................... 35
Chapter 6: Known Issues and Limitations

Appendix A: Regulatory and Compliance Information
- CE Directives
- CE Standards
- Compliance Markings

Appendix B: Additional Resources and Legal Notices
- Xilinx Resources
- Documentation Navigator and Design Hubs
- References
- Please Read: Important Legal Notices
Chapter 1

Introduction

The Xilinx® Alveo™ U280 Data Center accelerator cards are Peripheral Component Interconnect express (PCIe®) Gen3 x16 compliant and Gen4 x8 compatible cards featuring the Xilinx 16 nm UltraScale+™ technology. The Alveo U280 card offers 8 GB of HBM2 at 460 GB/s bandwidth to provide high-performance, adaptable acceleration for memory-bound, compute-intensive applications including database, analytics, and machine learning inference.

The Alveo U280 Data Center accelerator cards are available in passive and active cooling configurations. Except where noted, this user guide applies to both the active and passive versions of the U280 card. The following figure shows a passively cooled Alveo U280 accelerator card.

Figure 1: Alveo U280 Data Center Accelerator Card (Passive Cooling)
CAUTION! The Alveo U280 accelerator card with passive cooling is designed to be installed into a data center server, where controlled air flow provides direct cooling. Due to the card enclosure, switches and LEDs are not accessible or visible. The card details in this user guide are provided to aid understanding of the card features. If the cooling enclosure is removed from the card and the card is powered-up, external fan cooling airflow MUST be applied to prevent over-temperature shut-down and possible damage to the card electronics. Removing the cooling enclosure voids the board warranty.

See Appendix B: Additional Resources and Legal Notices for references to documents, files, and resources relevant to the Alveo U280 accelerator cards.
**Block Diagram**

The block diagram of the Alveo U280 accelerator card is shown in the following figure.

*Figure 2: U280 Block Diagram*

**Card Features**

The Alveo U280 accelerator card features are listed in this section. Detailed information for each feature is provided in Chapter 5: Card Component Description.
• Alveo U280 accelerator card:
  ○ XCU280 UltraScale+ device
• Memory (two independent dual-rank DDR4 interfaces)
  ○ 32 gigabyte (GB) DDR4 memory
  ○ 2x DDR4 16 GB, 2400 mega-transfers per second (MT/s), 64-bit with error correcting code (ECC) DIMM
  ○ x4/x8 unregistered dual inline memory module (UDIMM) support
• Configuration options
  ○ 1 gigabit (Gb) Quad Serial Peripheral Interface (SPI) flash memory
  ○ Micro-AB universal serial bus (USB) JTAG configuration port
• 16-lane PCI Express
• Two QSFP28 connectors
• USB-to-UART FT4232HQ bridge with Micro-AB USB connector
• PCIe Integrated Endpoint block connectivity
  ○ Gen1, 2, or 3 up to x16
  ○ Gen4 x8
• I2C bus
• Status LEDs
• Power management with system management bus (SMBus) voltage, current, and temperature monitoring
• Dynamic power sourcing based on external power supplied
• 65W PCIe slot functional with PCIe slot power only
• 150 W PCIe slot functional with PCIe slot power and 6-pin PCIe AUX power cable connected
• 225 W PCIe slot functional with PCIe slot power and 8-pin PCIe AUX power cable connected
• Onboard reprogrammable flash configuration memory
• JTAG and universal asynchronous receiver-transmitter (UART) access through the USB port
• UltraScale+ device configurable over USB/JTAG and Quad SPI configuration flash memory
Design Flows

The preferred optimal design flow for targeting the Alveo Data Center accelerator card uses the Vitis™ unified software platform. However, traditional design flows, such as RTL or HLx are also supported using the Vivado® Design Suite tools. The following figure shows a summary of the design flows.

**Figure 3: Alveo Data Center Accelerator Card Design Flows**

<table>
<thead>
<tr>
<th>Traditional Flows</th>
<th>Vitis</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTL Flow</td>
<td>HLx Flow (IP integrator)</td>
</tr>
</tbody>
</table>

- High complexity
- Simplicity
- Complexity abstracted
- Slowest
- Time to Market
- Fastest
- High
- Hardware Expertise Required
- Low

Requirements for the different design flows are listed in the following table.

**Table 1: Requirements to Get Started with Alveo Data Center Accelerator Card Design Flows**

<table>
<thead>
<tr>
<th>Requirements to Get Started</th>
<th>RTL Flow</th>
<th>HLx Flow</th>
<th>Vitis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow documentation</td>
<td>UG949(^1)</td>
<td>UG895(^2)</td>
<td>UG1416(^3)</td>
</tr>
<tr>
<td>Hardware documentation</td>
<td>UG1314</td>
<td>UG1314</td>
<td>N/A</td>
</tr>
<tr>
<td>Vivado tools support</td>
<td>Board support XDC</td>
<td>Board support XDC</td>
<td>N/A</td>
</tr>
<tr>
<td>Programming the FPGA</td>
<td>Vivado Hardware Manager</td>
<td>Vivado Hardware Manager</td>
<td>UG1301(^4)</td>
</tr>
</tbody>
</table>

**Notes:**
3. *Vitis Accelerated Flow in the Vitis Unified Software Platform Documentation (UG1416).*
4. *Getting Started with Alveo Data Center Accelerator Cards (UG1301).*
Chapter 2

Vitis Design Flow

The following section is aimed at designers, and is intended as a quick start guide to the U280 target platform. It describes how to link a kernel into the target platform and then run this kernel in hardware on the platform. Background information on the structure of the platform is also provided.

For more details on the Vitis™ and hardware platforms, refer to the Vitis Accelerated Flow in the Vitis Unified Software Platform Documentation (UG1416) and Embedded Processor Platform Development in the Vitis Unified Software Platform Documentation (UG1416).

Target Platform Overview

The target platform consists of a Vivado® IP integrator block design with all of the required board interfaces configured and connected to the device I/Os.

The Vitis environment platforms require the device to remain up and running on a remote host while applications are downloaded to it. Vitis target platforms use partial reconfiguration (PR) technology to enable compiled binary downloads to the accelerator device while the device remains online and linked to the PCIe® bus of the host. There is a fixed static logic partition that contains the board interface logic. The static partition is not reimplemented when the Vitis applications are run. This is known as a target platform because it describes the boundary of the physical board with the kernel logic. The target platform is intended to keep the device alive and to control reconfiguration and debugging of the device.

The dynamically programmable region defines the programmable device logic partition that integrates the user kernels from the Vitis environment. It contains memories, memory interfaces, and AXI interconnect logic. This logic is dynamically configured and implemented along with the Vitis kernel logic each time an application is run. Each time a kernel application is run, partial reconfiguration technology is applied to reconfigure the dynamic region.

The dynamically programmable region has several DDR/HBM/PLRAM memory interfaces coupled with interconnect logic. The term PLRAM refers to internal UltraRAM/block RAM that can be accessed by host and user kernels. The dynamically programmable region uses the Memory Subsystem (MSS) IP (for DDR/PLRAM) and the HBM Memory Subsystem (HMSS) IP (for HBM). These subsystems are unique to the Vitis platforms. They contain multiple memory interfaces, coupled with the appropriate interconnect IP. When the dynamic region is being built.
(that is, when kernels are applied and connected to memory resources), the IP automatically trim away any of the unused DDR/PLRAM interfaces and disable the unused HBM interfaces which reside in the dynamic region. This allows for more of the device to be available for kernel logic, and reduces run time. Dynamic memory enablement is also optimal for power, because unused memories consume minimal power.

