Configuration via Protocol (CvP) Implementation in Altera FPGAs

User Guide
Chapter 1. Overview
Getting Started ................................................. 1–1
Embedded PCIe Hard IP Blocks in Altera FPGAs ...................... 1–1
  Autonomous PCIe Hard IP Block ..................................... 1–1
CvP Scheme ....................................................... 1–2

Chapter 2. FPGA Configuration in CvP
CvP Setup Interfaces .................................................. 2–1
  Conventional Configuration Scheme .................................... 2–1
  PCIe Link ............................................................... 2–1
  CvP Configuration Image ............................................... 2–2
  CvP Modes .............................................................. 2–3
    CvP Off Mode ......................................................... 2–4
    CvP Initialization and Update mode ................................... 2–4
    CvP Update mode ...................................................... 2–5

Chapter 3. FPGA Fabric Image Update in CvP
Fabric Image Update ................................................... 3–1

Chapter 4. CvP Topologies
Single End Point ...................................................... 4–1
Multiple End Points ................................................... 4–2
Mixed Chain ............................................................. 4–4
Daisy Chain .............................................................. 4–6

Chapter 5. Design Considerations
Pin Descriptions ....................................................... 5–1
CvP Configuration Features ............................................ 5–3
  Data Compression ...................................................... 5–3
  Data Encryption ....................................................... 5–3
  FPGA Image Update ................................................... 5–3
Designing CvP for an Open System .................................. 5–4
  FPGA Power Supplies Ramp Time Requirement .................... 5–4
  PCIe Wake-Up Time Requirement in an Open System .............. 5–5
Supported Configuration Schemes for CvP Modes in Stratix V Devices .................................................... 5–8

Chapter 6. Software Support
Device and Pin Options Setting ...................................... 6–1
Stratix V Hard IP for PCI Express version 11.0 ....................... 6–2

Additional Information
Document Revision History ........................................... Info–1
How to Contact Altera ................................................ Info–1
Typographic Conventions ............................................. Info–1
1. Overview

This user guide describes the features and implementations of the Configuration via Protocol (CvP) scheme in supported Altera® FPGA families. This user guide only shows you how to use the CvP scheme in the supported FPGA, and does not explain the PCI Express® (PCIe®) protocol.

Getting Started

Use this user guide in conjunction with an understanding of the following PCI Express specifications:

- PCI Express Base Specification for 1.1
- PCI Express Base Specification for 2.1
- PCI Express Base Specification for 3.0
- PCI Express CEM Specification for 2.0

Embedded PCIe Hard IP Blocks in Altera FPGAs

Altera FPGAs contain embedded PCIe hard IP blocks that you can configure as end points or a root complex for PCIe applications. However, you cannot use the embedded PCIe hard IP block as a root complex when it is configured for CvP.

To establish the PCIe link between a root complex and an end point successfully, the end point must be functional within the time period set by the PCIe power-up and wake-up timing specification.

Autonomous PCIe Hard IP Block

After the PCIe hard IP block is configured, it is in autonomous mode. The autonomous PCIe feature allows the hard IP block in the FPGA to start link training before the fabric is configured. This feature helps the FPGA to accommodate to the PCIe timing requirement in both the open and closed systems.

In an open system, all the electronic systems attached to the open system must obey the PCIe wake-up time requirement as defined in the PCIe specifications. In a closed system, the electronic system attached to the closed system does not obey the PCIe wake-up time requirement as defined in the PCIe specifications.

In case the FPGA fabric is not fully functional when the PCIe host starts accessing the PCIe hard IP block to perform the data transaction, the autonomous PCIe hard IP block responds to the PCIe host with a configuration retry status (CRS) transaction. When the PCIe host receives the CRS transaction, it continues the attempt to access the end point for one second before it recognizes the end point is faulty. This allows more time for the fabric configuration and avoid the PCIe host to recognize the end point is faulty before the fabric configuration completes. All PCIe hard IP blocks are part of the periphery image. Regardless of the CvP mode, the periphery image is always configured before the fabric configuration.
CvP Scheme

The CvP is a new FPGA configuration scheme supported in Stratix® V devices. It uses the autonomous PCIe hard IP block feature of the embedded PCIe hard IP block to allow the FPGA to meet the PCIe power-up timing specification. The benefit of the autonomous PCIe hard IP block is that the embedded PCIe core is operational before the FPGA fabric is configured. Conventionally, the embedded PCIe hard IP block can only be operational after the full FPGA image is fully configured.

The CvP scheme offers a solution for configuring the FPGA fabric through the PCIe link. The following are the benefits of using the CvP scheme:

- **Reduced cost**—The FPGA fabric configuration data is stored in the memory of the PCIe host and the external flash is used only to store the periphery configuration data. This can reduce the flash memory size required because the periphery configuration data file size is smaller when compared with the full FPGA image configuration data, which saves system cost.

- **Design protection**—CvP ensures the PCIe host can exclusively access the FPGA fabric image (the most important portion of the FPGA configuration data). This provides better protection for the FPGA against unauthorized design tampering or copying.

- **Image update without system down time**—CvP allows the FPGA fabric to be updated through the PCIe link without a host reboot or FPGA full chip reinitialization.
This chapter describes the interfaces involved in the CvP setup, CvP configuration image, and types of CvP modes.

**CvP Setup Interfaces**

CvP setup involves two interfaces: the conventional configuration interface and the PCIe link.

**Conventional Configuration Scheme**

The conventional configuration interface is to connect the FPGA to a JTAG configuration interface, the configuration device for the Active Serial (AS) configuration scheme, or the configuration host for both the Fast Passive Parallel (FPP) and Passive Serial (PS) configuration schemes.

