# Contents

Overview .................................................................................................................................................. 3  
Figure 1: Key Components in a CvP Design .......................................................................................... 3  
Theory of operation ............................................................................................................................... 4  
Software requirements ............................................................................................................................ 5  
Hardware setup ..................................................................................................................................... 5  
Reference Design Walkthrough ............................................................................................................ 6  
Figure 2: Configuration via Protocol for Arria 10 FPGA Development Kit Flow ................................. 6  
Figure 3: MSEL/DIP switch SWS top view for Active Serial x 4 mode .............................................. 6  
Figure 4: This figure shows the options that you specified .................................................................. 7  
Figure 5: This figure shows the options to turn on CvP mode ............................................................... 8  
Figure 6: This figure shows the columns for Device Identification Registers ...................................... 9  
Figure 7: This figure shows the option for System Settings ................................................................. 9  
Figure 8: This figure shows the option to create CvP files .................................................................. 10  
Figure 9: Transcript from quartus_cvp command .............................................................................. 11  
Revision History .................................................................................................................................... 12
Overview

This reference design is targeted on Arria 10 GX FPGA Development Kit. It demonstrates on the usage of CvP (Configuration via Protocol) initialization mode in Windows system. CvP initialization mode involve interactions between a PCI Express host, the FPGA configuration control block, the Arria 10 Hard IP PCI Express IP core, and CRAM in FPGA.

Control block and FPGA CRAM is part of hidden logic, it is hidden and user will not be able to access them. The Hard IP for PCI Express architecture has an option to configure the FPGA and initialize the PCI Express link, which called as CvP mode. When user enable the CvP mode, the I/O bitstream and PCI Express link are configured first into the FPGA, allows the PCI Express link to reach L0 state and begin operation independently, before the rest of core is configured. The configuration method that mentioned is called as CvP Initialization mode. Initialization refers to the initial fabric configuration image loaded in the FPGA fabric after power up.

Figure 1: Key Components in a CvP Design
Theory of operation
In this reference design, the two IPs that instantiated in the reference design are “Example : Arria 10 Application for Avalon-Streaming Hard IP for PCI Express” and “Arria 10 Hard IP for PCI Express”. The reference design performs basic testing of the Application Layer logic that interfaces to the Hard IP for PCI Express. It includes the following components:

- DUT – A Gen 1 x 1 Endpoint. User can select the data rate, number of lanes, and Endpoint or Root Port mode.
- APPS – This Root Port BFM configures the DUT and drives read write TLPs (Transaction Layer Packets) to test DUT functionality.

The architecture of this reference design is derived from Arria 10 Avalon-ST Interface for PCIe Solutions User Guide. The link to the user guide is

Software requirements
In order to run the reference design in Windows system, the following software is needed:

- Altera Complete Design Suite (ACDS). Refer to the version indicated at the Quartus II version field of this reference design in the Design Store
- Altera Arria 10 GX FPGA Development Kit Installer
- Jungo WinDriver

User can obtain more information from *Altera Configuration via Protocol user guide*.

Hardware setup
In order to run the reference design, the following hardware is needed:

- Arria 10 GX FPGA Development Kit
- USB Blaster
- A DUT PC with PCI Express slot to plug in the Arria 10 GX FPGA development kit
- A host PC running the ACDS software to program the periphery image
Reference Design Walkthrough

Figure 2: Configuration via Protocol for Arria 10 FPGA Development Kit Flow

Modify MSEL on Arria 10 Dev-Kit

- On the FPGA Development Kit, set the MSEL/DIP switch labeled SW5 setting to Active Serial mode, which MSEL [2:0] = 3'b011. From top to bottom, the sequence is left, left, right as shown in Figure 4 below.

Figure 3: MSEL/DIP switch SW5 top view for Active Serial x 4 mode

![Figure 3: MSEL/DIP switch SW5 top view for Active Serial x 4 mode](image)

Download reference design from Design Store

- Download the reference design in *.par format and open the New Project Wizard (File menu) in Quartus II software
- Use the Design Template Installation dialog box to install the template
- Compile the design by select Start compilation (Processing menu). A successful compilation creates a *.sof file accordingly
Setting up CvP parameters in Device and Pin Options GUI

- On the Quartus II Assignments menu, select **Device**, and then click **Device and Pin Options**.
- Under Category select **CvP Settings**, and specify the following settings as per Figure 4.
- By enable CvP_CONFDONE pin, this pin will indicate when the core is fully programmed in CvP mode.