See Alveo Data Center Accelerator Card Platforms User Guide (UG1120) for information on available target platforms and resources.

High Bandwidth Memory Overview

U280 High Bandwidth Memory (HBM) devices incorporate 4 GB HBM stacks. Using stacked silicon interconnect technology, the programmable logic communicates to the HBM stacks through memory controllers. The U280 has access to two stacks of 4 GB HBM, each consisting of 16 pseudo channels with direct access to 256 MB. A high-level diagram of the two HBM stacks is shown below.

The programmable logic has 32 HBM AXI interfaces. HBM AXI interfaces can access any memory location in any of the 32 HBM PCs on either of the HBM stacks through a built-in switch providing access to the full 8 GB memory space. For more detailed information on the HBM, refer to AXI High Bandwidth Controller LogiCORE IP Product Guide (PG276). The flexible connection between the programmable logic and the HBM stacks results in easy floorplanning and timing closure as well as offering flexibility for kernel implementation.

In addition to HBM, two Memory IP instances enable access to on-board DDR4 SDRAM. The U280 accelerator card contains two channels of DDR4-2400 SDRAM, at 16 GB per channel for a total of 40 GB global memory between internal HBM and external DDR4.
The following figure shows a high-level view of the target platform and dynamic region in the U280 card.

**Figure 5: Static Region (Target Platform) and Dynamic Region of U280 Card**

---

**Vitis HBM Memory Subsystem**

The complexity of the HBM and required ancillary soft logic are abstracted by the HBM Memory Subsystem (HMSS) IP. When creating the target platform, the HMSS instantiates the HBM, enables all HBM PCs, and automatically connects the XDMA to the HBM for host access to global memory.

When the built target platform is used with the Vitis compiler, the HMSS is automatically re-customized by the flow to activate only the necessary memory controllers and ports as defined by kernel connectivity to HBM PCs, and to connect both the user kernels and the XDMA to those memory controllers for optimal bandwidth and latency.

It is important to specify the HBM PC mapping for each kernel in the system. Specifying this enables the optimal connection to be made from the kernel to the HBM within the HMSS.

The HMSS automatically handles the placement and timing complexities of AXI interfaces crossing super logic regions (SLRs) in SSI technology devices. You must specify the SLR location of the user kernels (this defaults to SLR0), which instructs the HMSS as to the location of each.
The HBM subsystem in U280 devices performs well in applications where sequential data access is required. However, for applications requiring random data access, performance can vary significantly depending on the application requirements (for example, the ratio of read and write operations, minimum transaction size, and size of the memory space being addressed). The Random Access Memory Attachment (RAMA) IP helps to address such problems by significantly improving memory access efficiency in cases where the required memory exceeds 256 MB (one HBM pseudo channel). Refer to RAMA LogiCORE IP Product Guide (PG310) for more information.

For using the RAMA IP to be considered, a kernel master should meet the following criteria:

- The master randomly accesses locations across more than one 256 MB HBM PC.
- The master uses a static, single ID on the AXI transaction ID ports (AxID), or the master uses slowly changing (pseudo-static) AXI transaction IDs.

If these conditions are not met, the thread creation used in the RAMA IP to improve performance has little effect, and consumes programmable logic for no purpose.

**Memory Subsystem**

When creating the target platform, the Memory Subsystem (MSS) instantiates all available DDR4 memory controllers and automatically constructs an optimized SmartConnect-based network to connect the XDMA to the memory controllers (two memory controllers in the case of the U280 card) for host access to global memory. PLRAM memory (internal SRAM – UltraRAM and block RAM) can also be accessed and is also created coupled to a similar optimal switch network.

When the built target platform is used with the Vitis compiler, the MSS is automatically re-customized by the flow to instantiate only the necessary memory as defined by kernel connectivity to memory resources, and to automatically construct an optimized SmartConnect-based network to connect both the user kernels and the XDMA to those memory resources. Unmapped memory resources are not instantiated, and the SmartConnect network is optimized accordingly.

The MSS automatically handles the placement and timing complexities of AXI interfaces crossing SLRs in SSI technology devices, in the event that user kernels in a given SLR are mapped to one or more memory controllers in a different SLR. Accordingly, you must specify the DDR/PLRAM resource requirements for each kernel in the system.

**HBM Performance**

The unique feature of the U280 platform is the hardened HBM subsystem, which is encapsulated by the Vitis HBM Memory Subsystem (HMSS). Important performance figures, recommendations, and limitations are covered in this section.
Performance Figures

A summary of the most common and least complex use case performances is shown below:

- Linear access point-to-point: single kernel master per HBM PC.
- Small random accesses point-to-point: single kernel master per HBM PC.

For linear accesses, the efficiency for all transaction sizes (read-only and write-only transaction types) is 91% when the transaction size is larger than 32 bytes.

Table 2: Linear Accesses (Single Kernel Master per HBM PC Performance)

<table>
<thead>
<tr>
<th>Transaction Size</th>
<th>Type</th>
<th>Bandwidth¹</th>
<th>Efficiency</th>
<th>Total Bandwidth²</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 B+</td>
<td>WO</td>
<td>13.1</td>
<td>91%</td>
<td>419</td>
</tr>
<tr>
<td>64 B+</td>
<td>RO</td>
<td>13.1</td>
<td>91%</td>
<td>419</td>
</tr>
<tr>
<td>32 B</td>
<td>WO</td>
<td>4.5</td>
<td>32%</td>
<td>144</td>
</tr>
<tr>
<td>32 B</td>
<td>RO</td>
<td>4.5</td>
<td>31%</td>
<td>144</td>
</tr>
</tbody>
</table>

Notes:
1. One master (GB/s).
2. 32 masters (GB/s).

The achievable bandwidth (read-only and write-only) for small random accesses is shown below.

Table 3: Small Random Accesses (Single Kernel Master per HBM PC performance)

<table>
<thead>
<tr>
<th>Transaction Size</th>
<th>Type</th>
<th>Bandwidth¹</th>
<th>Efficiency</th>
<th>Total Bandwidth²</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 B</td>
<td>WO</td>
<td>4.9</td>
<td>32%</td>
<td>147</td>
</tr>
<tr>
<td>64 B</td>
<td>RO</td>
<td>5.6</td>
<td>35%</td>
<td>160</td>
</tr>
<tr>
<td>32 B</td>
<td>WO</td>
<td>4.5</td>
<td>31%</td>
<td>144</td>
</tr>
<tr>
<td>32 B</td>
<td>RO</td>
<td>4.4</td>
<td>31%</td>
<td>141</td>
</tr>
</tbody>
</table>

Notes:
1. One master (GB/s).
2. 32 masters (GB/s).

Note: Because of the complexity and flexibility of the switch, there are many combinations that result in congestion at a particular memory location or in the switch itself. It is important to plan memory accesses so that kernels access limited memory where possible, and to isolate the memory accesses for different kernels into different HBM PCs.

