logiADAK-VDF-SDSoC

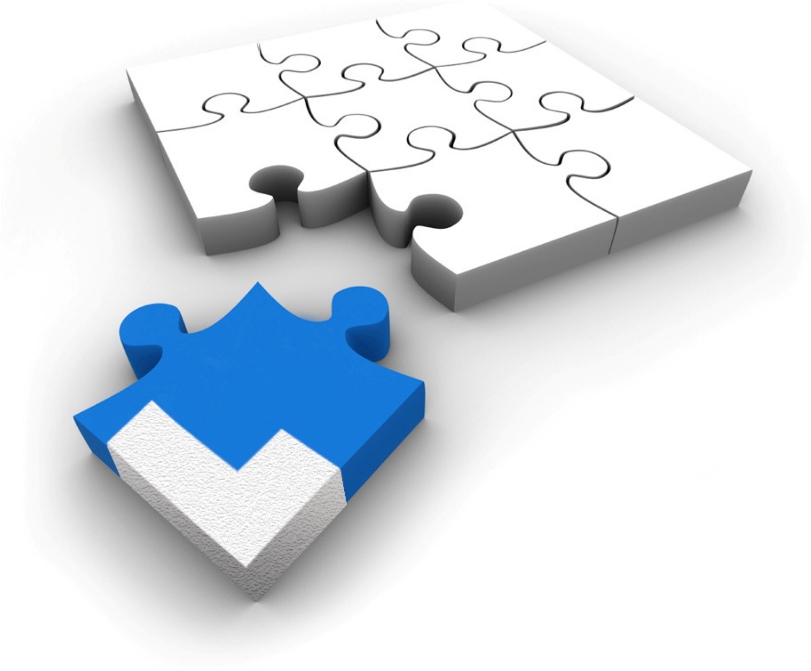
Video Design Framework - Reference Designs for Xylon logiADAK and logiVID-Z Development Kits

Prepared for the Xilinx® SDSoC™ Development Environment

User's Manual

Version: 2.00.a

logiADAK_VDF_SDSoC_v2_00_a.docx



This publication has been carefully checked for accuracy. However, Xylon does not assume any responsibility for the contents or use of any product described herein. Xylon reserves the right to make any changes to product without further notice. Our customers should ensure to take appropriate action so that their use of our products does not infringe upon any patents.

1 ABOUT THE FRAMEWORK

The logiADAK-VDF-SDSoC Video Design Framework enables Xylon logiADAK Automotive Driver Assistance Kit and logiVID-Z Vision Development Kit users to quickly utilize the provided hardware platform for their own development of multi-camera computer vision systems based on the Xilinx® Zynq®-7000 All Programmable SoC.

The framework includes pre-verified logicBRICKS reference designs for video capture from Xylon video cameras and the HDMI video input, and the display output under the Linux operating system running on the Xilinx® Zynq®-7000 All Programmable SoC. Reference designs are prepared for both, hardware-centric Vivado® Design Suite and software-centric SDSoC™ Development Environment. This document describes the reference designs prepared for the Xilinx SDSoC Development Environment. Xilinx Vivado compatible designs are described in the logiADAK-VDF document.

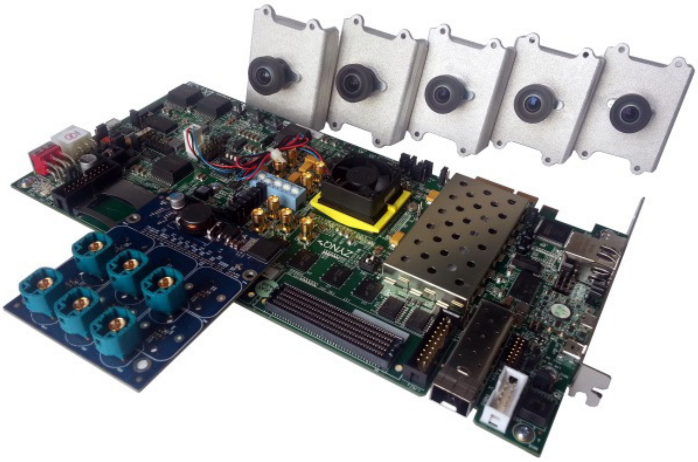


Figure 1: Xylon logiADAK Automotive Driver Assistance Kit

The complete camera-to-display SoC designs, which are compact and use just a fraction of available programmable logic, significantly save the design time. Instead of starting from scratch and having to spend months designing and building a new design framework, logiADAK-VDF-SDSoC design framework users can immediately focus on specific vision-based parts of their next SoC design for ADAS. Supported hardware platforms can be installed on test vehicles (cars, robots...) and used in exhaustive tests, i.e. for testing of the new ADAS developments in the test vehicle and under different road conditions.

The logiADAK-VDF-SDSoC reference designs include Xylon logicBRICKS IP cores and hardware design files prepared for Xilinx SDSoC Development Environment. To provide the SDSoC users the complete embedded C/C++ development experience, the supplied SDSoC reference designs include the Sobel video filter example. This example shows kit users how to integrate their own vision processing logic between video input and video output IP cores, and how to implement it in software or in programmable logic. The Linux OS and software drivers for logicBRICKS IP cores enable software developers to efficiently work with the framework, without knowing the hardware implementation details.

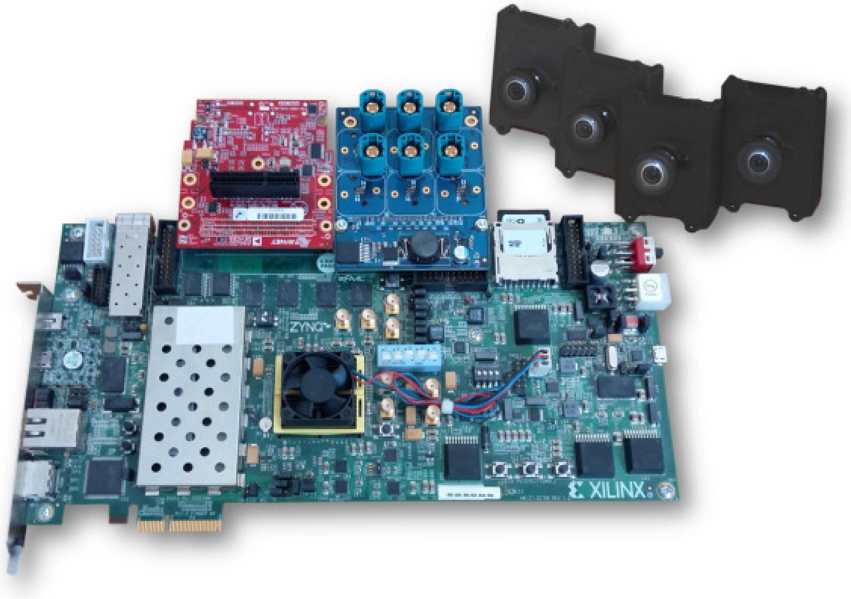


Figure 2: Xylon logiVID-Z Vision Development Kit

The logiADAK-VDF-SDSoC Video Design Framework is functionally equal to Xylon's logiADAK-VDF Video Design Framework reference design, which is implemented fully in the Xilinx Vivado® Design Suite.

1.1 Programmable Logic Utilization

The logiADAK-VDF-SDSoC reference designs utilize just small fractions (Table 3) of available programmable logic in the Xilinx Zynq-7000 AP SoC XC7Z045 device. Free resources can be utilized by users who can also alter the pre-defined logicBRICKS configurations and change the programmable logic utilization.

Table 1: CAM-HDMI Reference Design Programmable Logic Utilization

Family (Device)	F (MHz)			LCs	Slices[1] (FFs/ LUTs)	IOB[2]	CMT	BRAM Tile (RAMB36/RAMB18)	MULT/ DSP48/E	DCM / CMT	GTx	Design Tools
	mclk[4]	vclk[4]	rclk									
Zynq-7000[3] (XC7Z045-2)	(150/100)	(80/100)	100	14803	2313 (7464/ 5835)	60	0	11 (5/12)	7	0	0	Vivado 2016.2

Table 2: FOUR-CAM Reference Design Programmable Logic Utilization

Family (Device)	F (MHz)			LCs	Slices[1] (FFs/ LUTs)	IOB[2]	CMT	BRAM Tile (RAMB36/RAMB18)	MULT/ DSP48/E	DCM / CMT	GTx	Design Tools
	mclk[4]	vclk[4]	rclk									
Zynq-7000[3] (XC7Z045-2)	(150/100)	(80/100)	100	46048	7195 (23754/ 18551)	49	0	65 (17/96)	28	0	0	Vivado 2016.2

Notes:

1) Assuming the following configuration: ITU656, RGB565 output, 32-bit AXI4-Lite register interface, 64-bit AXI4 memory interface with max. burst size of 64 words, scaling in both directions with multipliers (DSP48s), output stride set to 1024 pixels
2) Assuming only video inputs are routed off-chip, register and memory interfaces are connected internally
3) Only burst size of 16 words is supported on HP ports in the Xilinx Zynq-7000 SoC
4) logiCVC/logiWIN clock frequencies

Table 3: Free Programmable Logic Resources

	Available in XC7Z045	Used Resources	
		CAM-HDMI	FOUR-CAM
Logic Cells (LCs)	350K	~ 4%	~ 13%
Flip Flops (FFs)	437,200	~ 2%	~ 5%
Look-Up Tables (LUTs)	218,600	~ 3%	~ 8%
Block RAM (36 kB BRAM)	545	~ 3%	~ 20%
DSP slices (MULT/DSP)	900	~ 1%	~ 3%

1.2 About Xilinx SDSoC Development Environment

The Xilinx SDSoC development environment is a member of the Xilinx SDx™ family that provides a greatly simplified ASSP-like C/C++ programming experience including an easy to use Eclipse IDE and a comprehensive design environment for heterogeneous Zynq® All Programmable SoC and MPSoC deployment.

Complete with the industry's first C/C++ full-system optimizing compiler, SDSoC delivers system level profiling, automated software acceleration in programmable logic, automated system connectivity generation, and libraries to speed programming.

To access the capabilities of SDSoC, please visit www.xilinx.com/sdsoc

Xylon is an SDSoC development environment-qualified Xilinx Alliance Member and offers logicBRICKS IP cores, complete Xilinx All Programmable based solutions and design services.