For more information about the conventional configuration scheme, refer to the *Configuration, Design Security, and Remote System Upgrades in Stratix V Devices* chapter in volume 2 of the *Stratix V Device Handbook*.

**PCIe Link**

The PCIe link connects the PCIe hard IP block to the PCIe host. Only one designated PCIe hard IP block has the interface to the internal configuration control block, which you can use for CvP. You can also use this designated PCIe hard IP block to perform other PCIe applications after the FPGA enters user mode. The other PCIe hard IP blocks do not have the dedicated interface to the internal configuration control block; therefore you can use these other PCIe hard IP blocks for PCIe applications only and not to perform CvP.

For Stratix V devices, the PCIe hard IP block located at the bottom left from the top view (which is GXB_L0 bank) of the FPGA has the connection to the internal configuration control block, which you can use for CvP.

For more information about the PCIe hard IP block location, refer to the *Transceiver Architecture in Stratix V Devices* chapter in volume 3 of the *Stratix V Device Handbook*. 
Figure 2–1 shows the high-level hardware setup for FPGA configuration in the CvP scheme.

**Figure 2–1. CvP Hardware Setup**

Notes to Figure 2–1:

1. The configuration interface can be any of the supported configuration schemes such as JTAG, FPP, PS, or AS. The choice of configuration device depends on the chosen configuration scheme.
2. This PCIe hard IP block is for CvP and other PCIe applications.
3. This PCIe hard IP blocks only support PCIe applications. You cannot use it for CvP.

**CvP Configuration Image**

A full FPGA configuration image has two portions: the periphery image and the fabric image. The periphery image contains the configuration data for PCIe hard IP block and I/O buffer settings, which includes all user I/Os and dedicated transceiver channels. Fabric image contains the configuration data of the logic elements (LEs), digital signal processing (DSP) blocks, and embedded memory of the design.

In the conventional configuration scheme supported in previous FPGA device families, the periphery image and fabric image are combined in a single configuration file. You can separate these images into two different configuration files with the CvP scheme.
Figure 2–2 shows the periphery and fabric location of the FPGA and does not represent the transceiver and I/O bank location of the FPGA.

For more information about the location of the transceiver banks and I/O banks in the FPGA, refer to the High-Speed Differential I/O Interfaces and DPA in Stratix V Devices chapter in volume 1 of the Stratix V Device Handbook.

**CvP Modes**

FPGA configuration in the CvP scheme supports three different modes. Depending on the CvP mode, the conventional configuration interface and the PCIe link are used to transfer different FPGA configuration data. The CvP modes are described in the following sections.

Table 2–1 lists the features of each CvP mode.

<table>
<thead>
<tr>
<th>Feature</th>
<th>CvP Off Mode</th>
<th>CvP Initialization and Update Mode</th>
<th>CvP Update Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCIe Link Data Rate (2), (3)</td>
<td>Gen1, Gen2, Gen3</td>
<td>Gen1, Gen2</td>
<td>Gen1, Gen2, Gen3</td>
</tr>
<tr>
<td>PCIe Link Usage</td>
<td>PCIe application in user mode</td>
<td>Initial FPGA fabric configuration, FPGA fabric image update, and PCIe application in user mode</td>
<td>FPGA fabric image update and PCIe application in user mode</td>
</tr>
<tr>
<td>FPGA Configuration Method</td>
<td>Full configuration through conventional configuration schemes (AS, PS, FPP, and JTAG)</td>
<td>Periphery configuration through conventional schemes (AS, PS, FPP, and JTAG). Fabric configuration through the PCIe link</td>
<td>Full configuration through conventional configuration schemes (AS, PS, FPP, and JTAG)</td>
</tr>
</tbody>
</table>
CvP Off Mode

In CvP Off mode, CvP is off and the FPGA is fully configured through the conventional configuration schemes. The configuration image used in CvP Off mode is a single configuration file that contains the configuration data for the periphery image and fabric image. This configuration image is stored in an external configuration device and is loaded to the FPGA after device power up. The CONF_DONE pin is used to indicate if the full configuration image is fully loaded into the FPGA. After the full configuration completes, the FPGA enters initialization mode and user mode. If INIT_DONE signal is enabled, the INIT_DONE signal goes high after the initialization completes and FPGA enters user mode.

You can use the PCIe hard IP blocks for PCIe applications in user mode.

CvP Initialization and Update mode

There are two steps to configure the FPGA using CvP Initialization and Update mode: periphery configuration and fabric configuration. Therefore, the configuration image in CvP Initialization and Update mode is split into two different files. The first configuration file contains the periphery image (stored in the external configuration device) and the second configuration file contains the fabric image (stored in a memory that is accessible by the PCIe host).

During device power up, the periphery image is loaded from the external configuration device to the FPGA through the conventional configuration interface. The configuration bits for the PCIe hard IP blocks are part of the periphery image.

After the periphery configuration is completed, the CONF_DONE signal goes high and allows the FPGA to start the PCIe link training. When the PCIe link training completes, the PCIe link transitions to L0 state and the PCIe host begins to initiate the fabric configuration through the PCIe link. After the fabric is successfully configured, the CVP_CONFDONE pin goes high, signifying the completion of the full FPGA configuration. In CvP Initialization and Update mode, you must observe both CONF_DONE and CVP_CONFDONE to verify that the FPGA is fully configured. After the FPGA is fully configured, the FPGA enters initialization and user mode. If INIT_DONE signal is enabled, the INIT_DONE signal goes high after the initialization completes and FPGA enters user mode.
After the device enters user mode, you can use the PCIe link for normal PCIe transactions. In CvP Initialization and Update mode, you can also use the PCIe link to perform an FPGA fabric image update. To achieve this, generate multiple FPGA fabric images in the Quartus® II software that keep the existing same periphery functionality. For more information about fabric image updates in CvP Initialization and Update mode, refer to the “FPGA Fabric Image Update in CvP” on page 3–1.