Interleaved read and write transactions cause a drop in efficiency with respect to read-only or write-only due to memory controller timing parameters (bus turnaround). The exact performance drop is dictated by the read/write ratio and timing.
Performance Limitations

There is a bottleneck when a write transaction passes from HBM Stack 1 to HBM Stack 2. This bottleneck has the following effects:

- 33% of expected performance for 32B write transactions.
- 66% of expected performance for 64B write transactions.
- Other transaction sizes are not affected.
- Read transactions are not affected.

When planning memory resources, kernel masters planning to write at sizes of 32B/64B should not access HBM memory resources on both stacks.

Figure 6: Write Bottleneck for Small Transactions

Performance Notes

The following should be noted with respect to the platform:

- The host is connected to one of the HBM ports on the platform.
  - Up to 32 kernel masters can access HBM.
  - The HBM switch provides for 32 masters to be connected. The host therefore shares a switch port with one of the 32 kernel masters if required (this might not be optimal).
  - Xilinx recommends that no more than 31 kernel ports should be used if host and kernel interference is expected to cause a problem.
- A kernel requiring access to HBM and DDR4 must use separate masters.
This is a platform simplification.

- DDR4/PLRAM access can be shared on a single master.

- A kernel master should only access a contiguous subset of the 32 HBM pseudo channels.
  - This makes optimization of the HBM connection point more straightforward.

- By default, without user memory planning (`--connectivity.sp option on v++`), all kernel masters access a single HBM PC.

- By default, all kernels are assumed in SLR0.
  - This might lead to device congestion for complex designs, leading to a need for kernel floorplanning.

Recommendations

Recommendations to get best performance from the platform are as follows:

- Select the kernel master data width and kernel frequency to match the following requirements:
  - The hardened HBM core clock runs at 450 MHz.
  - The hardened HBM AXI ports are 256 bits wide.
  - For linear transactions, the kernel should be configured to match the following bandwidth:
    - 256 bits at 450 MHz (this might be a challenging design).
    - 512 bits at 225 MHz (a larger but less challenging design).
  - It might be possible to achieve the same bandwidth with lower kernel frequency/bus width combinations. This depends on the traffic profile/switch usage.
  - For random transactions, it might be possible to scale the kernel master bandwidth down; a kernel clock of 300 MHz and a width of 256 bits, for example. This must be assessed on a case-by-case basis.

- Use only HBM memory resources where possible.
  - DDR interfaces consume a large number of resources in the middle of SLR0 and SLR1, and might in some cases cause congestion when routing to the HBM.
  - PLRAM resources also consume resources, but are more flexible in placement and are therefore preferable to DDR if HBM is not sufficient.

- The platform offers a large amount of flexibility in terms of memory access patterns.
  - Measure memory subsystem performance to validate assumptions when not using simple access (single master to single memory resource).

- Use as many HBM PCs as possible in parallel.
High bandwidth can only be achieved with many kernel ports and many HBM PCs being used in parallel.

- A kernel master should be capable of at least 64 outstanding read transactions and 32 outstanding write transactions for random access.
  - More than this might add some performance at the expense of increased logic (up to 128 outstanding transactions supported).
- Smaller transactions offer less efficiency than larger transactions. Use large transactions where possible.
- Avoid routing small write transactions across the boundary between HBM stacks 1 and 2.
  - When using 32B or 64B write transactions, ensure that a kernel accesses HBM[0:15] (stack 1) or HBM[16:31] (stack 2); not both.

**PLRAM Use and Modification**

The U280 platform contains HBM DRAM and DDR DRAM memory resources. An additional memory resource in the platform is internal FPGA SRAM (UltraRAM and block RAM). The platform initially contains six instances of 128 KB of PLRAM (block RAM). There are two PLRAM instances in each SLR. The size and type (UltraRAM/block RAM) of each PLRAM can be changed before kernels are inserted. A pre-system-linkage Tcl file is used to change the PLRAM. Usage of the pre-system-linkage Tcl file can be enabled as follows on the v++ command line:

```
v++ --xp param:compiler.userPreSysLinkTcl=<full_path_to>/pre_user_tcl_file.tcl
```

An API is provided to change attributes of the PLRAM instance or memory resource:

```
sdx_memory_subsystem::update_plram_specification <memory_subsystem_bdcell> <plram_resource_name> <plram_specification>
```

In this API, `<plram_specification>` is a Tcl dictionary consisting of the following entries (entries below are the defaulted values for each instance in the platform):

```
{
  SIZE 128K # Up to 4M
  AXI_DATA_WIDTH 512 # Up to 512
  SLR_ASSIGNMENT SLR0 # SLR0 / SLR1 / SLR2
  MEMORY_PRIMITIVE BRAM # BRAM or URAM
  READ_LATENCY 1 # To optimise timing path
}
```
In the example below, \texttt{PLRAM\_MEM00} is changed to be 2 MB in size and composed of UltraRAM; \texttt{PLRAM\_MEM01} is changed to be 4 MB in size and composed of UltraRAM. These resources correspond to the \texttt{v++} command line memory resources \texttt{PLRAM[0:1]}.

\begin{verbatim}
# Setup PLRAM
sdx_memory_subsystem::update_plram_specification
[get_bd_cells /memory_subsystem] PLRAM\_MEM00 { SIZE 2M AXI\_DATA\_WIDTH 512
SLR\_ASSIGNMENT SLR0 READ\_LATENCY 10 MEMORY\_PRIMITIVE URAM}

sdx_memory_subsystem::update_plram_specification
[get_bd_cells /memory_subsystem] PLRAM\_MEM01 { SIZE 4M AXI\_DATA\_WIDTH 512
SLR\_ASSIGNMENT SLR0 READ\_LATENCY 10 MEMORY\_PRIMITIVE URAM}

validate\_bd\_design -force
save\_bd\_design
\end{verbatim}

The \texttt{READ\_LATENCY} is an important attribute, because it sets the number of pipeline stages between memories cascaded in depth. This varies by design, and affects the timing QoR of the platform and the eventual kernel clock rate. In the example above for \texttt{PLRAM\_MEM01 (PLRAM[1])}:

- 4 MB of memory are required in total.
- Each UltraRAM is 32 KB (64 bits wide). 4 MB \times 32 KB \rightarrow 128 UltraRAMs in total.
- Each PLRAM instance is 512 bits wide \rightarrow 8 UltraRAMs are required in width.
- 128 total UltraRAMs with 8 UltraRAMs in width \rightarrow 16 UltraRAMs in depth.
- A good rule of thumb is to pick a read latency of depth/2 + 2 \rightarrow in this case, \texttt{READ\_LATENCY} = 10.

This allows a pipeline on every second UltraRAM, resulting in the following:

- Good timing performance between UltraRAMs.
- Placement flexibility; not all UltraRAMs need to be placed in the same UltraRAM column for cascade.
Chapter 3

Vivado Design Flow

This section provides a starting point for expert HDL developers using the RTL flows, or developers who want to customize in HLx beyond the standard support in the Vivado® tools.

Downloading the Alveo U280 Card

For either the RTL or HLx flow, designers can start by downloading the U280 accelerator card files (zip).

1. In Vivado, select Create New Project→RTL Project to open the New Project dialog box.
2. Click Update Board Repositories for the latest board files, and then select the U280 card:
The New Project Summary dialog box appears.
3. **Click Finish.**

The Project Summary Overview page appears. You can now create the RTL-based project.
Creating an MCS File and Programming the Alveo Card

For custom RTL flow, this section outlines the procedures to do the following:

- Create an MCS file (PROM image)
- Flash programming through the USB-JTAG (Micro-USB) interface

Create an MCS File (PROM Image)

The Alveo accelerator card contains a Quad SPI configuration flash memory part that can be configured over USB-JTAG. This part contains a protected region, with the factory base image at the 0x00000000 address space. This base image points to the customer programmable region at an 0x01002000 address space offset.