1.3 Hardware Requirements

The logiADAK Automotive Driver Assistance Kit (Figure 1) includes the following hardware, which is utilized by reference designs provided in the logiADAK-VDF-SDSoC design framework:

- 1x Xilinx Zynq-7000 SoC ZC706 Development Kit[1] with XC7Z045 FFG900 -2 AP SoC
- 1x Xylon FMC add-on board for up to 6 camera connections
- 5x Xylon Cameras
- 1x SD card
- 5x Rosenberger cables

Learn more about the logiADAK kit: http://www.logicbricks.com/Products/logiADAK.aspx

The logiVID-Z Vision Development Kit (Figure 2) includes the following hardware, which is utilized by reference designs provided in the logiADAK-VDF design framework:

- 1x Xilinx Zynq-7000 SoC ZC706 Development Kit[1] with XC7Z045 FFG900 -2 AP SoC
- 1x Xylon FMC add-on board for up to 6 camera connections
- 1x Avnet HDMI Input/Output FMC Module
- 4x Xylon Cameras
- 1x SD card
- 4x Rosenberger cables
- Power supply

Learn more about the logiVID-Z kit: http://www.logicbricks.com/Products/logiVID-Z.aspx

[1] – OEM kit version without the Xilinx Vivado Design Suite seat

1.3.1 HDMI Input/Output FMC Module

The CAM-HDMI reference design delivered with the logiADAK-VDF-SDSoC requires the HDMI Input/Output FMC module (Figure 3), which is available with the logiVID-Z kit and not included with the Xylon logiADAK Kit. In order to use this reference design, the logiADAK kit users needs to purchase the HDMI Input/Output FMC module from Avnet (Part Number: AES-FMC-HDMI-CAM-G).

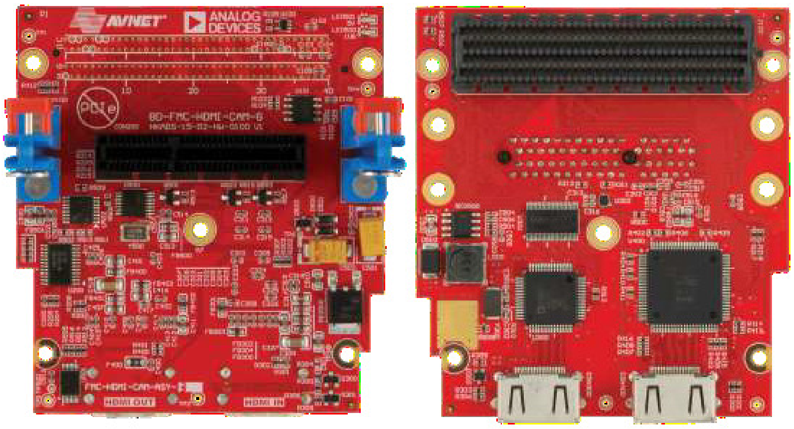


Figure 3: Avnet HDMI Input/Output FMC Module

1.3.2 Xylon Camera

For transmissions of the high-definition uncompressed video and camera control data, Xylon kits include hardware and software that is completely developed by Xylon: the LVDS-based serial interface, the Xylon video camera and an add-on LVDS FMC receiver board for up to six camera connections to the video processor implemented in the Xilinx Zynq-7000 AP SoC.

Each Xylon video camera (Figure 1) includes the OmniVision OV10635 1-megapixel camera sensor that combines high-definition 1280x800p30 WXGA (HD) video with the color high dynamic range (HDR) functionality, LVDS serializer (transmitter) board, the Sunex DSL219 miniature fish-eye Wide FOV lens and a short cable lead with a connector. The logiADAK kit also includes the 5th camera equipped with the Sunex DSL947 Narrow Field of View (FOV) miniature lens, which is in Xylon ADAS demos used for the forward-looking collision avoidance and in-cabin driver status monitoring.

All camera parts are enclosed in the waterproof aluminium housing. The housing is sealed with rubber gaskets to ensure a weather-proof rating of IP67. Its rugged metal construction provides excellent lens and imager module protection and enables safe and easy test vehicle installations.

1.4 Software Requirements

The logiADAK-VDF-SDSoC reference designs and Xylon logicBRICKS IP cores are fully compatible with Xilinx SDSoC Development Environment 2016.2. Future design releases shall be synchronized with the newest Xilinx development tools.

1.5 Design Deliverables

1.5.1 SDSoC platform

- Two reference designs prepared for Xilinx SDSoC Development Environment

- Contains the pre-built hardware files for faster software development
- Supports Linux applications
- Includes software drivers for included logicBRICKS IP cores
- Xylon evaluation logicBRICKS IP cores:
  - logiCVC-ML Compact Multilayer Video Controller
  - logiWIN Versatile Video Input
  - logiI2C I2C Bus Master Controller
  - logiVLINK Vanilla LVDS Multimedia Data Link Receiver

1.5.2 Software

- Linux user space drivers with driver examples
- Demo application sources
- Bare-metal software drivers for logicBRICKS IP cores

1.5.3 Binaries

- Precompiled SD Card image for the fastest demo startup
  - Camera/HDMI demoes
  - Four Camera demoes

1.6 Reference Designs

The logiADAK-VDF-SDSoC video design framework includes two reference designs:

- FOUR-CAM reference design implements four parallel video inputs from Xylon cameras, and the display output with RGB graphic overlay. All video inputs are stored in the video memory, and by mean of the on-board push buttons, the design user can select each of them for the full screen display output. The design also includes the Sobel filter example to show framework users how to add their own vision processing logic.

- CAM-HDMI reference design implements a single video input, and the display output with the RGB graphic overlay. The video input can be sourced from the attached Xylon video camera, or through the HDMI video input. The HDMI video input is particularly suited for use with PC and playback of prepared test videos, such as the video recordings of road and traffic situations that make test cases for the video ADAS processing. The design displays out a single video source, filtered with the Sobel filter, and automatically switches to the HDMI video input upon detection of the plugged-in HDMI cable. The Sobel example shows kit users how to integrate their own vision processing logic between video input and video output IP cores, and how to implement it in software or in programmable logic.

2 LOGICBRICKS IP CORES

2.1 About logicBRICKS IP Library

Xylon's logicBRICKS IP core library provides IP cores optimized for Xilinx All Programmable FPGA and SoC devices. The logicBRICKS IP cores shorten development time and enable fast design of complex embedded systems based on Xilinx All Programmable devices.

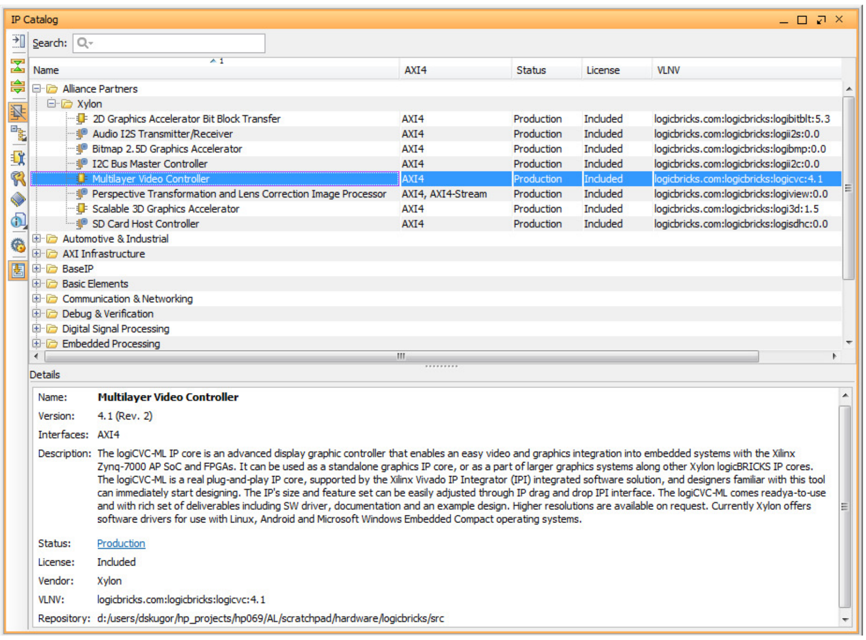
The key features of the logicBRICKS IP cores are:

- logicBRICKS can be used in the same ways as Xilinx IP cores within the Xilinx Vivado Design Suite and require no skills beyond general tools knowledge. IP users can setup feature sets and programmable logic utilization through implementation tools' Graphical User Interface (GUI).

- Each logicBRICKS IP core comes with the extensive documentation, reference design examples and can be evaluated on reference hardware platforms. Xylon provides evaluation logicBRICKS IP cores to enable risk-free evaluation prior to purchase.

- Broad software support – from bare-metal software drivers to standard software drivers for different operating systems (OS). Standard software support allows graphics designers and software developers to use logicBRICKS in a familiar and comfortable way.

- Xylon assures skilled technical support.