### CvP Update mode

In CvP Update mode, the FPGA is fully configured through a conventional configuration scheme on device power up. A single configuration file that contains the full FPGA configuration image is stored in the external configuration device and is loaded to the FPGA. The `CONF_DONE` signal goes high after the completion of the full FPGA configuration. After the FPGA is fully configured, the FPGA enters initialization and user mode. If `INIT_DONE` signal is enabled, the `INIT_DONE` signal goes high after the initialization completes and FPGA enters user mode. After the device enters user mode, the PCIe links are available for the normal PCIe application.

Besides the normal PCIe applications, you can use the PCIe link for an FPGA fabric image updates. To achieve this, you can generate multiple FPGA fabric images in the Quartus II software that keep the same periphery functionality.

For the image update in CvP Update mode, refer to the “FPGA Fabric Image Update in CvP” on page 3–1.
This chapter describes the process of the FPGA fabric image update through CvP.

**Fabric Image Update**

After the FPGA enters user mode, the PCIe host triggers an FPGA fabric image update through the PCIe link. FPGA fabric image update is supported in CvP Initialization and Update mode, and CvP Update mode. You cannot update the periphery image through CvP.

Altera recommends that you fix the periphery functionality for each fabric image design in CvP. After each fabric image update, the core fabric designs keep the same functionality to the periphery interface.

You can update the PLLs and transceivers using dynamic reconfiguration. For more information about dynamic reconfiguration, refer to the *Dynamic Reconfiguration in Stratix V Devices* chapter in volume 3 of the *Stratix V Device Handbook*.

**Figure 3–1** shows an example of how you can create multiple FPGA fabric images for use with CvP Update.

**Figure 3–1. Example of Multiple User Design Planning for CvP Fabric Image Update**

In **Figure 3–1**, the periphery design is fixed from user design revision 1 to revision n. However, the fabric design changes in every user design revision. These fabric designs must have the same functionality to the periphery interface. You must ensure the system keeps the same periphery functionality for each fabric image that is created.

In the Quartus II software, you can generate multiple fabric images within a single design project. However, you must generate only one periphery image for that design project because the periphery image must remain the same for all the fabric images to work with CvP Update. The periphery image is stored in the configuration device and you may not modify or update the periphery image during the fabric image update. The multiple fabric images are stored in memory that is accessible by the PCIe host, as shown in **Figure 3–2**.
If a fabric image update attempt is triggered, the \texttt{CVP\_CONF_DONE} pin is pulled low, which indicates that a fabric image update has started. The FPGA fabric is reinitialized and reconfigured with the new fabric image. During the fabric image update through a PCIe link, the \texttt{n\_CONFIG} and \texttt{n\_STATUS} pins of the FPGA are logic high. When the fabric image update completes, the \texttt{CVP\_CONF\_DONE} pin is released high again and indicates that the FPGA has entered user mode.

Support for multiple fabric images creation will be available in a future version of the Quartus II software.

Figure 3–2 shows the arrangement of the periphery image and multiple fabric images stored for fabric image update.

**Figure 3–2. Periphery and Fabric Images Storage Arrangement in CvP Fabric Image Update (Note 1)**

![Figure 3–2. Periphery and Fabric Images Storage Arrangement in CvP Fabric Image Update (Note 1)](image)

**Note to Figure 3–2:**
(1) The periphery image remains the same for different fabric image updates. If you change the periphery image, you must reprogram the configuration device with the new periphery image.
This chapter describes some of the topologies that can be used with CvP. The CvP is supported for single FPGA or multiple FPGAs configurations.

**Single End Point**

Use the single end point topology for a single FPGA configuration. In this topology, the PCIe link involves only one PCIe end point in the FPGA and the PCIe root complex in the host. Depending on the CvP mode you select, the conventional configuration interface is used to configure the periphery image or the full FPGA image after system power up. The PCIe link can be used for fabric image configurations or updates, in addition to the usual PCIe link application.

Figure 4–1 shows the single end point topology.

**Figure 4–1. Single End Point Topology in CvP**

Notes to Figure 4–1:

1. The configuration interface can be any of the supported configuration schemes such as AS, PS, FPP, or JTAG. The **MSEL [4..0]** are used to indicate the chosen configuration scheme.
2. The PCIe link includes the **PERST#** signal and other PCIe signals as defined in the PCIe base specification.
Multiple End Points

Use the multiple end points topology to configure multiple FPGAs. In this topology, one root complex connects to multiple FPGA end points through a PCIe switch. The PCIe switch controls the fabric image configuration through the PCIe link to the targeted FPGA end point. This topology provides flexibility for you to select the FPGA that you must configure or update through the PCIe link. There is no limitation on the number of FPGAs that can be configured using this topology. The root complex must have the ability to respond to the PCIe switch and direct the configuration data transaction to the designated end point based on the end point media access control (MAC) address reported by the PCIe switch.
Figure 4–2 shows the multiple end points topology.