To ensure that the PROM image is successfully loaded onto the Alveo accelerator card at power on, the starting address must be set to 0x01002000 and the interface set to spix4 when creating the MCS file. Details on adding this to the MCS file can be found in the UltraScale Architecture Configuration User Guide (UG570).
In addition, the following code must be placed in the project XDC file to correctly configure the MCS file.

```bash
# Bitstream Configuration
# ------------------------------------------------------------------------
set_property CONFIG_VOLTAGE 1.8 [current_design]
set_property BITSTREAM.CONFIG.CONFIGFALLBACK Enable [current_design]
set_property BITSTREAM.GENERAL.COMPRESS TRUE [current_design]
set_property CONFIG_MODE SPIx4 [current_design]
set_property BITSTREAM.CONFIG.SPI_BUSWIDTH 4 [current_design]
set_property BITSTREAM.CONFIG.CONFIGRATE 85.0 [current_design]
set_property BITSTREAM.CONFIG.EXTMASTERCCLK_EN disable [current_design]
set_property BITSTREAM.CONFIG.SPI_FALL_EDGE YES [current_design]
set_property BITSTREAM.CONFIG.UNUSEDPIN Pullup [current_design]
set_property BITSTREAM.CONFIG.SPI_32BIT_ADDR Yes [current_design]
# ------------------------------------------------------------------------
```

After the XDC file has been updated, generate the MCS file with the following command (including the quotation marks.)

```bash
write_cfgmem -force -format mcs -interface spix4 -size 1024 -loadbit "up 0x01002000 <input_file.bit>" -file "<output_file.mcs>"
```

Where:
- `<input_file.bit>` is the filename of the input .bit file
- `<output_file.mcs>` is the MCS output filename

### Program the Alveo Card

After the MCS file is created, see the procedure in the "Programming the FPGA Device" chapter in the *Vivado Design Suite User Guide: Programming and Debugging (UG908)* to connect to the Alveo Data Center accelerator card using the hardware manager.

1. Select **Add Configuration Device** and select the mt25qu01g-spi-x1_x2_x4 part.
2. Right-click the target to select **Program the Configuration Memory Device**.
   a. Select the MCS file target.
   b. Select **Configuration File Only**.
   c. Click **OK**.
3. After programming has completed, disconnect the card in the hardware manager, and disconnect the USB cable from the Alveo accelerator card.
4. Perform a cold reboot on the host machine to complete the card update.

**IMPORTANT!** If you are switching between an Alveo Data Center accelerator card target platform and a custom design, revert the card to the golden image before loading an alternate image into the PROM. See *Getting Started with Alveo Data Center Accelerator Cards (UG1301)* for more information.
Xilinx Design Constraints (XDC) File

RTL users can reference the *Vivado Design Suite User Guide: Using Constraints* (UG903) for more information. The Alveo accelerator card XDC files are available for download from their respective websites along with this user guide.
Chapter 4

Card Installation and Configuration

Electrostatic Discharge Caution

**CAUTION!** ESD can damage electronic components when they are improperly handled, and can result in total or intermittent failures. Always follow ESD-prevention procedures when removing and replacing components.

To prevent ESD damage:

- Use an ESD wrist or ankle strap and ensure that it makes skin contact. Connect the equipment end of the strap to an unpainted metal surface on the chassis.
- Avoid touching the adapter against your clothing. The wrist strap protects components from ESD on the body only.
- Handle the adapter by its bracket or edges only. Avoid touching the printed circuit board or the connectors.
- Put the adapter down only on an antistatic surface such as the bag supplied in your kit.
- If you are returning the adapter to Xilinx Product Support, place it back in its antistatic bag immediately.

Installing Alveo Data Center Accelerator Cards in Server Chassis

Because each server or PC vendor's hardware is different, for physical board installation guidance, see the manufacturer's PCI Express® board installation instructions.

For programming and start-up details, see *Getting Started with Alveo Data Center Accelerator Cards (UG1301)*.
UltraScale+ Device Configuration

The Alveo U280 accelerator card supports two UltraScale+™ FPGA configuration modes:

- Quad SPI flash memory
- JTAG using USB JTAG configuration port

The FPGA bank 0 mode pins are hardwired to M[2:0] = 001 master SPI mode with pull-up/down resistors.

At power up, the FPGA is configured by the Quad SPI NOR flash device (Micron MT25QU01GBBA8E12-0SIT) with the FPGA_CCLK operating at a clock rate up to 105 MHz using the master serial configuration mode. The Quad SPI flash memory NOR device has a capacity of 1 Gb.

If the JTAG cable is plugged in, QSPI configuration might not occur. JTAG mode is always available independent of the mode pin settings.

For complete details on configuring the FPGA, see the *UltraScale Architecture Configuration User Guide (UG570).*

**Table 4: Configuration Modes**

<table>
<thead>
<tr>
<th>Configuration Mode</th>
<th>M[2:0]</th>
<th>Bus Width</th>
<th>CCLK Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master SPI</td>
<td>001</td>
<td>x1, x2, x4</td>
<td>FPGA output</td>
</tr>
<tr>
<td>JTAG</td>
<td>Not applicable – JTAG overrides</td>
<td>x1</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
Card Component Description

This chapter provides a functional description of the components of the Alveo™ U280 Data Center accelerator card.

UltraScale+ Device

The Alveo U280 accelerator card is populated with the 16 nm UltraScale+™ XCU280 FPGA. This device incorporates two 4 GB High Bandwidth Memory (HBM) stacks adjacent to the device die. Using SSI technology, the device communicates to the HBM stacks through memory controllers that connect through the silicon interposer at the bottom of the device. Each XCU280 FPGA contains two 4 GB HBM stacks, resulting in up to 8 GB of HBM per device. The device includes 32 HBM AXI interfaces used to communicate with the HBM. The flexible addressing feature that is provided by a built-in switch allows for any of the 32 HBM AXI interfaces to access any memory address on either one or both of the HBM stacks. This flexible connection between the device and the HBM stacks is helpful for floorplanning and timing closure.

Related Information
Known Issues and Limitations

DDR4 DIMM Memory

Two independent dual-rank DDR4 interfaces are available. The card is populated with two socketed single-rank Micron MTA18ASF2G72PZ-2G3B1IG 16 GB DDR4 RDIMMs. Each DDR4 DIMM is 72 bits wide (64 bits plus support for ECC). The DDR4 and HBM interfaces have a 100.000 MHz system reference clock input. The 100.000 MHz reference clock to bank 65 of SLR0 is used by both the HBM and the DDR4 C0 interface in banks 64, 65, and 66. The C1 DDR4 interface is located in banks 68, 69, and 70 with the 100.000 MHz system clock going into bank 69 of SLR1. Banks 64, 65, 66, 68, 69, and 70 are powered from an onboard 1.2V regulator.

The detailed FPGA and DIMM pin connections for the feature described in this section are documented in the Alveo U280 accelerator card XDC file.