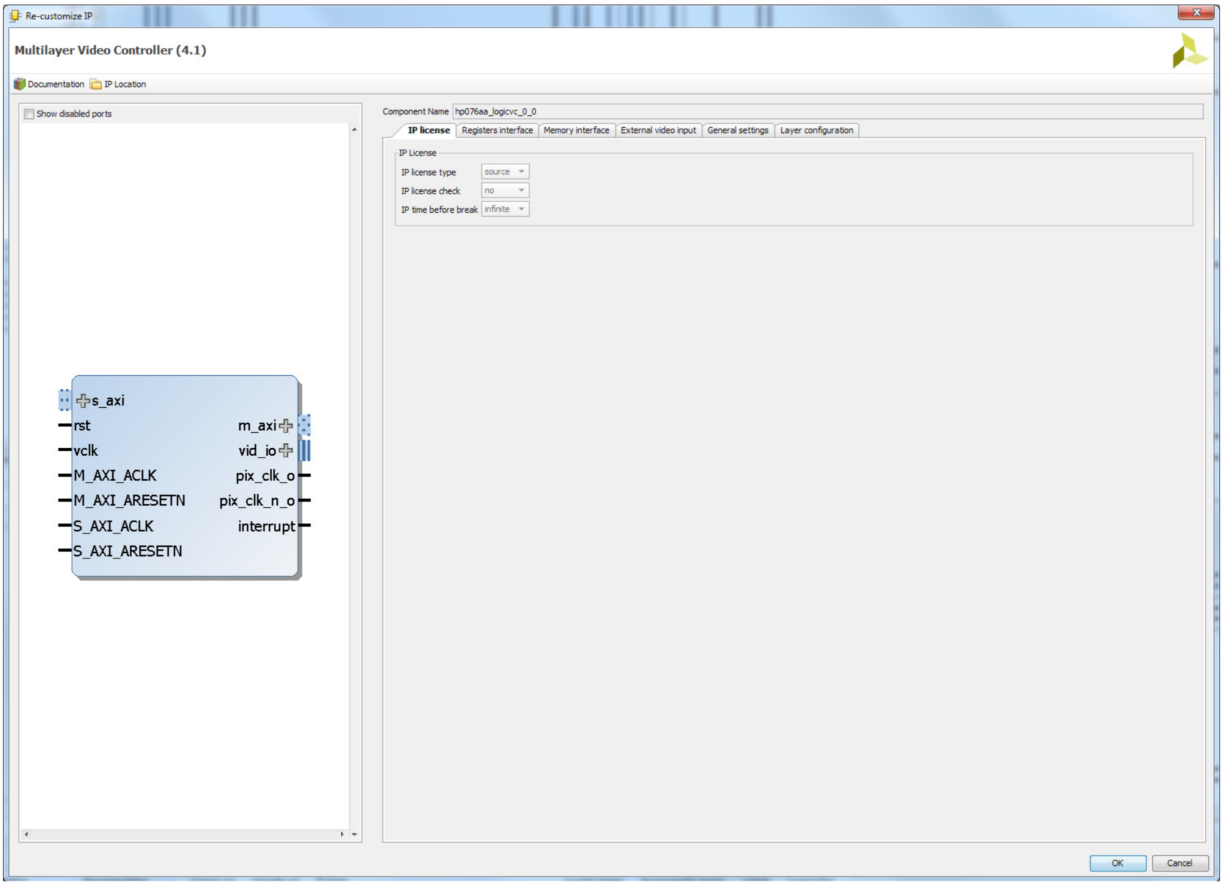
Name	AXI4	Status	License	VLNV
Alliance Partners				
Xylon				
2D Graphics Accelerator Bit Block Transfer	AXI4	Production	Included	logicbricks.com:logicbricks:logibitblt:5.3
Audio I2S Transmitter/Receiver	AXI4	Production	Included	logicbricks.com:logicbricks:logii2s:0.0
Bitmap 2.5D Graphics Accelerator	AXI4	Production	Included	logicbricks.com:logicbricks:logibmp:0.0
I2C Bus Master Controller	AXI4	Production	Included	logicbricks.com:logicbricks:logii2c:0.0
Multilayer Video Controller	AXI4	Production	Included	logicbricks.com:logicbricks:logicvc:4.1
Perspective Transformation and Lens Correction Image Processor	AXI4, AXI4-Stream	Production	Included	logicbricks.com:logicbricks:logiview:0.0
Scalable 3D Graphics Accelerator	AXI4	Production	Included	logicbricks.com:logicbricks:logi3d:1.5
SD Card Host Controller	AXI4	Production	Included	logicbricks.com:logicbricks:logisdhc:0.0
Automotive & Industrial				
AXI Infrastructure				
BaseIP				
Basic Elements				
Communication & Networking				
Debug & Verification				
Digital Signal Processing				
Embedded Processing				

Details

Name: **Multilayer Video Controller**
Version: 4.1 (Rev. 2)
Interfaces: AXI4
Description: The logiCVC-ML IP core is an advanced display graphic controller that enables an easy video and graphics integration into embedded systems with the Xilinx Zynq-7000 AP SoC and FPGAs. It can be used as a standalone graphics IP core, or as a part of larger graphics systems along other Xylon logicBRICKS IP cores. The logiCVC-ML is a real plug-and-play IP core, supported by the Xilinx Vivado IP Integrator (IPI) integrated software solution, and designers familiar with this tool can immediately start designing. The IP's size and feature set can be easily adjusted through IP drag and drop IPI interface. The logiCVC-ML comes readya-to-use and with rich set of deliverables including SW driver, documentation and an example design. Higher resolutions are available on request. Currently Xylon offers software drivers for use with Linux, Android and Microsoft Windows Embedded Compact operating systems.
Status: Production
License: Included
Vendor: Xylon
VLNV: logicbricks.com:logicbricks:logicvc:4.1
Repository: d:/users/dskugor/hp_projects/hp069/AL/scratchpad/hardware/logicbricks/src

Figure 4: logicBRICKS IP Cores Imported into the Vivado IP Catalog

The Figure 4 shows imported logicBRICKS IP cores into Vivado Design Suite, while the Figure 5 shows a typical logicBRICKS IP core's configuration GUI.



Re-customize IP

Multilayer Video Controller (4.1)

☐ Show disabled ports

Component Name: hp076aa_logicvc_0_0

IP License

IP license type: source

IP license check: no

IP time before break: infinite

Figure 5: Example of logicBRICKS IP Configuration GUI

To access logicBRICKS IP cores' user's manuals, double-click on the specific IP core's icon to open the IP core GUI and click on the Documentation icon. Choose either Product guide to open the manual or Change Log to open IP core's change log.

logicBRICKS User's Manuals contain all necessary information about the IP cores' features, architecture, registers, modes of operation, etc.

2.2 Evaluation logicBRICKS IP Cores

Xylon offers free evaluation logicBRICKS IP cores which enable full hardware evaluation:

- Import into the Xilinx Vivado tools (IP Integration)
- IP parameterization through the GUI interface
- Simulation (if Xilinx tools support it)
- Bitstream generation

The logicBRICKS evaluation IP cores are run-time limited and cease to function after some time. Proper operation can be restored by reloading the bitstream. Besides this run-time limitation, there are no other functional differences between the evaluation and fully licensed logicBRICKS IP cores.

Evaluation logicBRICKS IP cores are distributed as parts of the Xylon reference designs: http://www.logicbricks.com/logicBRICKS/Reference-logicBRICKS-Design.aspx.

Specific IP cores can be downloaded from Xylon's web shop: http://www.logicbricks.com/Products/IP-Cores.aspx.

2.3 logicBRICKS IP Cores Used in This Design

2.3.1 logiCVC-ML Compact Multilayer Video Controller

The logiCVC-ML IP core is an advanced display graphics controller for LCD and CRT displays, which enables an easy video and graphics integration into embedded systems with Xilinx Zynq-7000 All Programmable SoC and FPGAs.

This IP core is the cornerstone of all 2D and 3D GPUs. Though its main function is to provide flexible display control, it also includes hardware acceleration functions: three types of alpha blending, panning, buffering of multiple frames, etc.

- Supports all Xilinx FPGA families
- Supports LCD and CRT displays (easily tailored for special display types)
- 64x1 to 2048x2048 display resolutions
- Available SW drivers for: Linux, Android, QNX and Microsoft Windows Embedded Compact OS
- Support for higher display resolutions available on request
- Supports up to 5 layers; the last one configurable as a background layer
- Configurable layers' size, position and offset
- Alpha blending and Color keyed transparency
- Pixel, layer, or Color Lookup Table (CLUT) alpha blending mode can be independently set for each layer
- Packed pixel layer memory organization:
  - RGB – 8bpp, 8bpp using CLUT, 16bpp Hi-color RGB 565 and True-color 24bpp
  - YCbCr – 16bpp (4:2:2) and 24bpp (4:4:4)
- Configurable CoreConnect™ PLBv4.6, Xylon XMB or ARM® AMBA® AXI4 memory interface data width (32, 64 or 128)
- Programmable layer memory base address and stride
- Simple programming due to small number of control registers
- Support for multiple output formats:
  - Parallel display data bus (RGB): 12x2-bit, 15-bit, 16-bit, 18-bit or 24-bit
  - YCbCr 4:4:4 or 4:2:2 output format
  - Digital Video ITU-656: PAL and NTSC
  - LVDS output format: 3 or 4 data pairs plus clock
  - Camera link output format: 4 data pairs plus clock

  - DVI output format
- Supports synchronization to external parallel input
- HW cursors
- Versatile and programmable sync signals timing
- Double/triple buffering enables flicker-free reproduction
- Display power-on sequencing control signals
- Parametrical VHDL design that allows tuning of slice consumption and features set
- Prepared for Xilinx Vivado tools

More info: http://www.logicbricks.com/Products/logiCVC-ML.aspx
Datasheet: http://www.logicbricks.com/Documentation/Datasheets/IP/logiCVC-ML_hds.pdf

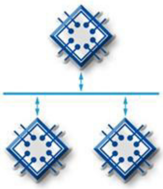
2.3.2 logiWIN Versatile Video Input

The logiWIN IP core enables easy implementation of video frame grabbers. Input video can be decoded, real-time scaled, de-interlaced, cropped, anti-aliased, positioned on the screen... Multiple logiWIN instances enable processing of multiple video inputs by a single Xilinx device.

- Supports versatile digital video input formats:
  - ITU656 and ITU1120 (PAL and NTSC)
  - RGB
  - YUV 4:2:2
- Maximum input and output resolutions are 2048 x 2048 pixels
- Built-in YcrCb to RGB converter, YUV to RGB converter and RGB to YcrCb converter
- Embedded image color enhancements: contrast, saturation, brightness and hue for ITU and YUV separately
- Real-time video scale-up (zoom in) up to 64x
- Real-time video scale-down (zoom out) down to 16 times
  - Lossless scaling down to 2x, or 4x in cascade scaling mode
- Supports video input cropping and smooth image positioning
- Configurable register interface: CoreConnect™ OPB or PLBv46 and ARM® AMBA® AXI4-Lite
- ARM® AMBA® AXI4 and AXI4-Lite bus compliant
- Compressed stencil buffer in BRAM (mask over output buffer)
- Supports pixel alpha blending
- Provides "Bob" and "Weave" de-interlacing algorithms
- Supported big and little Endianness memory layout
- Double or triple buffering for flicker-free video
- Prepared for Xilinx Vivado tools

More info: https://www.logicbricks.com/Products/logiWIN.aspx
Datasheet: https://www.logicbricks.com/Documentation/Datasheets/IP/logiWIN_hds.pdf

2.3.3 logiI2C I2C Bus Master



The logiI2C is Xylon logicBRICKS IP core compatible with the I2C serial bus interface standard. The IP core supports single master I2C communications and enables bug-free data transfers.

