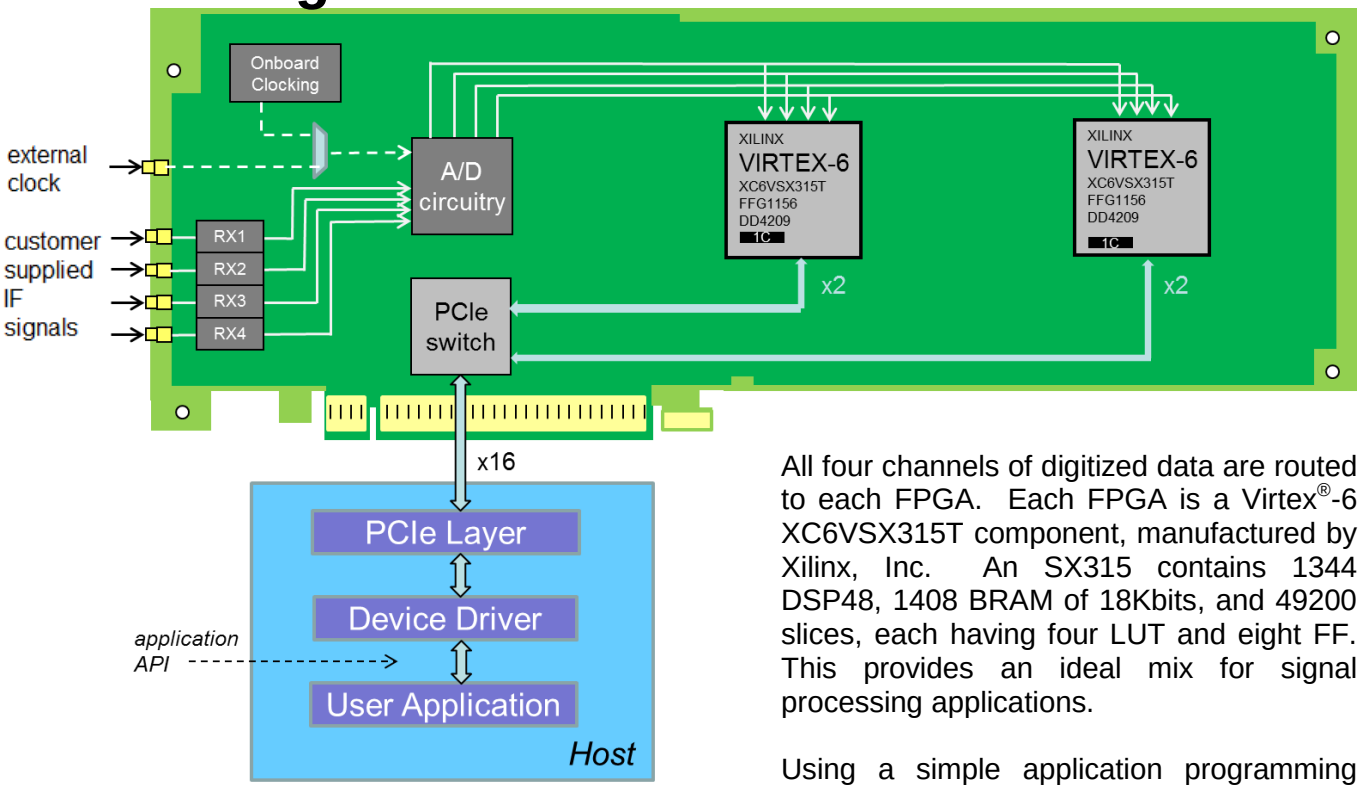
Programmable Virtex® Processor Card



The PVP-642 card combines a multi-channel digital receiver with FPGA-based signal processing elements, on a standard PCI Express® (PCIe®) card. When installed in a customer's host system, such as a rack-mount server or other PCIe host, a PVP-642 provides a complete IF-to-DMA path for receiving wireless signals, processing them in an FPGA, and sending results to host CPU(s) via PCI Express.