**Related Information**
Xilinx Design Constraints (XDC) File

---

### Quad SPI Flash Memory

The Quad SPI device provides 1 Gb of nonvolatile storage.

- Part number: MT25QU01GBBB8E12-0AAT (Micron)
- Supply voltage: 1.8V
- Datapath width: 4 bits
- Data rate: 100 MHz

For more flash memory details, see the Micron MT25QU01GBBB8E12-0AAT data sheet at the Micron website.

For configuration details, see the *UltraScale Architecture Configuration User Guide* (UG570). The detailed FPGA and Flash pin connections for the feature described in this section are documented in the Alveo U280 accelerator card XDC file, referenced in *Xilinx Design Constraints (XDC) File*.

**Related Information**
Xilinx Design Constraints (XDC) File

---

### FT4232HQ USB-JTAG/UART Interface

The USB connector for debug and development is located on the rear of the card and requires a micro-AB USB connection cable. The USB is connected to a FTDI FT4232HQ device, which has the following connectivity:

- **ADBUSH → JTAG** connection to the UltraScale+ device JTAG through a level shifter that can be disabled by the satellite controller to prevent user tampering.
  
  **Note:** This port is also connected to the satellite controller through a level shifter that can be disabled by the satellite controller to prevent user tampering.

- **ACBUS → RS232** serial port connection to the satellite controller.
  
  - Port connection needs to be configured for no parity, 8 data bits, 1 stop bit with a line rate of 115200.
• ABBUS → RS232 serial port connection to the UltraScale+ device.
  ○ Port connection needs to be configured by user and UltraScale+ device design.
  ○ USB_UART_TX connects to IO_L24N_T3U_N11_75 (pin A28) on the UltraScale+ device.
  ○ USB_UART_RX connects to IO_T3U_N12_75 (pin B33) on the UltraScale+ device.

The FTDI FT4232HQ data sheet is available on the FTDI website: https://www.ftdichip.com/.

---

**PCI Express Endpoint**

The Alveo U280 accelerator card implements a 16-lane PCI Express® edge connector that performs data transfers at the rate of 2.5 giga-transfers per second (GT/s) for Gen1, 5.0 GT/s for Gen2, 8.0 GT/s for Gen3 applications, and 16.0 GT/s for Gen4 applications.

The detailed UltraScale+ device and PCIe pin connections for the feature described in this section are documented in the accelerator card design constraints (XDC) file. The endpoint is compliant to the v3.0 specification, and compatible with the specification (specifications are available here). For additional information on Gen4 compatible features, see *UltraScale+ Devices Integrated Block for PCI Express LogiCORE IP Product Guide* (PG213).

**Related Information**

Xilinx Design Constraints (XDC) File

---

**QSFP28 Module Connectors and Front Panel Indicators**

The Alveo accelerator cards host two 4-lane small form-factor pluggable (QSFP) connectors that accept an array of optical modules and copper cables. Each connector is housed within a single QSFP cage assembly.

The QSFP+ connectors sideband control signals (MODSELL, RESETL, MODPRSL, INTL, and LPMODE) are behind an I2C expander and are not accessible from the UltraScale+ device. The QSFP+ I2C interface and sideband control signals on the Alveo U280 accelerator card are only accessible through the satellite controller, and are currently not supported with the exception that the satellite controller does actively drive the LPMODE signal low so that an optical module is enabled when inserted. The MODSELL, RESETL, MODPRSL, INTL, and LPMODE sideband signals are defined in the small form factor (SFF) specifications listed below.

The components visible through the card PCIe panel/bracket top to bottom are:
Done, power bad, and 2x status LEDs
  - Currently, the two status LEDs (green and yellow) are undefined and have no current functionality.
  - The done LED is blue and indicates that the UltraScale+ device is configured.
  - The power bad LED is red and indicates there is not a power good situation. This signal is derived from the AND'ing of all the power supplies power good signals on the card and inverting the result.

QSFP0: the QSFP0 interface is the upper QSFP connector and has the high-speed connections to the UltraScale+ device listed in the following table.

Table 5: QSFP0 Interface High-Speed Connections

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Bank</th>
<th>Pin Reference</th>
<th>Pin Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>MGT_SI570_CLOCK0_C_N</td>
<td>134</td>
<td>MGTREFCLK0N_134</td>
<td>T43</td>
</tr>
<tr>
<td>MGT_SI570_CLOCK0_C_P</td>
<td>134</td>
<td>MGTREFCLK0P_134</td>
<td>T42</td>
</tr>
<tr>
<td>QSFP0_CLOCK_N</td>
<td>134</td>
<td>MGTREFCLK1N_134</td>
<td>R41</td>
</tr>
<tr>
<td>QSFP0_CLOCK_P</td>
<td>134</td>
<td>MGTREFCLK1P_134</td>
<td>R40</td>
</tr>
<tr>
<td>QSFP0_RX1_N</td>
<td>134</td>
<td>MGTYRXN0_134</td>
<td>L54</td>
</tr>
<tr>
<td>QSFP0_RX2_N</td>
<td>134</td>
<td>MGTYRXN1_134</td>
<td>K52</td>
</tr>
<tr>
<td>QSFP0_RX3_N</td>
<td>134</td>
<td>MGTYRXN2_134</td>
<td>J54</td>
</tr>
<tr>
<td>QSFP0_RX4_N</td>
<td>134</td>
<td>MGTYRXN3_134</td>
<td>H52</td>
</tr>
<tr>
<td>QSFP0_RX1_P</td>
<td>134</td>
<td>MGTYRXP0_134</td>
<td>L53</td>
</tr>
<tr>
<td>QSFP0_RX2_P</td>
<td>134</td>
<td>MGTYRXP1_134</td>
<td>K51</td>
</tr>
<tr>
<td>QSFP0_RX3_P</td>
<td>134</td>
<td>MGTYRXP2_134</td>
<td>J53</td>
</tr>
<tr>
<td>QSFP0_RX4_P</td>
<td>134</td>
<td>MGTYRXP3_134</td>
<td>H51</td>
</tr>
<tr>
<td>QSFP0_TX1_N</td>
<td>134</td>
<td>MGTYTXN0_134</td>
<td>L49</td>
</tr>
<tr>
<td>QSFP0_TX2_N</td>
<td>134</td>
<td>MGTYTXN1_134</td>
<td>L45</td>
</tr>
<tr>
<td>QSFP0_TX3_N</td>
<td>134</td>
<td>MGTYTXN2_134</td>
<td>K47</td>
</tr>
<tr>
<td>QSFP0_TX4_N</td>
<td>134</td>
<td>MGTYTXN3_134</td>
<td>J49</td>
</tr>
<tr>
<td>QSFP0_TX1_P</td>
<td>134</td>
<td>MGTYTXP0_134</td>
<td>L48</td>
</tr>
<tr>
<td>QSFP0_TX2_P</td>
<td>134</td>
<td>MGTYTXP1_134</td>
<td>L44</td>
</tr>
<tr>
<td>QSFP0_TX3_P</td>
<td>134</td>
<td>MGTYTXP2_134</td>
<td>K46</td>
</tr>
<tr>
<td>QSFP0_TX4_P</td>
<td>134</td>
<td>MGTYTXP3_134</td>
<td>J48</td>
</tr>
</tbody>
</table>

QSFP1: the QSFP1 interface is the lower QSFP connector and has the high-speed connections to the UltraScale+ device listed in the following table.