Figure 4–2. Multiple End Points Topology in CvP

Notes to Figure 4–2:
(1) The configuration interface can be any of the supported configuration schemes such as AS, PS, FPP, and JTAG. The MSEL[4..0] signals are used to indicate the configuration scheme.
(2) The PCIe link includes the PERST# signal and other PCIe signals as defined in the PCIe base specification.
**Mixed Chain**

Use mixed chain topology to support multiple FPGA configurations. In this topology, the root complex of the host only responds to the end point in the master FPGA (the first FPGA in the chain) through the PCIe link. You can configure the slave FPGAs in the chain through a PS or FPP configuration scheme. Only the master FPGA is configured in CvP mode.

You must design user IP in the master FPGA to fetch the data from the root complex to the slave devices in the chain. The data is latched out from the master FPGA through GPIOs and latched into the slave FPGAs through the PS or FPP configuration pins such as DCLK, DATA line, or DATA bus.

The DCLK, DATA bus, nCONFIG, nSTATUS, and CONF_DONE pins of the slave devices are tied together, allowing the slave devices to enter user mode at the same time. If there is a configuration error during the slave device configuration, the slave device chain reinitializes and reconfigures by having the nSTATUS line pulled low and released by the FPGAs again. You must ensure there is a suitable line buffering on the DCLK and data bus if there are more than four slave devices in the chain.

You can cascade the slave devices in two ways:

- If the slave devices are configured with different configuration files, the nCEO pin of one slave device is connected to the nCE pin of the next slave device in the chain. If the first slave device completes the configuration, the device pulls the nCEO pin that enables the next slave device. The next slave device starts its configuration and pulls the nCEO pin low to enable the subsequent device. This process continues until the last slave device in the chain is configured. You can leave the nCEO pin of the last device unconnected or use as a user I/O, as shown in Figure 4–3 with the slave devices configured in FPP mode.

- If the slave devices are configured with the same configuration file for identical user applications, the nCE pins of the slave devices are connected to the nCEO pins of the master device. The nCEO pins of the slave devices are left unconnected or used as user I/Os. In this topology, all the slave devices are configured at the same time, as shown in Figure 4–4 with the slave devices configured in FPP mode.
Figure 4–3 shows the mixed chain topology.

Figure 4–3. Mixed Chain Topology in CvP (Different Configuration Files for Slave Device Configurations)

Notes to Figure 4–3:

1. The configuration interface can be any of the supported configuration schemes such as AS, PS, FPP, or JTAG. The MSEL[4..0] signals are used to indicate the configuration scheme.
2. User IP is a user-designed IP programmed in the first FPGA to fetch data from the root complex through the PCIe link, which is sent to the slave FPGAs in the chain through a PS or FPP configuration interface.
3. DCLK out and DATA out are the GPIO pins used to fetch the PS or FPP configuration data. The nCONFIG out and nCEO out are the GPIO pins used to connect to the nCONFIG and nCE pins of the slave device.
4. Configuration data input to the slave FPGAs through PS or FPP configuration pins.
5. Connect the MSEL pins to the PS or FPP scheme.
6. You can leave the nCEO pin of the last FPGA in the chain unconnected or use it as a GPIO.
7. The PCIe link includes the PERST# signal and other PCIe signals as defined in the PCIe base specification.
Figure 4–4 shows the mixed-chain topology in CvP with all the slave devices configured at the same time.

**Figure 4–4. Mixed Chain Topology in CvP (Single Configuration File for Slave Device Configurations)**

Notes to Figure 4–4:

1. The configuration interface can be any of the supported configuration schemes such as AS, PS, FPP, or JTAG. The MSEL[4..0] signal and choice of configuration device depend on the chosen configuration scheme.
2. User IP is a user-designed IP programmed in the first FPGA to fetch data from the root complex through the PCIe link, which is sent to the slave FPGAs in the chain through a PS or FPP configuration interface.
3. DCLK_out and DATA_out are the GPIO pins used to fetch the PS or FPP configuration data. The nCONFIG_out and nCEO_out are the GPIO pins used to connect to the nCONFIG and nCE pins of the slave device.
4. Configuration data input to the slave FPGAs through PS or FPP configuration pins.
5. Connect the MSEL pins to the PS or FPP scheme.
6. You can leave the nCEO pin of the last FPGA in the chain or use it as a GPIO.
7. The PCIe link includes the PERST# signal and other PCIe signals as defined in the PCIe base specification.

**Daisy Chain**

A daisy chain is a CvP topology for configuring multiple FPGAs. Configuration devices are attached to each FPGA for periphery configuration (CvP Initialization and Update mode) or full FPGA configuration (CvP Update mode) after the system powers up. The nCE, nSTATUS, nCONFIG, CONFDONE, and CvP_CONF_DONE pins of the FPGAs are not tied together, allowing the FPGAs to receive the configuration data individually from their respective configuration device.

Each FPGA in the daisy chain has two PCIe IP cores configured as an end point and a root complex, respectively. In this topology, the root complex in the host responds to the end point of the first FPGA in the chain. The fabric image of the first FPGA is configured through the PCIe link between the host and its end point. After the configuration is completed, the first FPGA enters user mode and the user IP in the first FPGA initiates the fabric configuration of the subsequent device. The process continues until the last device in the chain is configured. You must design a user IP as part of the FPGA fabric image. The user IP routes data from the host to the FPGA root...
complexes. The root complex of the first FPGA is connected to the end point of the subsequent FPGA through the PCIe link. The fabric image of the second FPGA is fetched from the host and is routed through the first FPGA before it is loaded into the second FPGA through the PCIe link. This process continues until the last FPGA in the chain is configured.

Figure 4–5 shows the daisy chain topology.

**Notes to Figure 4–5:**

1. The configuration interface can be any of the supported configuration schemes such as AS, PS, FPP, or JTAG. The MSEL[4..0] signal is used to indicate the chosen configuration scheme.

