



# Aurora **FDK** for Zebra **Rapixo Pro** **>>>**

Harness the full power and flexibility of FPGAs  
for image processing

## Overview

### Image processing offload and acceleration

Aurora - FPGA Development Kit (FDK) provides a library of components that enable development of custom field-programmable gate array (FPGA) configurations for [Zebra Rapixo CXP](#) frame grabbers, based on Xilinx Kintex® UltraScale™ devices, and the [Zebra Rapixo CL Pro](#) frame grabbers, based on Xilinx Kintex™-7 devices. In both instances, the Aurora FDK is used in combination with Xilinx Vivado® Design suite to create FPGA configurations that offload and accelerate image processing functions from the host system.

### Focus on custom image processing functions

Developers with software backgrounds can use the Vivado High-Level Synthesis (HLS) tool to write custom image processing functions as FPGA design components using the C or C++ languages. With the Aurora FDK, developers can focus on creating custom FPGA design components vital to their application rather than the peripheral logic.

### Quick assembly of FPGA design components

Custom- and ready-made Aurora FPGA design components are graphically combined within the Xilinx Vivado IP Integrator tool to easily create custom FPGA configurations. In addition, Xilinx provides over 50 image-processing functions within their library with their source code. These can be compiled using Vivado HLS into building blocks compatible with the components provided by the Aurora FDK.

### Xilinx Vivado HLS

The Vivado HLS tool is designed for software application developers and FPGA designers seeking a more direct path to FPGA hardware. The Vivado HLS compiler is a high-level synthesis tool that lets developers compile C/C++ algorithms directly into optimized logic, ready for use with FPGA devices found on both models of the Zebra Rapixo Pro. The Vivado HLS tools enable highly iterative, software-oriented design methods for quick development of FPGA hardware modules from C/C++ source code. Several example functions are provided with the FDK; those can be used as-is or as a model for creating custom processing units.

### Aurora FDK at a glance

**Build custom FPGA configurations** for Zebra Rapixo Pro frame grabbers

**Code custom FPGA design components** in C/C++ with Xilinx Vivado HLS tool

**Accelerate creation of custom FPGA configurations** using Aurora Imaging's library, formerly Matrox Imaging Library (MIL), of ready-made FPGA design components

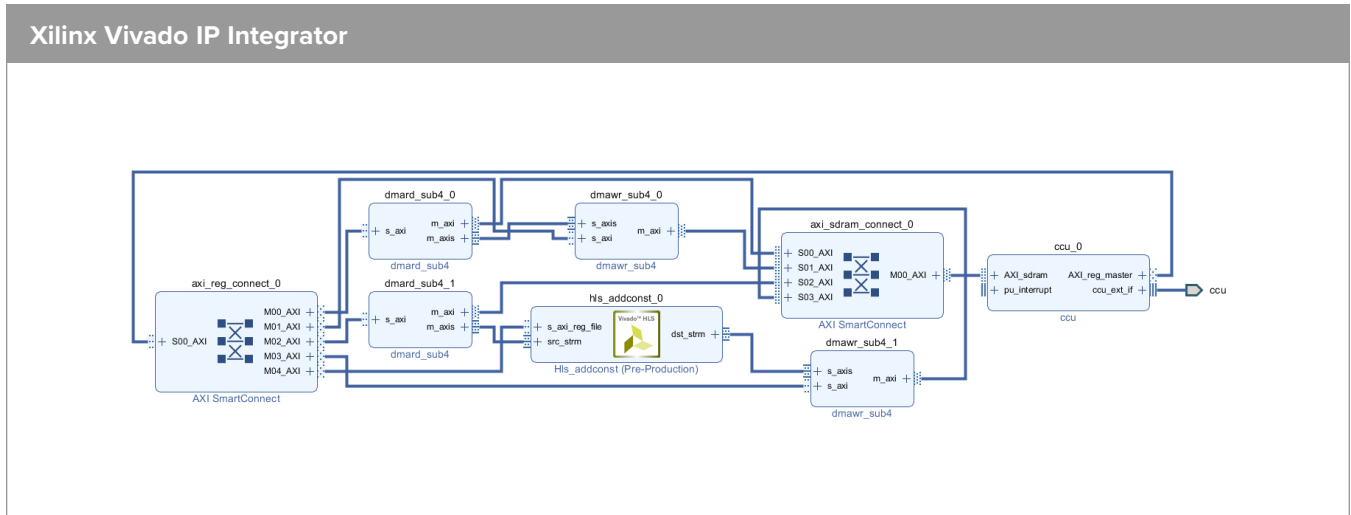
**Leverage more than 50 optimized image-processing functions** from the Xilinx IP library

**Assemble FPGA configurations quickly** with the graphical Xilinx Vivado IP Integrator

# Overview (cont.)

## Xilinx Vivado IP Integrator

Custom- and ready-made Aurora FPGA design components are graphically combined within the Xilinx Vivado IP Integrator tool to easily create custom FPGA configurations.