Table 6: QSFP1 Interface High-Speed Connections

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Bank</th>
<th>Pin Reference</th>
<th>Pin Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>MGT_SI570_CLOCK1_C_N</td>
<td>135</td>
<td>MGTREFCLK0N_135</td>
<td>P43</td>
</tr>
<tr>
<td>MGT_SI570_CLOCK1_C_P</td>
<td>135</td>
<td>MGTREFCLK0P_135</td>
<td>P42</td>
</tr>
</tbody>
</table>
Table 6: QSFP1 Interface High-Speed Connections (cont’d)

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Bank</th>
<th>Pin Reference</th>
<th>Pin Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSFP1_CLOCK_N</td>
<td>135</td>
<td>MGTREFCLK1N_135</td>
<td>M43</td>
</tr>
<tr>
<td>QSFP1_CLOCK_P</td>
<td>135</td>
<td>MGTREFCLK1P_135</td>
<td>M42</td>
</tr>
<tr>
<td>QSFP1_RX1_N</td>
<td>135</td>
<td>MGTYRXN0_135</td>
<td>G54</td>
</tr>
<tr>
<td>QSFP1_RX2_N</td>
<td>135</td>
<td>MGTYRXN1_135</td>
<td>F52</td>
</tr>
<tr>
<td>QSFP1_RX3_N</td>
<td>135</td>
<td>MGTYRXN2_135</td>
<td>E54</td>
</tr>
<tr>
<td>QSFP1_RX4_N</td>
<td>135</td>
<td>MGTYRXN3_135</td>
<td>D52</td>
</tr>
<tr>
<td>QSFP1_RX1_P</td>
<td>135</td>
<td>MGTYRXP0_135</td>
<td>G53</td>
</tr>
<tr>
<td>QSFP1_RX2_P</td>
<td>135</td>
<td>MGTYRXP1_135</td>
<td>F51</td>
</tr>
<tr>
<td>QSFP1_RX3_P</td>
<td>135</td>
<td>MGTYRXP2_135</td>
<td>E53</td>
</tr>
<tr>
<td>QSFP1_RX4_P</td>
<td>135</td>
<td>MGTYRXP3_135</td>
<td>D51</td>
</tr>
<tr>
<td>QSFP1_TX1_N</td>
<td>135</td>
<td>MGTYTXN0_135</td>
<td>G49</td>
</tr>
<tr>
<td>QSFP1_TX2_N</td>
<td>135</td>
<td>MGTYTXN1_135</td>
<td>E49</td>
</tr>
<tr>
<td>QSFP1_TX3_N</td>
<td>135</td>
<td>MGTYTXN2_135</td>
<td>C49</td>
</tr>
<tr>
<td>QSFP1_TX4_N</td>
<td>135</td>
<td>MGTYTXN3_135</td>
<td>A50</td>
</tr>
<tr>
<td>QSFP1_TX1_P</td>
<td>135</td>
<td>MGTYTXP0_135</td>
<td>G48</td>
</tr>
<tr>
<td>QSFP1_TX2_P</td>
<td>135</td>
<td>MGTYTXP1_135</td>
<td>E48</td>
</tr>
<tr>
<td>QSFP1_TX3_P</td>
<td>135</td>
<td>MGTYTXP2_135</td>
<td>C48</td>
</tr>
<tr>
<td>QSFP1_TX4_P</td>
<td>135</td>
<td>MGTYTXP3_135</td>
<td>A49</td>
</tr>
</tbody>
</table>

For additional information about the quad SFF pluggable (28 Gb/s QSFP+) module, see the SFF-8663 and SFF-8679 specifications for the 28 Gb/s QSFP+ at the SNIA Technology Affiliates website: https://www.snia.org/sff/specifications2.

Each QSFP connector has its own clock generator with two selectable frequencies of 161.132812 MHz or 156.2500 MHz. See Clocks for details on the QSFP+ clocking.

The detailed UltraScale+ device and QSFP pin connections for the feature described in this section are documented in the Xilinx Design Constraints (XDC) File.

Related Information

Xilinx Design Constraints (XDC) File
I2C Bus

The Alveo U280 accelerator cards implement an I2C bus network to communicate with numerous sensors and voltage regulators on the card. These I2C interfaces are not accessible from the UltraScale+ device and are only accessible via the satellite controller. For more information on accessing various sensors on the card via the UltraScale+ device, see Card Management Solution Subsystem Product Guide (PG348) for card management system access and implementation (the device tree details are available in the board support package).

Card Power System

Limited power system telemetry is available through the card management system (CMS) I2C IP. See Card Management Solution Subsystem Product Guide (PG348) for information on how the I2C IP is instantiated during the UltraScale+ device design process that begins after the Alveo Data Center accelerator card is selected from the Vivado Design Suite Boards tab. See Design Flows for more information.

Clocks

The Alveo U280 card has reference and system clocks located in multiple banks to support the many features on the card.

- System clocks: the UltraScale+ device system reference clocks are supplied by a local 100.000 MHz Abracon AB-557-03-HHC-F-L-C-T oscillator and also connected to a Silicon Labs SI53340-B-GM 1:4 clock buffer. The local oscillator clock is distributed to bank 65 (DDR4), bank 69 (HBM and DDR4), and bank 75 (common logic).
  - SI53340-B-GM → OUT0 SYSCLK0_P/SYSCLK0_N 100.000 MHz. System clock for first DDR4 MIG interface and HBM memories.
    - Pins: IO_L12P_T1U_N10_GC_A08_D24_65/IO_L12N_T1U_N11_GC_A09_D25_65.
  - SI53340-B-GM → OUT1 SYSCLK1_P/SYSCLK1_N 100.000 MHz. System clock for second DDR4 MIG interface.
    - Pins: IO_L13P_T2L_N0_GC_QBC_69/IO_L13N_T2L_N1_GC_QBC_69.
  - SI53340-B-GM → OUT3 SYSCLK3_P/SYSCLK3_N 100.000 MHz - bank 75 100 MHz input clock.
    - Pins: IO_L11P_T1U_N8_GC_75/IO_L11N_T1U_N9_GC_75.
The U280 card has several reference input clocks to support PCIe x16 and bifurcated PCIe x8.

- The PCIe edge connector to the 100 MHz clock source is buffered through a Silicon Labs SI53322-B-GM 1:2 clock buffer and supplied to bank 225 and 227 for the PCIe GTY transceivers.
  - PCIe fingers PEX_REFCLK_P/PEX_REFCLK_P 100.000 MHz is buffered through a Silicon Labs SI53322-B-GM 1:2 clock buffer and distributed to these pins:
    - OUT0 → PCIE_CLK0_P/PCIE_CLK0_N 100.000 MHz - PCIe REFCLK0 for bifurcated x8 lanes 0-7 synchronous clocking.
    - Pins: MGTREFCLK0P_227_AL15/MGTREFCLK0N_227_AL14.
    - OUT1 → PCIE_CLK1_P/PCIE_CLK1_N 100.000 MHz - PCIe REFCLK0 for x16 and bifurcated x8 lanes 8-15 synchronous clocking.
    - Pins: MGTREFCLK0P_225_AR15/MGTREFCLK0N_225_AR14.
- An onboard Abracon AB-557-03-HCHC-F-L-C-T @ 100.000 MHz dual output PCIe MEMs oscillator is also used with a Silicon Labs SI53340-B-GM 1-to-4 clock buffer to supply clocking to the PCIe interfaces on bank 224 through bank 227.
  - SI53340-B-GM → OUT2 SYSCLK2_P/SYSCLK2_N 100.000 MHz - PCIe REFCLK1 for bifurcated x8 lanes 0-7 asynchronous clocking.
    - Pins: MGTREFCLK1P_227_AK13/MGTREFCLK1N_227_AK12.
  - AB-557-03 OUT0 → SYS_CLK5_P/SYS_CLK5_N @ 100.000 MHz - PCIe REFCLK1 for x16 and bifurcated x8 lanes 8-15 asynchronous clocking.
    - Pins: MGTREFCLK1P_225_AP13/MGTREFCLK1N_225_AP12.