- Master I2C serial bus controller
- Supports single master operation
- ARM® AMBA® AXI4-Lite bus compliant
- 16 locations deep TX and RX data FIFO
- Supported transmission speeds:
  - Normal – 100 Kbps
  - Fast - 400 Kbps
  - High speed – 3.5 Mbps
- Prepared for Xilinx Vivado tools

More info: https://www.logicbricks.com/Products/logiWIN.aspx
Datasheet: https://www.logicbricks.com/Documentation/Datasheets/IP/logiWIN_hds.pdf

2.3.4 logiVLINK Vanilla Multimedia Data Link Receiver

The logiADAK and the logiVID-Z kits integrate proprietary logicBRICKS LVDS Camera Interface serial link for joint transmissions of high-speed video, audio and control data. Xylon ADAS solutions have been already successfully used in projects incorporating other video links (examples: TI FPD-Link III, Ethernet…) for video transmissions between different video cameras and the Zynq-7000 AP SoC or FPGA device. More info: info@logicbricks.com

2.4 logicBRICKS IP Cores for Video Processing

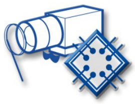
Xylon offers several logicBRICKS IP cores for video processing on Xilinx FPGA and SoC programmable devices:

logiVIEW Perspective Transformation and Lens Correction Image Processor

Removes fish-eye lens distortions and executes programmable transformations on multiple video inputs in a real time. Programmable homographic transformation enables: cropping, resizing, rotating, transiting and arbitrary combinations. Arbitrary non-homographic transformations are supported by programmable Memory Look-Up Tables (MLUT).

More info: http://www.logicbricks.com/Products/logiVIEW.aspx
Datasheet: http://www.logicbricks.com/Documentation/Datasheets/IP/logiVIEW_hds.pdf

logiISP Image Signal Processing (ISP) Pipeline



The logiISP Image Signal Processing Pipeline IP core is a full high-definition ISP pipeline designed for digital processing and image quality enhancements of an input video stream in Smarter Vision embedded designs based on Xilinx Zynq-7000 All Programmable SoC and 7 Series FPGA devices.

More info: http://www.logicbricks.com/Products/logiISP.aspx
Datasheet: http://www.logicbricks.com/Documentation/Datasheets/IP/logiISP_hds.pdf

logiHDR High Dynamic Range (HDR) Pipeline

Ultra High Definition (UHD, including 4K2Kp60) HDR pipeline for camera image quality enhancements. Enables extraction of the maximum detail from high-contrast scenes, i.e. scenes with objects highlighted by a direct sunlight and objects placed in extreme shades.

More info: http://www.logicbricks.com/Products/logiHDR.aspx
Datasheet: http://www.logicbricks.com/Documentation/Datasheets/IP/logiHDR_hds.pdf

3 GET AND INSTALL THE DESIGN FRAMEWORK

To download the evaluation version and to purchase the design framework, please visit our online catalog: http://www.logicbricks.com/Products/logiADAK-VDF.aspx

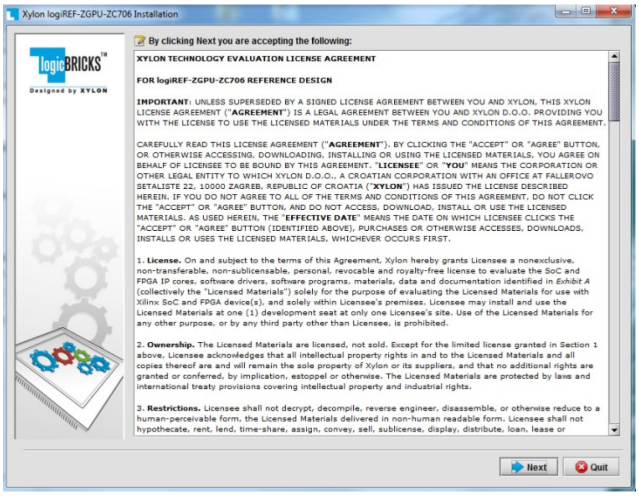
3.1 Installation Process

Installation process is quick and easy. The logiADAK-VDF-SDSoC framework can be downloaded as a cross-platform Java JAR self-extracting installer. Please make sure that you have a copy of the JRE (Java Runtime Environment) version 6 or higher on your system to run Java applications and applets. Double-click on the installer's icon to run the installation.

At the beginning, you will be requested to accept the design framework license – Figure 6. For installation in Linux OS, please follow instructions:

http://www.logicbricks.com/logicBRICKS/Reference-logicBRICKS-Design/Xylon-Reference-Designs-Linux-Installation.aspx.

If you agree with the conditions from the Xylon license, click NEXT and select the installation path for your logicBRICKS reference design (Figure 7). The installation process takes several minutes. It generates the folder structure described in the paragraph 3.1.1 Folder Structure.



By clicking Next you are accepting the following:

XYLON TECHNOLOGY EVALUATION LICENSE AGREEMENT

FOR logiREF-ZGPU-ZC706 REFERENCE DESIGN

IMPORTANT: UNLESS SUPERSEDED BY A SIGNED LICENSE AGREEMENT BETWEEN YOU AND XYLON, THIS XYLON LICENSE AGREEMENT ("**AGREEMENT**") IS A LEGAL AGREEMENT BETWEEN YOU AND XYLON D.O.O. PROVIDING YOU WITH THE LICENSE TO USE THE LICENSED MATERIALS UNDER THE TERMS AND CONDITIONS OF THIS AGREEMENT.

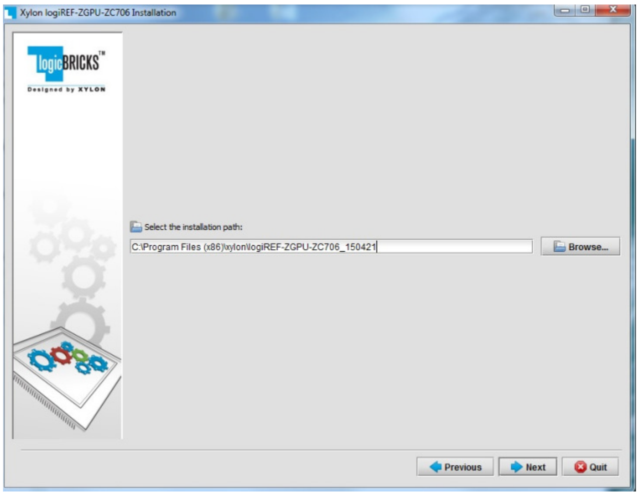
CAREFULLY READ THIS LICENSE AGREEMENT ("**AGREEMENT**"). BY CLICKING THE "ACCEPT" OR "AGREE" BUTTON, OR OTHERWISE ACCESSING, DOWNLOADING, INSTALLING OR USING THE LICENSED MATERIALS, YOU AGREE ON BEHALF OF LICENSEE TO BE BOUND BY THIS AGREEMENT. "**LICENSEE**" OR "**YOU**" MEANS THE CORPORATION OR OTHER LEGAL ENTITY TO WHICH XYLON D.O.O., A CROATIAN CORPORATION WITH AN OFFICE AT FALLEROVO SETALISTE 22, 10000 ZAGREB, REPUBLIC OF CROATIA ("**XYLON**") HAS ISSUED THE LICENSE DESCRIBED HEREIN. IF YOU DO NOT AGREE TO ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT, DO NOT CLICK THE "ACCEPT" OR "AGREE" BUTTON, AND DO NOT ACCESS, DOWNLOAD, INSTALL OR USE THE LICENSED MATERIALS. AS USED HEREIN, THE "**EFFECTIVE DATE**" MEANS THE DATE ON WHICH LICENSEE CLICKS THE "ACCEPT" OR "AGREE" BUTTON (IDENTIFIED ABOVE), PURCHASES OR OTHERWISE ACCESSES, DOWNLOADS, INSTALLS OR USES THE LICENSED MATERIALS, WHICHEVER OCCURS FIRST.

1. **License.** On and subject to the terms of this Agreement, Xylon hereby grants Licensee a nonexclusive, non-transferable, non-sublicensable, personal, revocable and royalty-free license to evaluate the SoC and FPGA IP cores, software drivers, software programs, materials, data and documentation identified in *Exhibit A* (collectively the "Licensed Materials") solely for the purpose of evaluating the Licensed Materials for use with Xilinx SoC and FPGA device(s), and solely within Licensee's premises. Licensee may install and use the Licensed Materials at one (1) development seat at only one Licensee's site. Use of the Licensed Materials for any other purpose, or by any third party other than Licensee, is prohibited.

2. **Ownership.** The Licensed Materials are licensed, not sold. Except for the limited license granted in Section 1 above, Licensee acknowledges that all intellectual property rights in and to the Licensed Materials and all copies thereof are and will remain the sole property of Xylon or its suppliers, and that no additional rights are granted or conferred, by implication, estoppel or otherwise. The Licensed Materials are protected by laws and international treaty provisions covering intellectual property and industrial rights.

3. **Restrictions.** Licensee shall not decrypt, decompile, reverse engineer, disassemble, or otherwise reduce to a human-perceivable form, the Licensed Materials delivered in non-human readable form. Licensee shall not hypothecate, rent, lend, time-share, assign, convey, sell, sublicense, display, distribute, loan, lease or

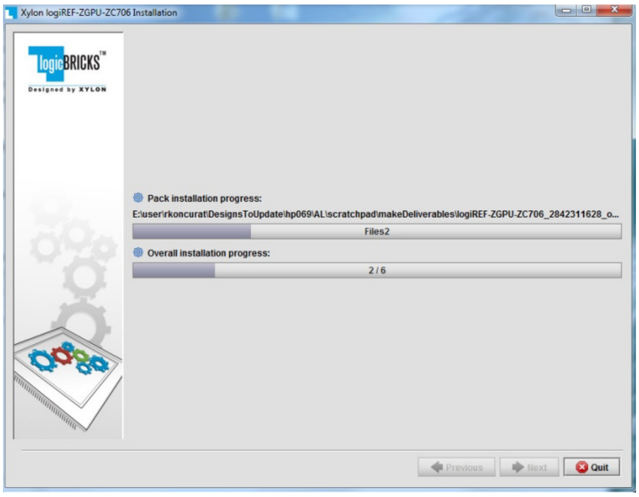
Figure 6: Installation Process – Step 1



Select the installation path:

C:\Program Files (x86)\xylon\logiREF-ZGPU-ZC706_150421

Figure 7: Installation Process – Step 2



Pack installation progress:

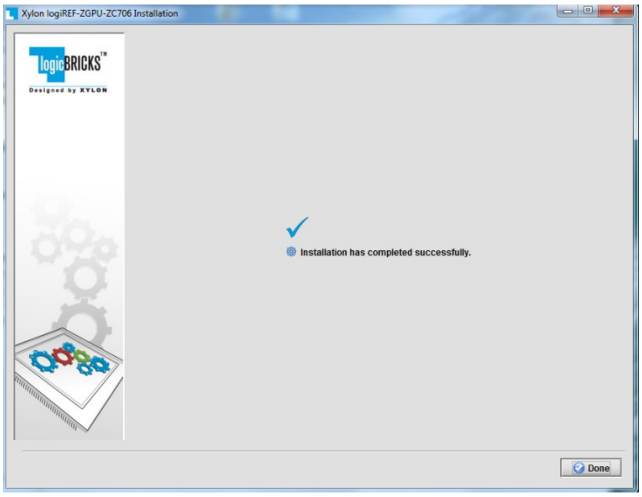
E:\user\rkoncurat\DesignsToUpdate\hp069\AL\scratchpad\makeDeliverables\logiREF-ZGPU-ZC706_2842311628_o...