2. User IP is a user-design IP programmed in the first FPGA to data from the root complex through the PCIe link and send to the slave FPGAs in the chain through FPP configuration interface.

3. The PCIe link includes the PERST# signal and other PCIe signals as defined in the PCIe base specification.
This section discusses the hardware design considerations if your FPGA design is configured in CvP mode.

**Pin Descriptions**

Depending on the CvP mode chosen, configuration involves the conventional configuration interface and the PCIe link. This section covers only the CvP pins that are not included in the conventional configuration interface and the typical PCIe link.

For more information about the configuration pins involved in the conventional configuration interface, refer to the *Configuration, Design Security, and Remote System Upgrades in Stratix V Devices* chapter in volume 2 of the *Stratix V Device Handbook*.

For more information about the typical PCIe link signals, refer to the *PCI Express Base Specification* and the transceiver chapters in volume 3 of the *Stratix V Device Handbook*. 
Table 5–1 lists the additional CvP pin descriptions and connections that are not listed in the *Configuration, Design Security, and Remote System Upgrades in Stratix V Devices* chapter and the PCI Express Base Specification.

Table 5–1. CvP Pin Descriptions and Connections

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>Pin Type</th>
<th>Pin Description</th>
<th>Pin Connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>CvP_CONFDONE</td>
<td>Output</td>
<td>Driven low during fabric configuration and released or driven high after the completion of the fabric configuration.</td>
<td>If this pin is set as dedicated output, the $V_{CCPGM}$ power supply must meet the input voltage specification of the receiving side. If this pin is set as an open drain output, the pin must be connected to an external 10 k$\Omega$-pull-up resistor to the $V_{CCPGM}$ power supply or a different pull-up voltage that meets the input voltage specification of the receiving side. This gives an advantage on the voltage leveling.</td>
</tr>
</tbody>
</table>
| nPERSTL0  | Input     | This pin is connected to the PCIe hard IP block as a dedicated fundamental reset pin for PCIe usage. If the signal is low, the transceivers and dedicated PCIe (CvP) hard IP block are in reset mode. If this signal is high, the transceivers and dedicated PCIe (CvP) hard IP block are released from the reset mode. If the PCIe hard IP block is not in use, you can use this pin as a user I/O pin. | For Stratix V devices, connect the nPERSTL0 to the PERST_N pin of the PCIe slot. During CvP, this pin is powered by the $V_{CCPGM}$ power supply. In the case that the $V_{CCPGM}$ power supply is not at 3.3 V, you can still drive an input signal to this pin at 3.3-V volts without requiring a voltage level shifter with the following requirements to conform to the PCIe electrical standards:  
| | | ■ The input signal must meet the $V_{IH}/V_{IL}$ specification of 3.3-V LVTTL I/O standard.  
| | | ■ The input signal must not exceed the overshoot specification for 100% operation as stated in the *Stratix V Device Datasheet* in volume 1 of the *Stratix V Device Handbook*. | |

Configuration via Protocol (CvP) Implementation in Altera FPGAs User Guide  
May 2011  Altera Corporation
CvP Configuration Features

CvP supports configuration features such as data decompression, decryption, and partial reconfiguration.

Table 5–2 lists the configuration features supported in each CvP mode.

Table 5–2. CvP with Configuration Features

<table>
<thead>
<tr>
<th>CvP Mode</th>
<th>Decompression (1)</th>
<th>Decryption (1)</th>
<th>Remote System Upgrade</th>
<th>Partial Reconfiguration (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CvP Off</td>
<td>Supported</td>
<td>Supported</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>CvP Initialization and Update</td>
<td>Supported</td>
<td>Supported</td>
<td>Not supported (2)</td>
<td>Supported</td>
</tr>
<tr>
<td>CvP Update Mode</td>
<td>Supported</td>
<td>Supported</td>
<td>Not supported (2)</td>
<td>Supported</td>
</tr>
</tbody>
</table>

Notes to Table 5–2:
(1) Feature support is not available in the Quartus II software version 11.0. This feature will be available in a future release of the Quartus II software.
(2) The system upgrade feature is supported through fabric image update through the PCIe link.

Data Compression

The data compression reduces the size of the fabric image. The periphery image is not compressed. Enabling data compression in CvP Off mode and CvP Update mode can help reduce both the external configuration memory required for full FPGA image storage and the configuration time of the full FPGA configuration.

If the data compression is enabled in CvP Initialization and Update mode, the external configuration memory required for periphery image storage remains the same but the memory size required for the fabric image in the PCIe host is reduced.

Data Encryption

Use the data encryption feature to encrypt the fabric image with the Advanced Encryption Standard (AES) algorithm to protect the data against unauthorized access. The periphery image is not encrypted by this feature. To configure the FPGA with encrypted data, a security key is pre-programmed to the FPGA and is used to decrypt the incoming bitstream.

FPGA Image Update

You can achieve the FPGA image update in CvP Off mode with the normal remote system upgrade feature. FPGA fabric image update in CvP Initialization and Update mode, and CvP Update mode is performed through the PCIe link. Partial reconfiguration is supported in all the CvP modes. However, software support for this feature will only be available in a future release of the Quartus II software.
Designing CvP for an Open System

In an open system, all the electronic systems attached to the open system must obey the PCIe wake-up time requirement as defined in the PCIe specifications. This section describes the requirements for FPGA power supply ramp time, PCIe wake-up time requirement in an open system, and configuration scheme selection for CvP modes in Stratix V devices.