Each QSFP quad has both reference clocks connected as follows:

- QSFP0 clock MGTREFCLK1:
  - Clock generator: Silicon Labs 546BAB001028BBG.
  - Output CLK_P/N: the QSFP0_CLOCK_P/N clock is an AC-coupled LVDS 156.25 MHz clock wired to the GTY bank 134 MGTREFCLK1clock input pins (MGTREFCLK1P_134_R40/MGTREFCLK1N_134_R41).
  - Clock controls:
    - OE_B ← Active-Low input to SI546 to enable output frequency.
      Pin: "QSFP0_OEB" - IO_L9P_T1L_N4_AD12P_75_H32.
    - FS ← Clock select pin.
      FS = 1 → 161.132812 MHz 1.8V LVDS (default when FPGA pin Hi-Z or driven High).
      FS = 0 → 156.25 MHz 1.8V LVDS when driven low by FPGA.
      Pin: "QSFP0_FS" - IO_L9N_T1L_N5_AD12N_75_G32.
• QSFP1 clock MGTREFCLK1.
  ○ Clock generator: Silicon Labs 546BAB001028BBG.
  ○ Output CLK_P/N: the QSFP0_CLOCK_P/N clock is an AC-coupled LVDS 156.25 MHz clock wired to the GTY bank 135 MGTREFCLK1 clock input pins (MGTREFCLK1P_135_M42/MGTREFCLK1N_135_M43).
  ○ Clock controls:
    - OE_B ← Active-Low input to SI546 to enable output frequency.
      Pin: "QSFP1_OEB" - IO_L8N_T1L_N3_AD5N_75_H30.
    - FS ← Clock select pin.
      FS = 1 → 161.132812 MHz 1.8V LVDS (default when FPGA pin Hi-Z or driven High).
      FS = 0 → 156.25 MHz 1.8V LVDS when driven low by FPGA.
      Pin: "QSFP1_FS" - IO_L7N_T1L_N1_QBC_AD13N_75_G33.

• QSFP0/1 clock MGTREFCLK0.
  ○ Clock generator: SI570 - SiLabs 570BAB000544DG at 156.250 MHz programmable oscillator
    Reprogramming I2C access only possible via the satellite controller through an I2C multiplexer. There is no access to the SI570 I2C interface from the FPGA. The output of the SI570 device is fed into a Silicon Labs SI53340-B-GM 1:4 LVDS clock buffer and distributed to the FPGA MGTREFCLK0 of banks 134 and 135. In addition, a third clock is distributed to bank 75.
  ○ SI53340-B-GM outputs:
    - OUT0: USER_SI570_CLOCK_P/USER_SI570_CLOCK_N 156.250 MHz - general purpose system clock.
      Pins: IO_L12P_T1U_N10_GC_75_G30/IO_L12N_T1U_N11_GC_75_F30.
    - OUT1: Not connected.
      Pins: N/A
    - OUT2 MGT_SI570_CLOCK0_C_P/MGT_SI570_CLOCK0_C_N 156.250 MHz - QSFP0 REFCLK0.
      Pins: MGTREFCLK0P_134_T42/MGTREFCLK0N_134_T43.
    - OUT3 MGT_SI570_CLOCK1_C_P/MGT_SI570_CLOCK1_C_N 156.250 MHz - QSFP0 REFCLK1.
      Pins: MGTREFCLK0P_135_P42/MGTREFCLK0N_135_P43.
In summary, the UltraScale+ device reference clocks are supplied by a local 100.000 MHz Abracon AB-557-03-HCHC-F-L-C-T oscillator also connected to a Silicon Labs SI53340-B-GM 1:4 clock buffer and the PCIe edge connector 100 MHz clock source. The local oscillator clock is distributed to bank 65, bank 69, bank 75, bank 225, and bank 227 for the DDR4, common logic and PCIe GTY transceivers. The PCIe edge connector 100 MHz clock source is buffered through a Silicon Labs SI53322-B-GM 1:2 clock buffer and supplied to bank 225 and bank 227 for the PCIe GTY transceivers. The detailed FPGA and clock pin connections for the feature described in this section are documented in the U280 accelerator card XDC file.

Related Information
Xilinx Design Constraints (XDC) File

## Miscellaneous I/O

On the U280 card, the UltraScale+ device has a single bank with various inputs and outputs to communicate with the satellite controller and other devices on the card. See the XDC file for all of the signal connections. This section only references the signals that are usable in bank 75. Bank 75 is powered from a 1.8V source.

**IMPORTANT!** The D32 pin must be connected appropriately or tied to logic 0. If D32 is pulled up, floated, or tied to 1, the U280 card might become unrecoverable after programming.

### Table 7: Bank 75 SLR2 Miscellaneous I/O Descriptions

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Direction</th>
<th>Pin</th>
<th>I/O</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPGA_RXD_MSP</td>
<td>Input</td>
<td>E28</td>
<td>IO_L18P_T2U_N10_AD2P_75</td>
<td>Satellite controller CMS UART receive (115200, no parity, 8 bits, 1 stop bit).</td>
</tr>
<tr>
<td>FPGA_TXD_MSP</td>
<td>Output</td>
<td>D29</td>
<td>IO_L18N_T2U_N11_AD2N_75</td>
<td>Satellite controller CMS UART transmit (115200, no parity, 8 bits, 1 stop bit).</td>
</tr>
<tr>
<td>I2C_FPGA_SDA</td>
<td>Bi Dir</td>
<td>C33</td>
<td>IO_T2U_N12_75</td>
<td>Slave I2C data connection from satellite controller to FPGA.</td>
</tr>
<tr>
<td>I2C_FPGA_SCL</td>
<td>Bi Dir</td>
<td>C30</td>
<td>IO_L19P_T3L_N0_DBC_AD9P_75</td>
<td>Slave I2C clock connection from satellite controller to FPGA.</td>
</tr>
<tr>
<td>I2C_MAIN_INT_B</td>
<td>Output</td>
<td>B31</td>
<td>IO_L19N_T3L_N1_DBC_AD9N_75</td>
<td>Slave I2C active-Low system interrupt output from FPGA to satellite controller.</td>
</tr>
<tr>
<td>I2C_MUX0_INTB_FPGA</td>
<td>Input</td>
<td>D31</td>
<td>IO_L16N_T2U_N7_QBC_AD3N_75</td>
<td>Slave I2C active-Low interrupt output from FPGA to satellite controller.</td>
</tr>
<tr>
<td>PEX_PWRBRKIN</td>
<td>Input</td>
<td>C32</td>
<td>IO_L17N_T2U_N9_AD10N_75</td>
<td>PEX_PWRBRKIN active-Low input from PCIe connector signaling PCIe card to shut down card power in server failing condition.</td>
</tr>
</tbody>
</table>
Table 7: Bank 75 SLR2 Miscellaneous I/O Descriptions (cont’d)