After external equipment mixes radio signals down to an appropriate intermediate frequency (IF), the signals are filtered and then input to the PVP card. The PVP-642 has four input channels, which are sampled to 12-bit resolution using a programmable sampling clock between 100 and 250 megasamples per second. Onboard sample clock generation is provided, or customers may supply their own clock via the external clock input port.

All four channels of digitized data are routed to each FPGA. Each FPGA is a Virtex®-6 XC6VSX315T component, manufactured by Xilinx, Inc. An SX315 contains 1344 DSP48, 1408 BRAM of 18Kbits, and 49200 slices, each having four LUT and eight FF. This provides an ideal mix for signal processing applications.

Using a simple application programming interface (API), customers can access the sampled radio signals, and write VHDL or Verilog code to process the signals. Results can be forwarded to host CPUs using the direct memory access API. Each FPGA contains 1024 DMA write engines, for pushing data to the host, and 1024 DMA read engines, for pulling data from the host. On the host side, the job of interfacing to each FPGA is handled by a Linux device driver supplied with the card. Convenient access to the device drivers is provided by a set of C routines called the application API.

In summary, the PVP-642 provides four digital receivers supplying two FPGA processing elements on an industry-standard PCI Express card. Take full advantage of these features, and jump start your signal processing project today!

 Contact: sales@starcommva.com (888)899-3555

PVP-642 Product Specifications

Made in the U.S.A.

Digital Receiver

Number of Channels	4
Resolution	12 bits
Sampling Frequency	100 to 250 MHz
Receiver Bandwidth	50 to 250 MHz
Impedance	50 Ω
Connector Type	MMCX jack (e.g. Amphenol 908-24100)

Clocking

Sample Rate	Programmable
Sample Rate Range	100 to 250 Msps
Internal Reference Clock	±2.5 ppm frequency stability
External Reference Clock	+10 dBm sine wave at desired sampling frequency
External Clock Connector	MMCX jack (e.g. Amphenol 908-24100)

Processing

Operating System	Red Hat Enterprise Linux (RHEL) release 6.2
Radio Data API	VHDL-93
Host Data API	VHDL-93
Linux Device Driver	C, using the GNU compiler (gcc-4.4.6-3.el6.x86 64)
Application API	C, using the GNU compiler (gcc-4.4.6-3.el6.x86 64)
FPGA Development Tool	Xilinx ISE™ version 13.3
Development Interface	JTAG using Xilinx Platform Cable USB (Model DLC9G)
Development Connector	Molex 87832-1420 (standard Xilinx JTAG connector)
Firmware Storage	Xilinx Platform Flash XL (1 per FPGA, 128 Mb each)
Number of FPGAs	2 (Virtex-6 XC6VSX315T)

Host Interface

Interface Type	PCI Express version 1.1 (Gen1)
Signaling Rate	2.5 Gbit/sec per lane
Number of Active Lanes	1 to 16 lanes per card, 1 to 2 lanes per FPGA
PCIe Connector	x16 standard PCIe card edge connector
Configuration Registers	PCI™ Type 0 (Endpoint) Configuration Space
Data Transfer	DMA
Number of DMA Engines	1024 per FPGA

Electro-Mechanical

Card Size (exact)	PCIe standard height, full length, x16 graphics add-in card
Card Size (approx.)	12.3 by 4.4 by 0.8 inches
Front Panel Bracket	Included (aka I/O bracket)
Power Consumption	Processing dependent; 225 Watts, maximum
Power Connectors	2 PEG-6 standard PCIe connectors (e.g. Molex 39-28-8060)
Power Connector Location	Factory option (fixed or floating)

With the purchase of a PVP-642 card and software development kit license, customers receive the following items: one (1) PVP-642 card with installed front panel, four (4) coaxial cables for MMCX-to-SMA adaption, an FPGA software development kit using VHDL, and a C software development kit including Linux device drivers and API routines for use on the host system.