**FPGA Power Supplies Ramp Time Requirement**

This section describes the FPGA power supplies ramp-up time requirement to accommodate a CvP in an open system.

The POR circuit generates a POR signal that keeps the device in a reset state until the power supply outputs are within the operating range. If power is applied to a Stratix V device, a POR event occurs if the POR monitored power supplies reach the recommended operating range within the maximum power supply ramp time ($t_{RAMP}$).

To use CvP, the total time must be within 10 ms, from the first power supply ramp-up to the last power supply ramp-up (refer to Figure 5–1). You must set the PORSEL pin high for fast POR, in which the POR delay time is in the range of 4–12 ms, leaving enough time after the POR trip for the PCIe to start initialization and configuration.

Figure 5–1 shows the power supplies ramp-up time and POR.

*Figure 5–1. Power Supplies Ramp-Up Time and POR*

For more information about POR, refer to the *Hot Socketing and Power-On Reset in Stratix V Devices* chapter in volume 2 of the *Stratix V Device Handbook*. 
**PCIe Wake-Up Time Requirement in an Open System**

To accommodate the open system of PCIe applications, the PCIe link connected to the FPGA must meet the PCIe wake-up timing specification that must transition from power-on to the link active (L0) state within 200 ms. The overall timing from FPGA power up to the PCIe hard IP block in the FPGA is ready for link training must be within 120 ms.

For CvP Off mode and CvP Update mode, the full FPGA image is configured through the conventional configuration schemes. To accommodate your system into an open system, the full FPGA configuration in CvP must be within 120 ms. Therefore, you must choose the right conventional configuration scheme for your device to ensure the configuration time is able to meet the 120-ms requirement. For more information about the supported configuration scheme that you can choose to design an open system with CvP Off mode and CvP Update mode, refer to Table 5–3.

Figure 5–2 shows the PCIe power-up sequence and the FPGA configuration timing in CvP Off mode and CvP Update mode.

**Figure 5–2. PCIe Timing Sequence in CvP Off Mode and CvP Update Mode**

Notes to Figure 5–2:

1. To ensure successful configuration, all the POR-monitored power supplies to the FPGA must ramp up monotonically to operating range within the 10 ms ramp-up time.

2. One of the auxiliary signals specified in the PCIe electromechanical specification. The PERST# signal is sent from host to the FPGA device and it is used to indicate if the power supplies of the FPGA device are within their specified voltage tolerance and are stable. It also initializes the FPGA state machines and other logic after power supplies are stabilized.

3. When the PCIe link turns active, the PCIe function starts up. The PCIe link supports PCIe application in user mode for CvP Off mode and CvP Update mode. For CvP Update mode, you can also use the PCIe link for fabric image update.
Table 5–3 lists the power-up sequence timing in CvP Off mode and CvP Update mode, as shown in Figure 5–2.

<table>
<thead>
<tr>
<th>Timing Sequence in Figure 5–2</th>
<th>Timing Range (ms)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>10</td>
<td>Maximum ramp-up time requirement for all POR-monitored power supplies in the FPGA to reach their respective operating range</td>
</tr>
<tr>
<td>b</td>
<td>4-12</td>
<td>FPGA POR delay time</td>
</tr>
<tr>
<td>c</td>
<td>100</td>
<td>Minimum PERST# signal active time</td>
</tr>
<tr>
<td>d</td>
<td>20</td>
<td>Minimum PERST# signal inactive time before the PCIe link enters training state</td>
</tr>
<tr>
<td>e</td>
<td>120</td>
<td>Maximum time from the FPGA power up to the end of the full FPGA configuration in CvP Off mode and CvP Update mode</td>
</tr>
<tr>
<td>f</td>
<td>100</td>
<td>Minimum PERST# signal inactive time before the PCIe link enter active state</td>
</tr>
</tbody>
</table>

For CvP Initialization and Update mode, the PCIe hard IP block is guaranteed to meet the 120 ms time limit because the periphery image configuration time is significantly smaller when compared with the full FPGA configuration time. When designing CvP for an open system in CvP Initialization and Update mode, you can choose any of the conventional configuration schemes for the periphery image configuration. However, the ramp-up time for the POR-monitored power supplies of the FPGA must not exceed 10 ms.
Figure 5–3 shows the PCIe power up and the FPGA configuration timing in CvP Initialization and Update mode.

**Notes to Figure 5–3:**

1. To ensure successful configuration, all POR-monitored power supplies to the FPGA must ramp up monotonically to the operating range within the 10 ms ramp-up time.

2. One of the auxiliary signals specified in the PCIe electromechanical specification. The \texttt{PERST#} signal is sent from the host to the FPGA device and it is used to indicate if the power supplies of the FPGA device are within their specified voltage tolerance and are stable. It also initializes the FPGA state machines and other logic after power supplies stabilize.

3. After the PCIe link turns active, the PCIe function starts up. For CvP Initialization and Update mode, the PCIe link supports the FPGA fabric configuration, FPGA fabric image update, and PCIe application in user mode.
Table 5–4 lists the power-up sequence timing in CvP Initialization and Update mode, as shown in Figure 5–3.