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Direction</th>
<th>Pin</th>
<th>I/O</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCIE_PERST</td>
<td>Input</td>
<td>BH26</td>
<td>IO_L13P_T2L_N0_GC_QBC_67</td>
<td>PCIe_PERSTN active-Low input from PCIe connector to FPGA to detect presence.</td>
</tr>
<tr>
<td>HBM_CATTRIP</td>
<td>Output</td>
<td>D32</td>
<td>IO_L17P_T2U_N8_AD10P_75</td>
<td>HBM_CATTRIP active-High indicator to satellite controller to indicate the HBM has exceeded its maximum allowable temperature. This signal is not a dedicated FPGA output and is a derived signal in RTL. Making this signal active shuts off the FPGA power rails.</td>
</tr>
<tr>
<td>DDR4_RESET_GATE</td>
<td>Output</td>
<td>H33</td>
<td>IO_L7P_T1L_N0_QBC_AD13P_75</td>
<td>DDR4_RESET_GATE active-High output from FPGA to hold all external DDR4 interfaces in self-refresh. This output disconnects the memory interface reset and holds it in active and pulls the clock enable signal on the memory interfaces. See Fast Calibration and Daisy Chaining Functions in DDR4 Memory Interfaces application note (XAPP1321) for details on self-refresh mode.</td>
</tr>
<tr>
<td>USER_SI570_CLOCK_P</td>
<td>Input</td>
<td>G30</td>
<td>IO_L12P_T1U_N10_GC_75</td>
<td>General purpose LVDS 156.250 MHz clock input from SI570 programmable clock. The I2C interface is not accessible from the UltraScale+ device.</td>
</tr>
<tr>
<td>USER_SI570_CLOCK_N</td>
<td>Input</td>
<td>F30</td>
<td>IO_L12N_T1U_N11_GC_75</td>
<td>General purpose LVDS 156.250 MHz clock input from SI570 programmable clock. The I2C interface is not accessible from the UltraScale+ device.</td>
</tr>
<tr>
<td>SYS_CLK3_P</td>
<td>Input</td>
<td>G31</td>
<td>IO_L11P_T1U_N8_GC_75</td>
<td>General purpose LVDS 100.000 MHz clock reference derived from an onboard free running oscillator.</td>
</tr>
<tr>
<td>SYS_CLK3_N</td>
<td>Input</td>
<td>F31</td>
<td>IO_L11N_T1U_N9_GC_75</td>
<td>General purpose LVDS 100.000 MHz clock reference derived from an onboard free running oscillator.</td>
</tr>
</tbody>
</table>
Chapter 6

Known Issues and Limitations

AR 71752 is the master answer record for the Alveo Data Center accelerator cards. For Technical Support, open a Support Service Request.
Appendix A

Regulatory and Compliance Information

This product is designed and tested to conform to the European Union directives and standards described in this section.

CE Directives

2014/35/EC, Low Voltage Directive (LVD)
2014/30/EC, Electromagnetic Compatibility (EMC) Directive

CE Standards

EN standards are maintained by the European Committee for Electrotechnical Standardization (CENELEC). IEC standards are maintained by the International Electrotechnical Commission (IEC).

Electromagnetic Compatibility


This is a Class A product. In a domestic environment, this product can cause radio interference, in which case the user might be required to take adequate measures.

Safety

Compliance Markings

In August of 2005, the European Union (EU) implemented the EU Waste Electrical and Electronic Equipment (WEEE) Directive 2002/96/EC and later the WEEE Recast Directive 2012/19/EU. These directives require Producers of electronic and electrical equipment (EEE) to manage and finance the collection, reuse, recycling and to appropriately treat WEEE that the Producer places on the EU market after August 13, 2005. The goal of this directive is to minimize the volume of electrical and electronic waste disposal and to encourage re-use and recycling at the end of life.

Xilinx has met its national obligations to the EU WEEE Directive by registering in those countries to which Xilinx is an importer. Xilinx has also elected to join WEEE Compliance Schemes in some countries to help manage customer returns at end-of-life.

If you have purchased Xilinx-branded electrical or electronic products in the EU and are intending to discard these products at the end of their useful life, please do not dispose of them with your other household or municipal waste. Xilinx has labeled its branded electronic products with the WEEE Symbol to alert our customers that products bearing this label should not be disposed of in a landfill or with municipal or household waste in the EU.

This product complies with Directive 2002/95/EC on the restriction of hazardous substances (RoHS) in electrical and electronic equipment.

Appendix B

Additional Resources and Legal Notices

Xilinx Resources

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

Documentation Navigator and Design Hubs

Xilinx® Documentation Navigator (DocNav) provides access to Xilinx documents, videos, and support resources, which you can filter and search to find information. To open DocNav:

- From the Vivado® IDE, select Help → Documentation and Tutorials.
- On Windows, select Start → All Programs → Xilinx Design Tools → DocNav.
- At the Linux command prompt, enter docnav.

Xilinx Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In DocNav, click the Design Hubs View tab.
- On the Xilinx website, see the Design Hubs page.

Note: For more information on DocNav, see the Documentation Navigator page on the Xilinx website.

References

These documents provide supplemental material useful with this guide:
Product Websites

The most up-to-date information related to the Alveo™ U280 card and documentation is available on the Alveo U280 Data Center Accelerator Card.

Supplemental Documents

The following Xilinx documents provide supplemental material useful with this guide.

- AXI High Bandwidth Controller LogiCORE IP Product Guide (PG276)
- UltraFast Design Methodology Guide for the Vivado Design Suite (UG949)
- Getting Started with Alveo Data Center Accelerator Cards (UG1301)
- UltraScale Architecture Configuration User Guide (UG570)
- Vivado Design Suite User Guide: Programming and Debugging (UG908)
- Virtex UltraScale+ FPGA Data Sheet: DC and AC Switching Characteristics (DS923)
- UltraScale Architecture-Based FPGAs Memory IP LogiCORE IP Product Guide (PG150)
- UltraScale Architecture PCB Design User Guide (UG583)

Additional Links

The following links provide supplemental material useful with this guide.

- Xilinx, Inc: https://www.xilinx.com
- Micron Technology: http://www.micron.com
  (MTA18ASF2G72PZ-2G3B1IG, MT25QU01GBB8E12-OAAT)
- AB-557-03 Data Sheet: https://abracon.com/Oscillators/AB-557-03.pdf
  (FT4232HQ)
- QSFP+ module: https://members.snia.org/document/dl/25969
Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.
Appendix B: Additional Resources and Legal Notices

Copyright

© Copyright 2019-2020 Xilinx, Inc. Xilinx, the Xilinx logo, Alveo, Artix, Kintex, Spartan, Versal, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos. PCI, PCIe, and PCI Express are trademarks of PCI-SIG and used under license. AMBA, AMBA Designer, Arm, ARM1176JZ-S, CoreSight, Cortex, PrimeCell, Mali, and MPCore are trademarks of Arm Limited in the EU and other countries. All other trademarks are the property of their respective owners.