Figure 4: This figure shows the options that you specified
Hard IP setting in Qsys

- On the Tools menu, select Qsys.
- Open .qsys of the project.
- On the System Contents tab, right-click PCIe IP and select Edit.
- Under IP Settings → Configuration, Debug and Extension Options, turn on Enable Configuration via Protocol (CvP) as shown in the Figure 5.
- Under IP Settings → Device Identification Registers, user can key in the Vendor ID and Device ID accordingly and both of this ID will be use during core image configuration using quartus_cvp (Figure 6)
- Under IP Settings → System Settings, user can select the data rate, number of lanes, and Endpoint or Root Port mode (Figure 7)

Figure 5: This figure shows the options to turn on CvP mode
**Figure 6**: This figure shows the columns for Device Identification Registers

<table>
<thead>
<tr>
<th>IP Settings</th>
<th>Example Designs</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCI Express / PCI Capabilities</td>
<td>Avalon-ST Settings</td>
</tr>
<tr>
<td>System Settings</td>
<td>Base Address Registers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Physical Function</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor ID:</td>
<td>0x00001172</td>
</tr>
<tr>
<td>Device ID:</td>
<td>0x0000e001</td>
</tr>
<tr>
<td>Revision ID:</td>
<td>0x00000001</td>
</tr>
<tr>
<td>Class code:</td>
<td>0x00000000</td>
</tr>
<tr>
<td>Subsystem Vendor ID:</td>
<td>0x00004e30</td>
</tr>
<tr>
<td>Subsystem Device ID:</td>
<td>0x00004efc</td>
</tr>
</tbody>
</table>

**Figure 7**: This figure shows the option for System Settings

<table>
<thead>
<tr>
<th>IP Settings</th>
<th>Example Designs</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCI Express / PCI Capabilities</td>
<td>Avalon-ST Settings</td>
</tr>
<tr>
<td>System Settings</td>
<td>Base Address Registers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Application interface type:</th>
<th>Avalon-ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>HP mode:</td>
<td>Gen1×1 Interface: 64-bit, 125 MHz</td>
</tr>
<tr>
<td>Port type:</td>
<td>Native endpoint</td>
</tr>
<tr>
<td>RX buffer credit allocation for received requests vs completions:</td>
<td>Low</td>
</tr>
<tr>
<td>RX buffer completion credits:</td>
<td>Header 195 Data 773</td>
</tr>
</tbody>
</table>
Splitting the SOF file

- Open the **Convert Programming File** (File menu) window to create CvP files. Turn on the **Create CvP files** (Generate *.periph.jic and *.core.rbf) parameter in the Output Programming Files section. Click **Generate** to create *.periph.jic and *.core.rbf respectively as shown in Figure 6.

**Figure 8: This figure shows the option to create CvP files**

![Figure 8: This figure shows the option to create CvP files](image-url)
Configure CvP image

- Program the periphery image (*.periph.jic) into EPCQ flash on the FPGA development kit
- Plug in the FPGA development kit to the PCI Express slot of the DUT PC and power it on
- User can use any system software driver to verify the PCI link status
- Once the PCIE link verified, open a **DOS command** window
- Change to appropriate Quartus bin install directory
  (C:\altera\<quartus_version>\quartus\<bin>)
- Type the following command to program the core image accordingly:
  - `quartus_cvp -vid=1172 -did=e001 <path>/*.core.rbf`
- The figure below illustrates an example that shows the results of a successful CvP programming

Figure 9: Transcript from quartus_cvp command

![CvP Programming Transcript](image-url)
## Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016 January 6</td>
<td>1.0</td>
<td>Initial release</td>
</tr>
<tr>
<td>2016 March 9</td>
<td>1.1</td>
<td>Replaced the Quick Start Guide with Reference Design Walk through</td>
</tr>
</tbody>
</table>