## Xilinx Vivado HLS

Developers with software backgrounds can use the Xilinx Vivado HLS tool to write custom image processing functions as FPGA design components using C or C++ languages.

### Xilinx Vivado HLS

```
0  
1 void hls_addconst(fdc_stream& src_strm, fdc_stream& dst_strm, int ctrl, int val, int numpix)  
2 {  
3     #pragma HLS INTERFACE axis port=src_strm  
4     #pragma HLS INTERFACE axis port=dst_strm  
5     #pragma HLS INTERFACE s_axilite port=ctrl bundle=Regfile  
6     #pragma HLS INTERFACE s_axilite port=val bundle=Regfile  
7     #pragma HLS INTERFACE s_axilite port=numpix bundle=Regfile  
8     #pragma HLS INTERFACE s_axilite port=return bundle=Regfile  
9  
10    int i, n;  
11    unsigned char sat, optype, dest;  
12    short dst, valshort;  
13    FDC_STREAM_TYPE srch, dsth;  
14  
15    // Pick up parameters  
16    optype = ctrl & 0x3;  
17    sat = (ctrl >> 2) & 0x1;  
18    dest = (ctrl >> 8) & 0xff;  
19    valshort = (short)val;  
20  
21    // Process one frame  
22    for (i = 0; i < numpix; i += NPIX)  
23    {  
24        #pragma HLS pipeline  
25  
26        // Read N pixels  
27        srch = src_strm.read();  
28  
29        // Process N pixels in parallel  
30        for (n = 0; n < NPIX; n++)  
31        {  
32            if (optype == 0) // sign extend each byte  
33                dst = (char)(FIX8(srch, n));  
34            else  
35                dst = (unsigned char)(PIX(srch, n));  
36  
37            // Add as 16-bit to avoid overflow  
38            dst += valshort;  
39  
40        }  
41  
42        dsth.write(dst);  
43    }  
44 }  
45  
46  
47
```

Vivado HLS Console  
Checking type = 0, saturate = 0... CORRECT  
Checking type = 0, saturate = 1... CORRECT  
Checking type = 1, saturate = 0... CORRECT  
Checking type = 1, saturate = 1... CORRECT  
INFO: [SIN 211-1] C-Sim done with 0 errors.  
INFO: [SIN 211-3] \*\*\*\*\* C-Sim Finish \*\*\*\*\*  
Finished C simulation.

# Specifications

## Supported environment

- 64-bit Microsoft® Windows® 7 and 10

# Ordering Information

Part number	Description
<b>Hardware</b>	
RAPPROFDK	Aurora FDK for Zebra Rapixo Pro. Must be ordered with 6x IMGPROSERV for personalized training and/or support. Requires Xilinx Vivado Design suite.
<b>Additional requirements (sold separately)</b>	
Compatible hardware	<a href="#">Zebra Rapixo CXP</a>
	<a href="#">Zebra Rapixo CL Pro</a>
Compatible software	<a href="#">Aurora Imaging Library</a> or <a href="#">Aurora Imaging Library-Lite</a>
	<a href="#">Xilinx Vivado Design Suite – HLx Editions<sup>1</sup></a>
Operating system support	Windows 7 (64-bit)
	Windows 10 (64-bit)

Endnotes:

1. Purchased from Xilinx Inc.



**NA and Corporate Headquarters**  
 +1 800 423 0442  
[inquiry4@zebra.com](mailto:inquiry4@zebra.com)

**Asia-Pacific Headquarters**  
 +65 6858 0722  
[contact.apac@zebra.com](mailto:contact.apac@zebra.com)

**EMEA Headquarters**  
[zebra.com/locations](http://zebra.com/locations)  
[contact.emea@zebra.com](mailto:contact.emea@zebra.com)

**Latin America Headquarters**  
[zebra.com/locations](http://zebra.com/locations)  
[la.contactme@zebra.com](mailto:la.contactme@zebra.com)