Files2

Overall installation progress:

2 / 6

Figure 8: Installation Process – Step 3



✓ Installation has completed successfully.

Figure 9: Installation Process – Step 4

3.1.1 Filesystem Permissions of the Installed Folder (Windows 7)

The reference design installed in the default path `C:\Program Files\xylon` will inherit read-only filesystem permissions from the parent folder. This will block you in opening the hardware project file in Xilinx Vivado tools. Therefore it is necessary to change the filesystem permissions for the current user to "Full control" preferably.

To change the user permissions for `C:\Program Files\xylon` folder and all of its subdirectories, right click on the `C:\Program Files\xylon` folder and select "Properties". Under "Security" tab select "Edit". Select "Users" group in the list and check "Full control" checkbox in the "Allow" column.

Folder Structure

Figure 10 gives a top level view of the directories and files included with the logiADAK-VDF-SDSoC video design framework. Table 4 explains the purpose of directories.

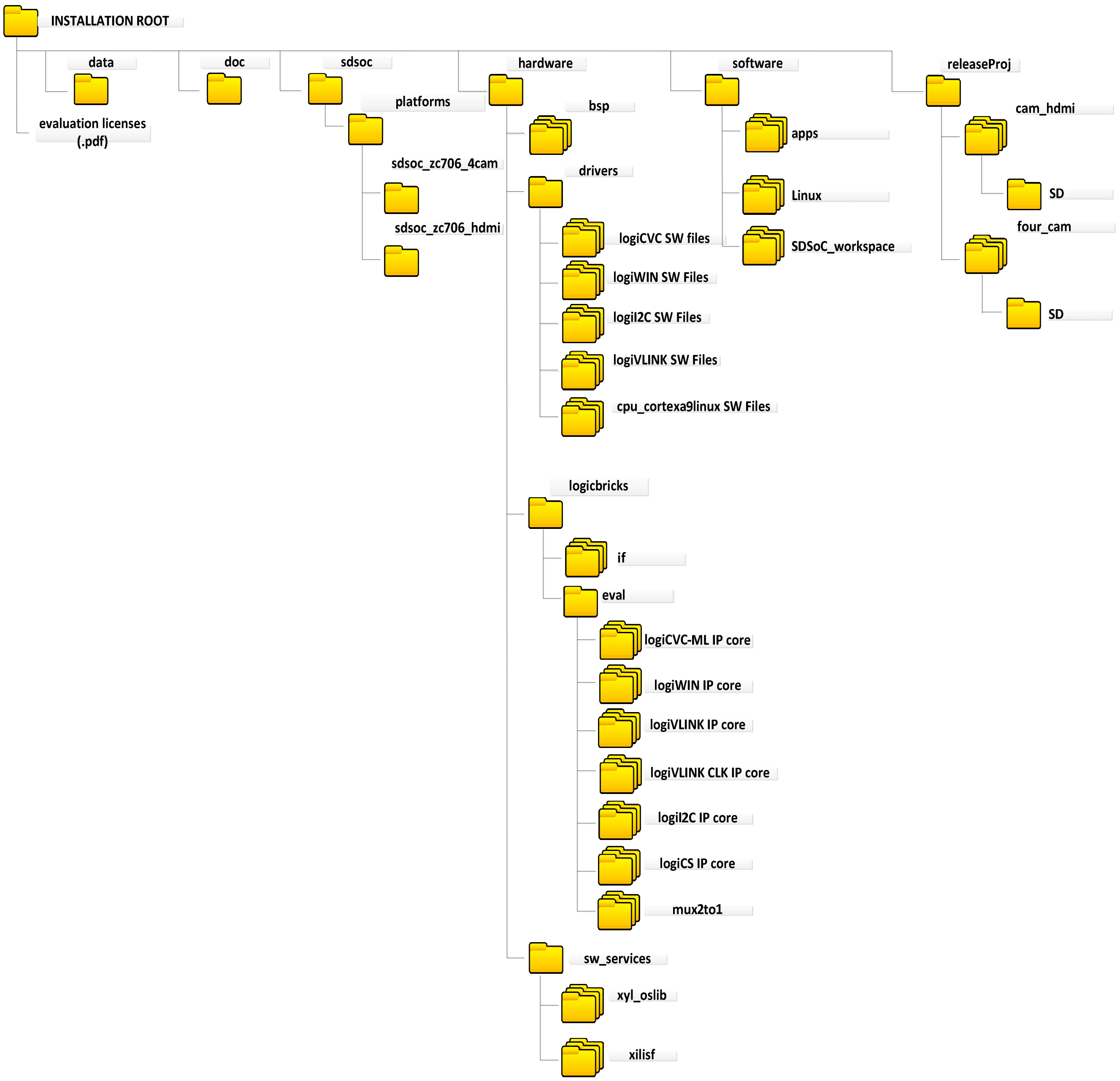


Figure 10: Folder Structure

Folder		Purpose
INSTALLATION ROOT		This folder contains the *start.html* page – the jump-start navigation page through the reference design.
doc		Project documentation.
sdsoc/platforms		SDSoC platform files
	sdsoc_zc706_hdmi	Cam/hdmi platform files
	sdsoc_zc706_4cam	Four cam platform files
hardware		
	bsp	Xylon Linux user space Board Support Package (BSP); custom Xylon BSP compatible with the Xilinx SDK. It enables users to quickly build Linux User space applications within the SKD workspace.
	drivers	Standalone (bare-metal) drivers for logicBRICKS IP cores with documentation and examples.
	logicbricks/if	Xylon custom IP core interfaces (bus definitions).
	logicbricks/eval	Evaluation logicBRICKS IP cores. IP cores' User's Manuals are stored in doc subdirectories.
	sw_services	xyl_oslib – Xylon OS abstraction library for Xilinx Xilkernel embedded kernel – use in standalone (non-OS) applications.
software		
	apps	Demo applications source code files
	Linux/kernel	Linux kernel and device tree configuration files.
	SDSoC_workspace	Xilinx SDSoC workspace folder for building bare-metal applications.
releaseProj		Prepared binaries ready for download to SD card.
	cam_hdmi/SD	Cam/hdmi platform with hardware implemented sobel filter
	four_cam/SD	Four cam platform with hardware implemented sobel filter

Table 4: Explanation of the logiADAK-VDF folder structure

4 GETTING LOGICBRICKS IP LICENSES

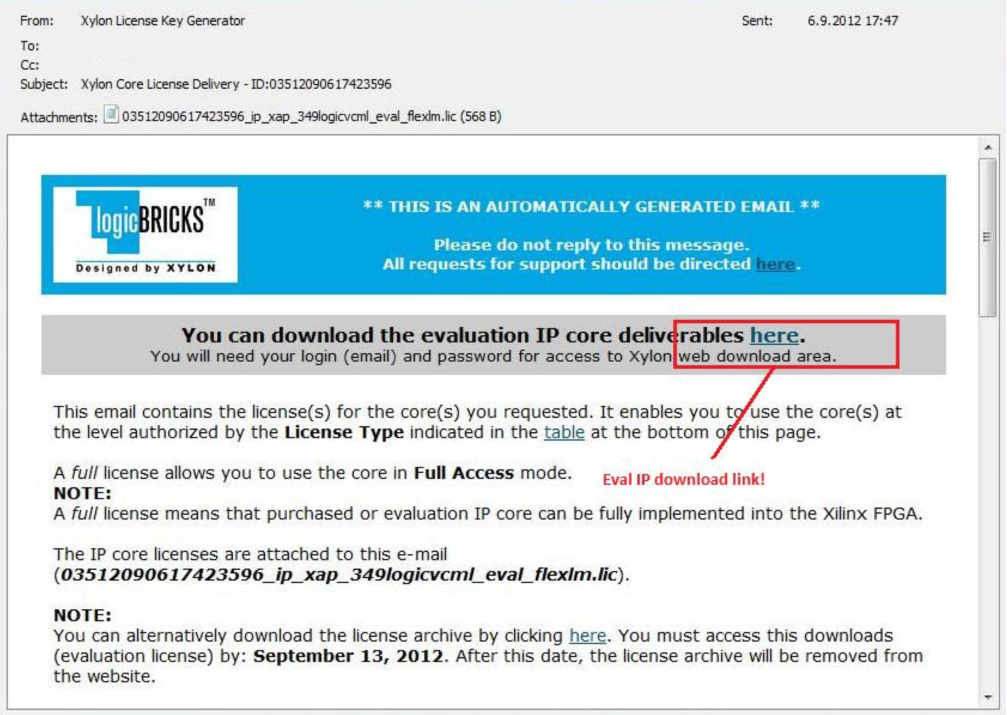
The logiADAK-VDF-SDSoC installation comes with the evaluation versions of the logicBRICKS IP cores, and in order to be able to change the provided reference designs, you need to request the proper licenses from Xylon.

Please contact Xylon Technical Support Service support@logicbricks.com and immediately provide your Ethernet MAC ID number or Sun Host ID.

For instructions how to find your Ethernet MAC or host ID, please visit: http://www.logicbricks.com/Documentation/Article.aspx?articleID=KBA-01186-M0JXKD..

For each logicBRICKS IP core used in the logiADAK-VDF-SDSoC reference designs Xylon will generate and send to you separated e-mails with the license keys (file) and full instructions for setting up the license key and downloading the logicBRICKS IP core. Please follow the provided instructions.

If you experience any troubles during the registration process, please contact Xylon Technical Support Service – support@logicbricks.com.


From: Xylon License Key Generator
Sent: 6.9.2012 17:47
To:
Cc:
Subject: Xylon Core License Delivery - ID:03512090617423596
Attachments: 03512090617423596_ip_xap_349logicvcml_eval_flexlm.lic (568 B)

** THIS IS AN AUTOMATICALLY GENERATED EMAIL **

Please do not reply to this message.
All requests for support should be directed here.