### Table 5–4. Power-Up Sequence Timing in CvP Initialization and Update mode

<table>
<thead>
<tr>
<th>Timing Sequence in Figure 5–3</th>
<th>Timing Range (ms)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>10</td>
<td>Maximum ramp-up time requirement for all POR-monitored power supplies in the FPGA to reach their respective operating range</td>
</tr>
<tr>
<td>b</td>
<td>4-12</td>
<td>FPGA POR delay time</td>
</tr>
<tr>
<td>c</td>
<td>100</td>
<td>Minimum PERST# signal active time</td>
</tr>
<tr>
<td>d</td>
<td>20</td>
<td>Minimum PERST# signal inactive time before the PCIe link enters training state</td>
</tr>
<tr>
<td>e</td>
<td>120</td>
<td>Maximum time from the FPGA power up to the end of periphery configuration in CvP Initialization and Update mode</td>
</tr>
<tr>
<td>f</td>
<td>100</td>
<td>Minimum PERST# signal inactive time before the PCIe link enter active state</td>
</tr>
</tbody>
</table>

**Supported Configuration Schemes for CvP Modes in Stratix V Devices**

Figure 5–2 shows that the FPGA configuration time for CvP Off mode and CvP Update mode that is constrained to 120 ms to meet the 200-ms PCIe wake-up timing specification for an open system. You must ensure the chosen configuration scheme for full FPGA configuration in CvP Off mode and CvP Update mode is able to meet the configuration time requirement.
Table 5–5 lists supported configuration schemes for different CvP modes in Stratix V devices to meet the 200 ms PCIe wake-up timing specification.

### Table 5–5. Supported Configuration Schemes for CvP Modes for Stratix V Devices (Note 1), (2)

<table>
<thead>
<tr>
<th>CvP Mode</th>
<th>Stratix V GX</th>
<th>Stratix V GT</th>
<th>Stratix V GS</th>
<th>Stratix V E</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A3</td>
<td>A4</td>
<td>A5</td>
<td>A7</td>
</tr>
<tr>
<td>CvP Off</td>
<td>FPP x16</td>
<td>FPP x16</td>
<td>FPP x32</td>
<td>FPP x32</td>
</tr>
<tr>
<td>CvP Initialization and Update</td>
<td>All conventional configuration schemes (1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CvP Update</td>
<td>FPP x16</td>
<td>FPP x16</td>
<td>FPP x32</td>
<td>FPP x32</td>
</tr>
</tbody>
</table>

**Note to Table 5–5:**

1. For periphery configuration in CvP Initialization and Update mode, all the conventional configuration schemes such as AS, PS, FPP, or JTAG are supported.
2. Data compression and encryption feature are disabled. These features require different data to clock ratio, which causes the total configuration time to prolong and does not meet the 200-ms PCIe wake-up timing specification.

To design CvP in a closed system, you must perform the configuration time estimation to ensure the total FPGA configuration time—either for full FPGA configuration in CvP Off mode and CvP Update mode, or the periphery configuration in CvP Initialization and Update mode—is within the permitted time given by the PCIe host. To calculate or estimate the configuration time for each device density and configuration scheme, refer to the *Configuration, Design Security, and Remote System Upgrades in Stratix V Devices* chapter in volume 2 of the *Stratix V Device Handbook* to obtain the uncompressed raw binary file (.rbf) file size and the supported configuration speed for each device density and configuration scheme.
This chapter discusses the software support for CvP mode in the Quartus II software version 11.0.

Table 6–1 lists the features for CvP support that are not enabled in the Quartus II software version 11.0, but will be available in a future release of the Quartus II software.

<table>
<thead>
<tr>
<th>Upcoming CvP Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming file generation for periphery and fabric image</td>
<td>Generate programming file for periphery image and fabric image.</td>
</tr>
<tr>
<td>CvP with configuration features (compression, encryption, and partial reconfiguration)</td>
<td>Enable compression, encryption, and partial reconfiguration feature in CvP mode.</td>
</tr>
<tr>
<td>CvP programming</td>
<td>Perform CvP programming in the Quartus II Programme.</td>
</tr>
<tr>
<td>PCIe driver</td>
<td>Open source code and guidelines to develop or customize the PCIe driver for CvP.</td>
</tr>
</tbody>
</table>

**Device and Pin Options Setting**

By default, CvP mode is turned off in the Quartus II software. Before compiling your design, you must set the CvP mode and the CvP_CONF_DONE pin behavior accordingly.

Table 6–2 lists the settings you can specify in the Device and Pin Options dialog box to enable CvP mode in your design.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration via Protocol</td>
<td>Off</td>
<td>Enable CvP Off mode. The FPGA is fully configured through the conventional configuration scheme and the PCIe link is used for PCIe applications in user mode.</td>
</tr>
<tr>
<td>Power up and subsequent core configuration</td>
<td></td>
<td>Enable CvP Initialization and Update mode. The FPGA periphery is configured through the conventional configuration scheme and the FPGA fabric is configured through the PCIe link. Use the PCIe link for fabric image update or PCIe applications in user mode.</td>
</tr>
<tr>
<td>Subsequent core reconfiguration</td>
<td></td>
<td>Enable CvP Update mode. The FPGA is fully configured through the conventional configuration scheme. Use the PCIe link for the fabric image update or PCIe applications in user mode.</td>
</tr>
</tbody>
</table>
Table 6–2. CvP Settings in Device and Pin Options (Part 2 of 2)

<table>
<thead>
<tr>
<th>Setting</th>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CvP_CONF_DONE pin</td>
<td>Enable CvP_CONF_DONE</td>
<td>Enable CvP_CONF_DONE. If you turn on this option, the CvP_CONF_DONE pin is used to indicate whether or not the fabric configuration is successfully completed. If you turn off this option, the CvP_CONF_DONE pin functions as a normal user I/O.</td>
</tr>
<tr>
<td></td>
<td>Enable open drain on</td>
<td>Enable CvP_CONF_DONE as an open drain output pin. If you turn on this option, the CvP_CONF_DONE pin functions as an open drain output. If you turn off this option, the CvP_CONF_DONE is a dedicated output pin.</td>
</tr>
<tr>
<td></td>
<td>CvP_CONF_DONE pin</td>
<td></td>
</tr>
</tbody>
</table>

Stratix V Hard IP for PCI Express version 11.0

IP Compiler for PCI Express is the parameter entry tool for the PCIe hard IP block. To use CvP in your design, you must include the hard IP as part of your design. When include this PCIe IP compiler into your design, you must enable the Select the CvP capable HIP block check box and set the parameter settings according to the PCIe link of your system.