You can download the evaluation IP core deliverables here.
You will need your login (email) and password for access to Xylon web download area.

This email contains the license(s) for the core(s) you requested. It enables you to use the core(s) at the level authorized by the **License Type** indicated in the table at the bottom of this page.

A *full* license allows you to use the core in **Full Access** mode.

NOTE:
A *full* license means that purchased or evaluation IP core can be fully implemented into the Xilinx FPGA.

The IP core licenses are attached to this e-mail (*03512090617423596_ip_xap_349logicvcml_eval_flexlm.lic*).

NOTE:
You can alternatively download the license archive by clicking here. You must access this downloads (evaluation license) by: **September 13, 2012**. After this date, the license archive will be removed from the website.

Figure 11: E-mail with logicBRICKS License and Download Instructions

5 LOGIADAK-VDF-SDSOC REFERENCE DESIGN

The logiADAK-VDF- SDSoC referent design contains two SDSoC platform designs:

- sdsoc_zc706_hdmi
- sdsoc_zc706_4cam

Xylon's platforms for the targeted hardware kit are designed following Xilinx's SDSoC platform design specifications [REF 2].

Platform designs are based on the logiADAK-VDF reference designs functionality with an addition of the Sobel filter implementation. The Sobel filter is implemented between an active input and the display output. Filter implementation provides a simple example how to implement C-code based accelerator in the Programmable Logic within the SDSoC design environment.

The **`sdsoc_zc706_hdmi`** and **`sdsoc_zc706_4cam`** platforms support Linux applications and include precompiled Linux drivers for all logicBRICKS IP cores. Linux and Bare-metal software drivers source files are included within the reference design deliverables – please see the Table 4.

The provided platforms also include the pre-built SoC hardware folder for faster user workflow described in [REF 3]. The SoC configuration bitstream, which is contained in the platform hardware folder, enables users to run and verify custom developed applications, entirely in software and without engaging in the lengthy process of hardware implementation. However, that process cannot be used with the provided demo application when the Sobel filter is marked for hardware acceleration. Alternatively, user can generate bitstream for the Cam/HDMI and Four-Cam applications and then uncheck the **`Generate Bit Stream`**. It will enable the tools to rebuild only software portions of the design.

5.1 CAM-HDMI SDSoC Design and Memory Layout

This platform design (Figure 12) supports a single video camera or a single HDMI video input (only one active at a time), and the single display output.

The following Zynq SoC resources are not used by the **`sdsoc_zc706_hdmi`** platform and can be used for other purposes within the SDSoC development environment:

- Clocks:
    - Clock id 1 (default): 100 MHz
    - Clock id 2: 200 MHz
    - Clock id 3: 142.86 MHz
- PS-PL ports:
    - M_AXI_GP1
    - S_AXI_HP1

- S_AXI_HP3
- S_AXI_ACP

- Interrupts:
  - Over xlconcat IP, ports from 3 to 15

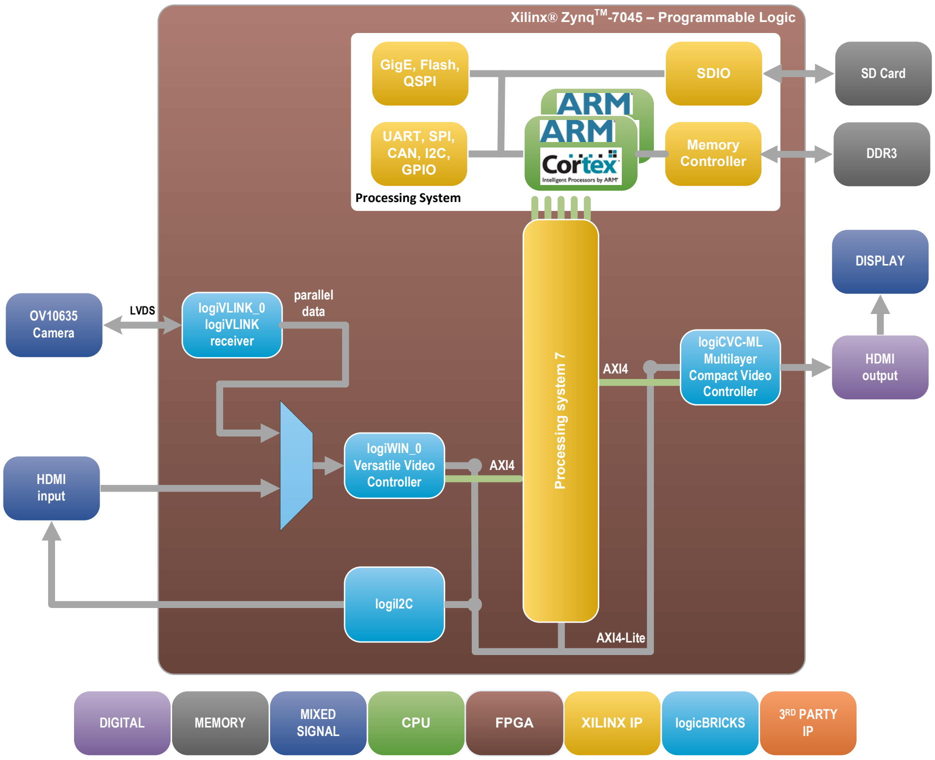


Figure 12: CAM-HDMI SoC Design – Block Diagram (Pre-Defined Platform)

(Clock Generator Module and other utility IP cores are not shown)

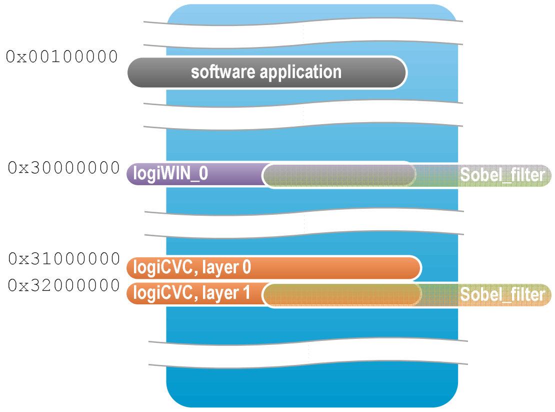


Figure 13: CAM-HDMI Design Memory Layout

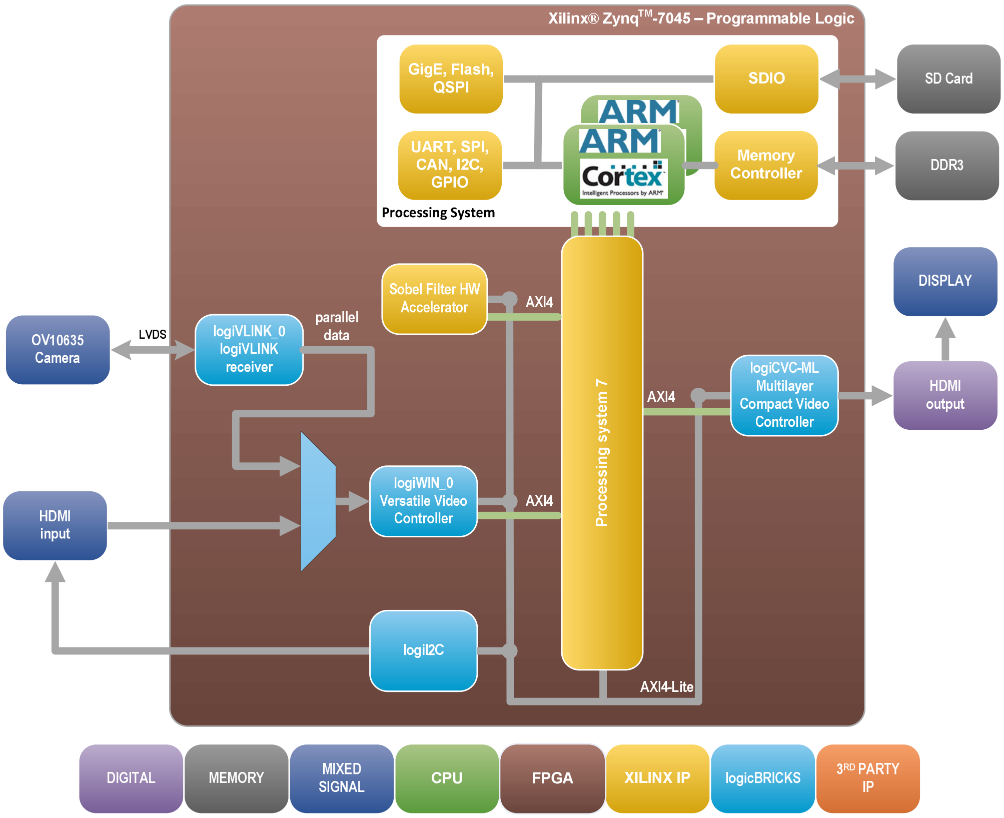


Figure 14: CAM-HDMI SoC Design – Block Diagram (Platform with the Sobel Filter IP)

(Clock Generator Module and other utility IP cores are not shown)

5.2 FOUR-CAM SoC Design and Memory Layout

This platform design (Figure 15) supports four camera inputs, and the single display output. The following Zynq SoC resources are not used by the `sdsoc_zc706_4cam` platform and can be used for other purposes within the SDSoC development environment:

- Clocks:
  - Clock id 1 (default): 100 MHz
  - Clock id 2: 200 MHz
  - Clock id 3: 142.86 MHz
- PS-PL ports:
  - M_AXI_GP0
  - S_AXI_HP2
  - S_AXI_ACP
- Interrupts:
  - Over xlconcat IP, ports from 5 to 15

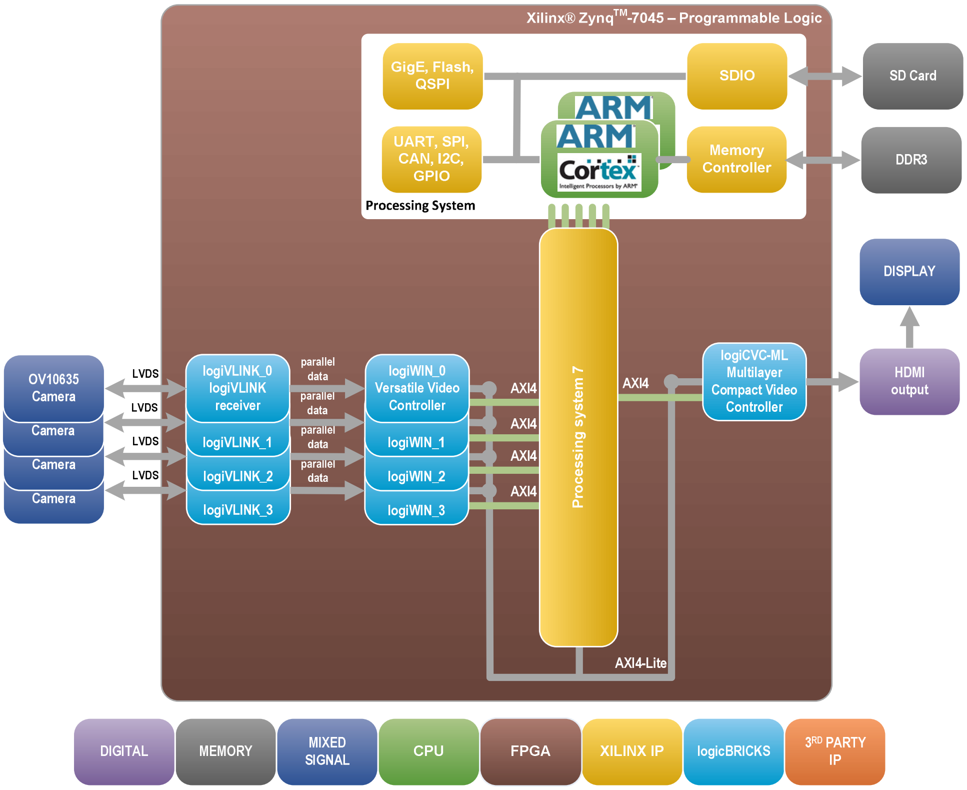


Figure 15: FOUR-CAM SoC Design– Block Diagram (Pre-Defined Platform)

(Clock Generator Module and other utility IP cores are not shown)

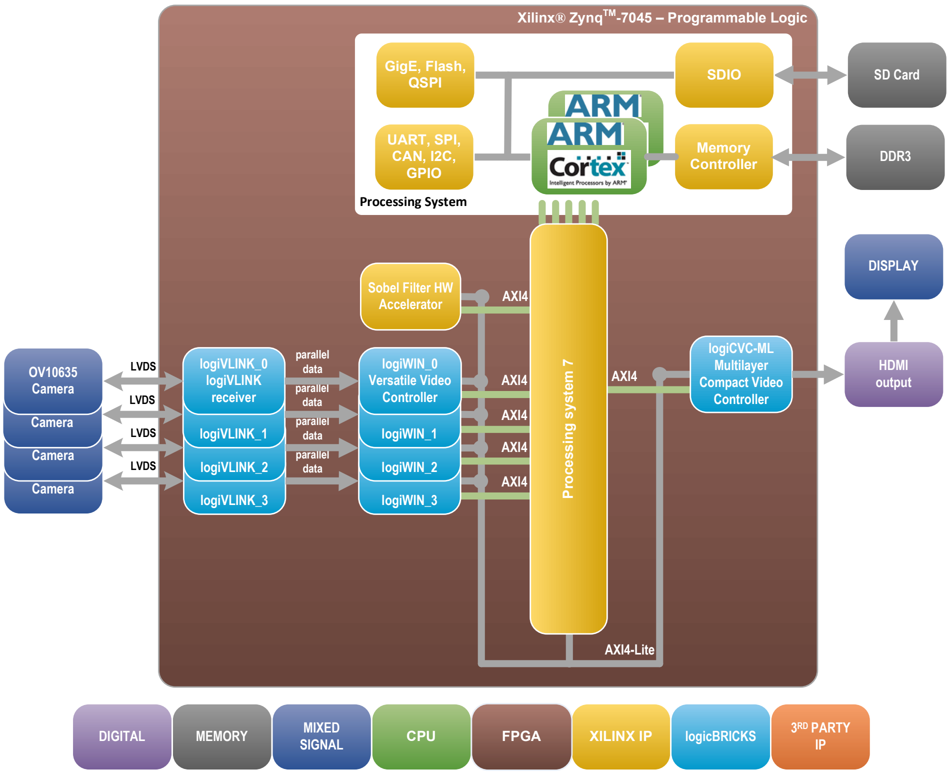


Figure 16: FOUR-CAM SoC Design– Block Diagram (Platform with the Sobel Filter IP)

(Clock Generator Module and other utility IP cores are not shown)

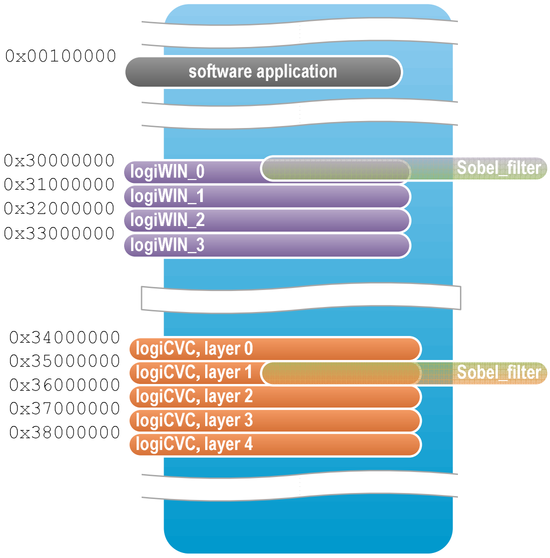


Figure 17: FOUR-CAM Design Memory Layout

5.3 Using the logiADAK-VDF-SDSoC Platforms and the Hardware Accelerator Based on the C-code Description

Please copy the *sdsoc_zc706_hdmi* and *sdsoc_zc706_4cam* platform folders to folder *<SDSoC_install_dir>/platforms* within your SDSoC installation folders.

5.4 Using and configuring Hardware Accelerator as C-code Based IP core

The provided software demo includes the Sobel filter algorithm as the C++ function `sds_sobel()`(*sds_sobel.cpp*). The function is used as a simple example of the C-code based hardware accelerator implementation generated by the SDSoC tool.

The HLS generated Sobel filter accelerator connects to the AXI-Stream accelerator adapter that drives all inputs and captures all outputs. The adapter has two FIFO interfaces for streaming video data to and from the accelerator, converts video data streams to the AXI-Stream format, and connect the AXI DMA Rx and Tx channels via the AXI-Stream router.The instantiated data mover is the AXI DMA in a scatter-gather mode. The width, height, and stride parameters are simple control signals (scalars) driven by the adapter.

Note that the m2m pipeline is not video-timing accurate. The DMA reads and writes video frames without inserting any horizontal blanking between video lines. An interrupt is issued by the DMA engine separately for the read and write channels upon completion of a frame read/write operation.
The DMA engine translates AXI-Stream data to AXI-Memory-mapped data. The output is connected to the SDSoC available HP high performance PS/PL interface via the AXI interconnect.
The AXI-Lite control interfaces is connected to the SDSoC available GP general purpose PS/PL interface.

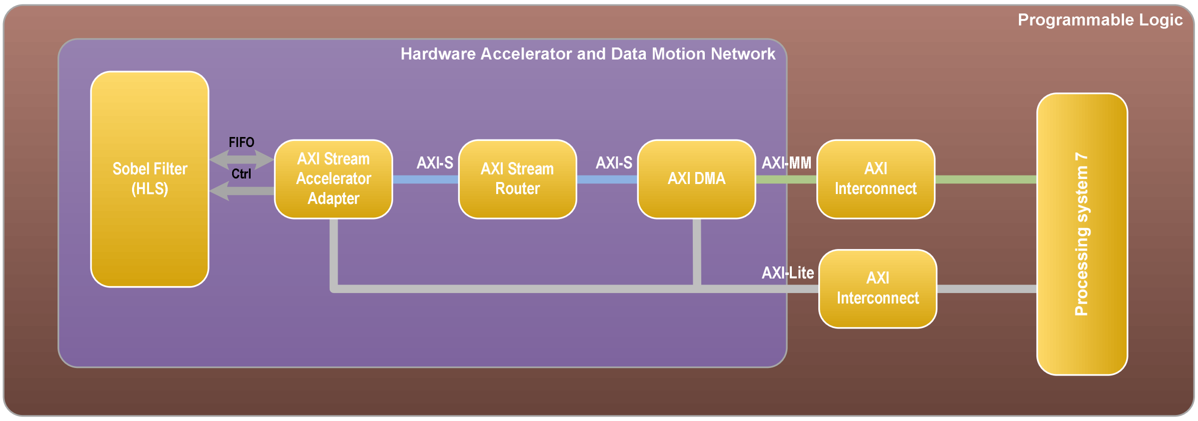


Figure 18: Hardware Accelerator and Data Motion Network entirely generated by the SDSoC tool based on a C-code description

5.5 Software Description

5.5.1 Main function

The `main.cpp`, which is the main application file in CAM-HDMI and FOUR-CAM reference designs, uses the `hw_Init` function to initialize the hardware platform, and the `main_loop` function to implement the Sobel video filter and to allocate memory for video framebuffers. Both reference designs use the double buffering synchronization between video inputs and the video outputs in order to avoid unwanted video display artifacts, such as random stripes caused by framebuffers overrun, i.e. the video input data is written in the memory framebuffer currently shown on the display. While the CAM-HDMI demo application initializes memory buffers for a single video channel, the FOUR-CAM application need to initialize buffers for every video camera. The same applies to video processing and panning functions.

Software code for double buffering synchronization mechanism uses several functions organized in the *VideoInput* and the *VideoOutput* mini libraries. Two video input and two video output framebuffers per video camera input are initialized before the infinite `while` loop (see Figure 19). All Xylon drivers, such as logiWIN and logiCVC software drivers, are User Space I/O (UIO) drivers. Current input framebuffer and output framebuffer are exchanged on the UIO interrupt of a logiWIN driver, which is initiated before each new video input frame. The example Sobel filter, which is implemented in the `startProcessing` function, filters video from the last refreshed output framebuffer, currently defined by the `VideoInput_waitNewBuff` function. The processed video is written to the current display output framebuffer. All framebuffers are deinitialized by `VideoInput_deintialize` and `VideoOutput_deintialize` functions that release the allocated memory space.