For complete information about the IP Compiler for PCIe, refer to the IP Compiler for PCI Express User Guide.

To perform CvP applications, follow these steps:
1. Open the MegaWizard™ Plug-In Manager.
2. In the MegaWizard Plug-In Manager, select Stratix V Hard IP for PCI Express v11.0.
3. In the MegaWizard Parameter settings, specify the PCIe settings according to your system.
4. Turn on the Select the CvP capable HIP block option.
5. Click Finish.
This section provides additional information about the document and Altera.

**Document Revision History**

The following table lists the revision history for this document.

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 2011</td>
<td>1.0</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>

**How to Contact Altera**

To locate the most up-to-date information about Altera products, refer to the following table.

<table>
<thead>
<tr>
<th>Contact (1)</th>
<th>Contact Method</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical support</td>
<td>Website</td>
<td><a href="http://www.altera.com/support">www.altera.com/support</a></td>
</tr>
<tr>
<td>Technical training</td>
<td>Website</td>
<td><a href="http://www.altera.com/training">www.altera.com/training</a></td>
</tr>
<tr>
<td></td>
<td>Email</td>
<td><a href="mailto:custrain@altera.com">custrain@altera.com</a></td>
</tr>
<tr>
<td>Product literature</td>
<td>Website</td>
<td><a href="http://www.altera.com/literature">www.altera.com/literature</a></td>
</tr>
<tr>
<td>Non-technical support (General)</td>
<td>Email</td>
<td><a href="mailto:nacomp@altera.com">nacomp@altera.com</a></td>
</tr>
<tr>
<td>(Software Licensing)</td>
<td>Email</td>
<td><a href="mailto:authorization@altera.com">authorization@altera.com</a></td>
</tr>
</tbody>
</table>

**Note to Table:**

(1) You can also contact your local Altera sales office or sales representative.

**Typographic Conventions**

The following table lists the typographic conventions this document uses.

<table>
<thead>
<tr>
<th>Visual Cue</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold Type with Initial Capital Letters</strong></td>
<td>Indicate command names, dialog box titles, dialog box options, and other GUI labels. For example, Save As dialog box. For GUI elements, capitalization matches the GUI.</td>
</tr>
<tr>
<td>bold type</td>
<td>Indicates directory names, project names, disk drive names, file names, file name extensions, software utility names, and GUI labels. For example, \qdesigns directory, D: drive, and chiptrip.gdf file.</td>
</tr>
<tr>
<td><strong>Italic Type with Initial Capital Letters</strong></td>
<td>Indicate document titles. For example, Stratix IV Design Guidelines.</td>
</tr>
<tr>
<td>italic type</td>
<td>Indicates variables. For example, \n + 1. Variable names are enclosed in angle brackets (&lt;&gt;). For example, &lt;file name&gt; and &lt;project name&gt;.pof file.</td>
</tr>
<tr>
<td>Initial Capital Letters</td>
<td>Indicate keyboard keys and menu names. For example, the Delete key and the Options menu.</td>
</tr>
<tr>
<td>Visual Cue</td>
<td>Meaning</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>“Subheading Title”</td>
<td>Quotation marks indicate references to sections within a document and titles of Quartus II Help topics. For example, “Typographic Conventions.”</td>
</tr>
<tr>
<td>Courier type</td>
<td>Indicates signal, port, register, bit, block, and primitive names. For example, <code>data1</code>, <code>tdi</code>, and <code>input</code>. The suffix <code>n</code> denotes an active-low signal. For example, <code>resetn</code>. Indicates command line commands and anything that must be typed exactly as it appears. For example, <code>c:\qdesigns\tutorial\chiptrip.gdf</code>. Also indicates sections of an actual file, such as a Report File, references to parts of files (for example, the AHDL keyword <code>SUBDESIGN</code>), and logic function names (for example, <code>TRI</code>).</td>
</tr>
<tr>
<td>↗</td>
<td>An angled arrow instructs you to press the Enter key.</td>
</tr>
<tr>
<td>1., 2., 3., and a., b., c., and so on</td>
<td>Numbered steps indicate a list of items when the sequence of the items is important, such as the steps listed in a procedure.</td>
</tr>
<tr>
<td>■ ■ ■</td>
<td>Bullets indicate a list of items when the sequence of the items is not important.</td>
</tr>
<tr>
<td>⌨</td>
<td>The hand points to information that requires special attention.</td>
</tr>
<tr>
<td>➥</td>
<td>A question mark directs you to a software help system with related information.</td>
</tr>
<tr>
<td>➥</td>
<td>The feet direct you to another document or website with related information.</td>
</tr>
<tr>
<td>!</td>
<td>A caution calls attention to a condition or possible situation that can damage or destroy the product or your work.</td>
</tr>
<tr>
<td>!</td>
<td>A warning calls attention to a condition or possible situation that can cause you injury.</td>
</tr>
<tr>
<td>💌</td>
<td>The envelope links to the Email Subscription Management Center page of the Altera website, where you can sign up to receive update notifications for Altera documents.</td>
</tr>
</tbody>
</table>