The `startProcessing` function applies Sobel filter on an input image, which is read from input buffer, and writes it to output framebuffer. By removing `sds_sobel` function user can easily put its own image processing function.

```
void startProcessing(InputBuffT *pInBuff, OutputBuffT *pOutBuff)
{
  sds_sobel((unsigned   short   *)pInBuff->pVirtBase,   (unsigned   short
*)pOutBuff->pVirtBase, pInBuff->vres, pInBuff->hres, pInBuff->stride/2);
}
```

5.5.2 The Video input Library

The `VideoInput_initBuffers` function maps input framebuffer, fills input buffer structure with used parameters and returns value of the UIO identifier.

The `VideoInput_waitNewBuff` function waits for the UIO interrupt and switches the video input framebuffer at the end of the new video frame signalized by the UIO interrupt.

The `VideoInput_deintialize` function deinitializes input framebuffers.

```
typedef struct
{
  void *pVirtBase; // pointer to input framebuffer
  int vres;        // vertical resolution
  int hres;        // horizontal resolution
  int stride;      // image stride in pixels
}InputBuffT;
```

The `InputBuffT` structure is used to store information about input framebuffer and to send it to Sobel filter.

5.5.3 The VideoOutput Library

The `VideoOutput_initBuffers` function maps output framebuffer, fills output buffer structure with used parameters and reads all `vinfo` data.

The `VideoOutput_waitVsync` function waits for V-sync if it hasn't appeared since the last call of `VideoOutput_showBuffer` function.

The `VideoOutput_showBuffer` function reads panning offset and pan the screen so the right framebuffer is shown.

The `VideoOutput_deintialize` function deinitializes output framebuffers.

```
typedef struct
{
  void *pVirtBase;                  // pointer to input framebuffer
  int phAddr;                       // phisical address of a framebuffer
  int vres;                         // vertical resolution
  int hres;                         // horizontal resolution
  int stride;                       // image stride in pixels
  int panOffset;                    // panning offset
  struct fb_var_screeninfo vinfo;   // screen vinfo
  int fbFileDesc;                   // framebuffer file descriptor
}OutputBuffT;
```

The `OutputBuffT` structure is used to store information about output framebuffer and to send it to Sobel filter. It also stores layer information which is used for panning the screen.

5.5.4 Application control

To change the camera input for the display output press the number, '1', '2', '3' or '4', on a keyboard, or by pressing left or right pushbutton on the ZC706 board. The application can be stopped by pressing 'Q' or 'q' on a keyboard. In CAM-HDMI reference design the HDMI input has higher priority than any camera input and will be immediately displayed when connected. Switching to the camera video input requires connect/disconnect of the HDMI video input. The application can be stopped by pressing 'Q' or 'q' on a keyboard.

5.5.5 Input resolution and the framerate

Input resolution in both reference designs is 1280x800. It is also resolution of the Xylon camera. The HDMI video input with a different video resolution is resized to 1280x800 resolution. Input framerate for FOUR-CAM and CAM-HDMI design is 30 frames per second.

5.5.6 Application functional and timing diagram

In a `while` loop all image processing function has to be done and also panning and waiting for interrupt, but they are quite fast. To avoid unwanted video display artifacts image processing function has to last maximum 30 ms.

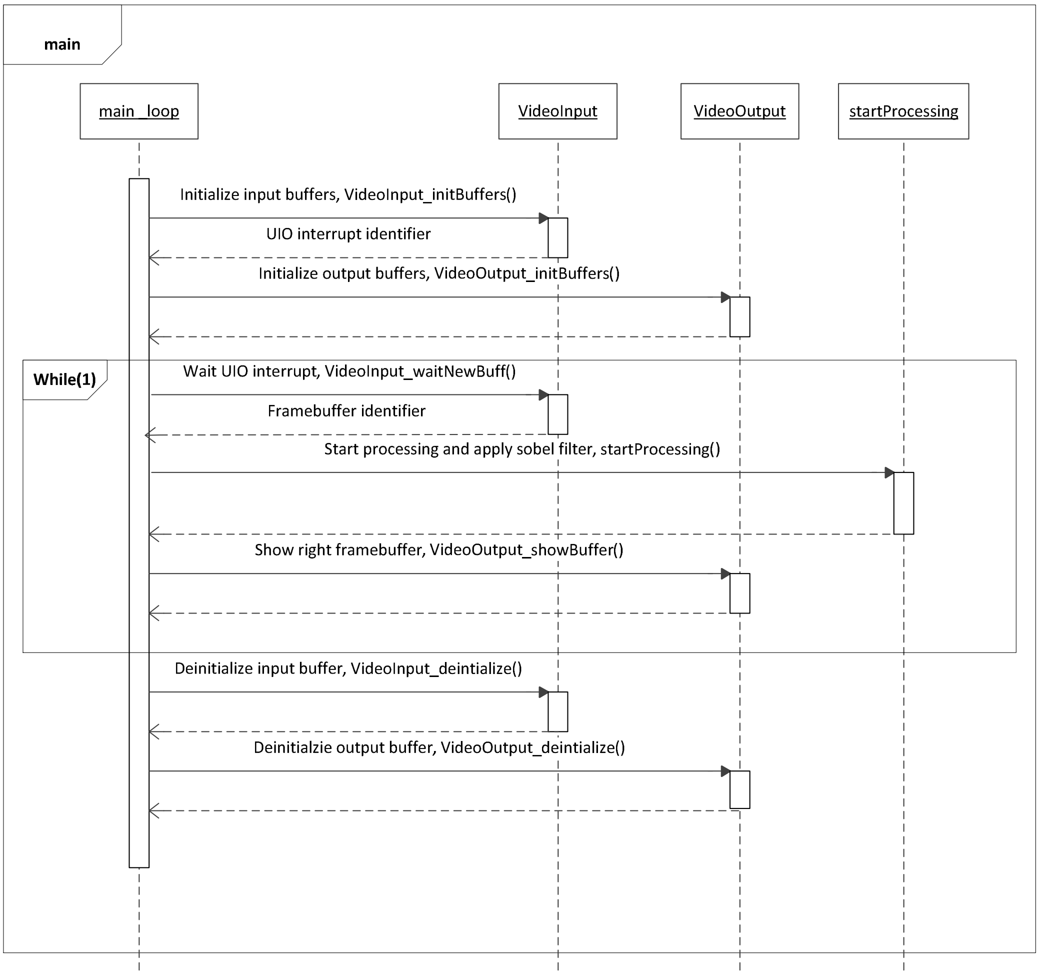

main
main_loop
VideoInput
VideoOutput
startProcessing
Initialize input buffers, VideoInput_initBuffers()
UIO interrupt identifier
Initialize output buffers, VideoOutput_initBuffers()
While(1)
Wait UIO interrupt, VideoInput_waitNewBuff()
Framebuffer identifier
Start processing and apply sobel filter, startProcessing()
Show right framebuffer, VideoOutput_showBuffer()
Deinitialize input buffer, VideoInput_deintialize()
Deinitialzie output buffer, VideoOutput_deintialize()

Figure 19: Application functional and timing diagram

6 QUICK START

6.1 Run the Precompiled Linux Demo Examples

To enable rapid testing of the hardware setup, Xylon provides application demo binaries in the *binaries* folder of the deliverables. There are two demo applications: the camera/HDMI and the four cameras demo.

To run the four cameras demo copy the content of the *binaries/four_cam/bin* folder to the root folder of the SD card.

To run the camera/HDMI demo, copy the content of the *binaries /cam_hdmi/bin* folder to the root folder of the SD card.

SD card should be formatted as FAT32.

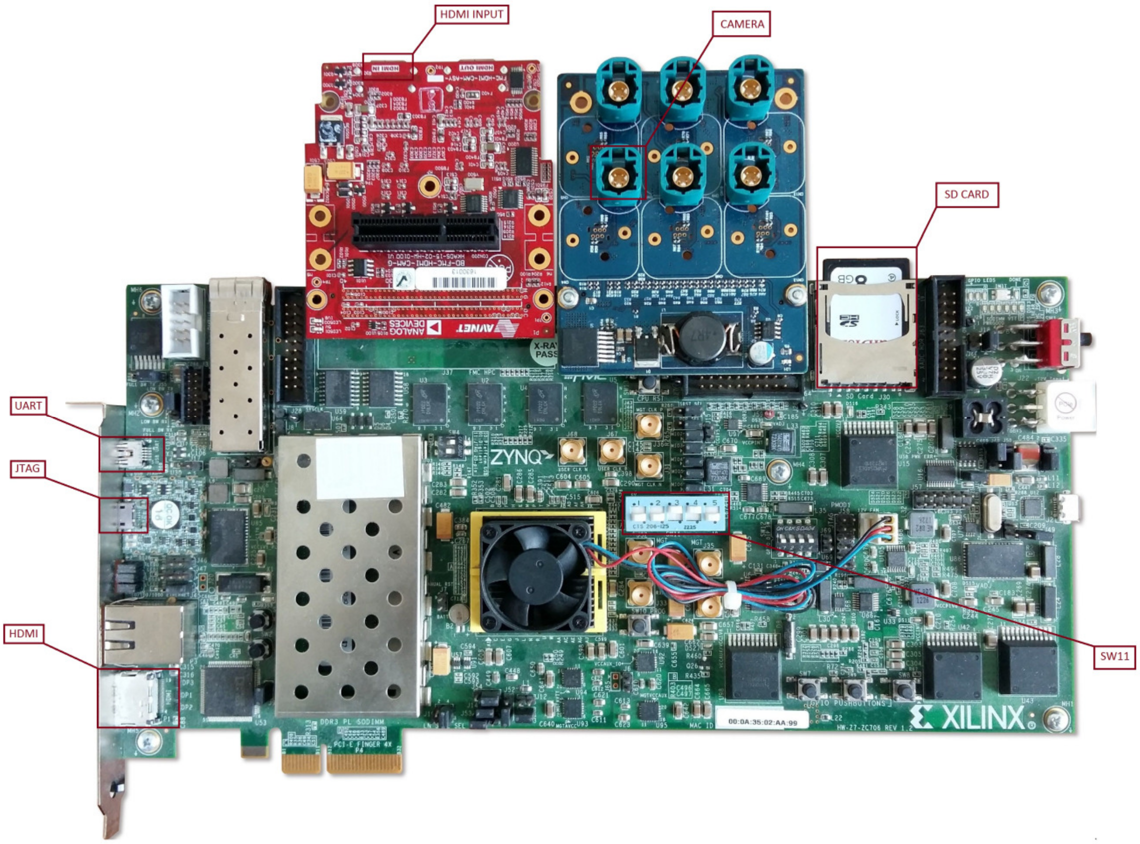


Figure 20: CAM-HDMI HW Setup

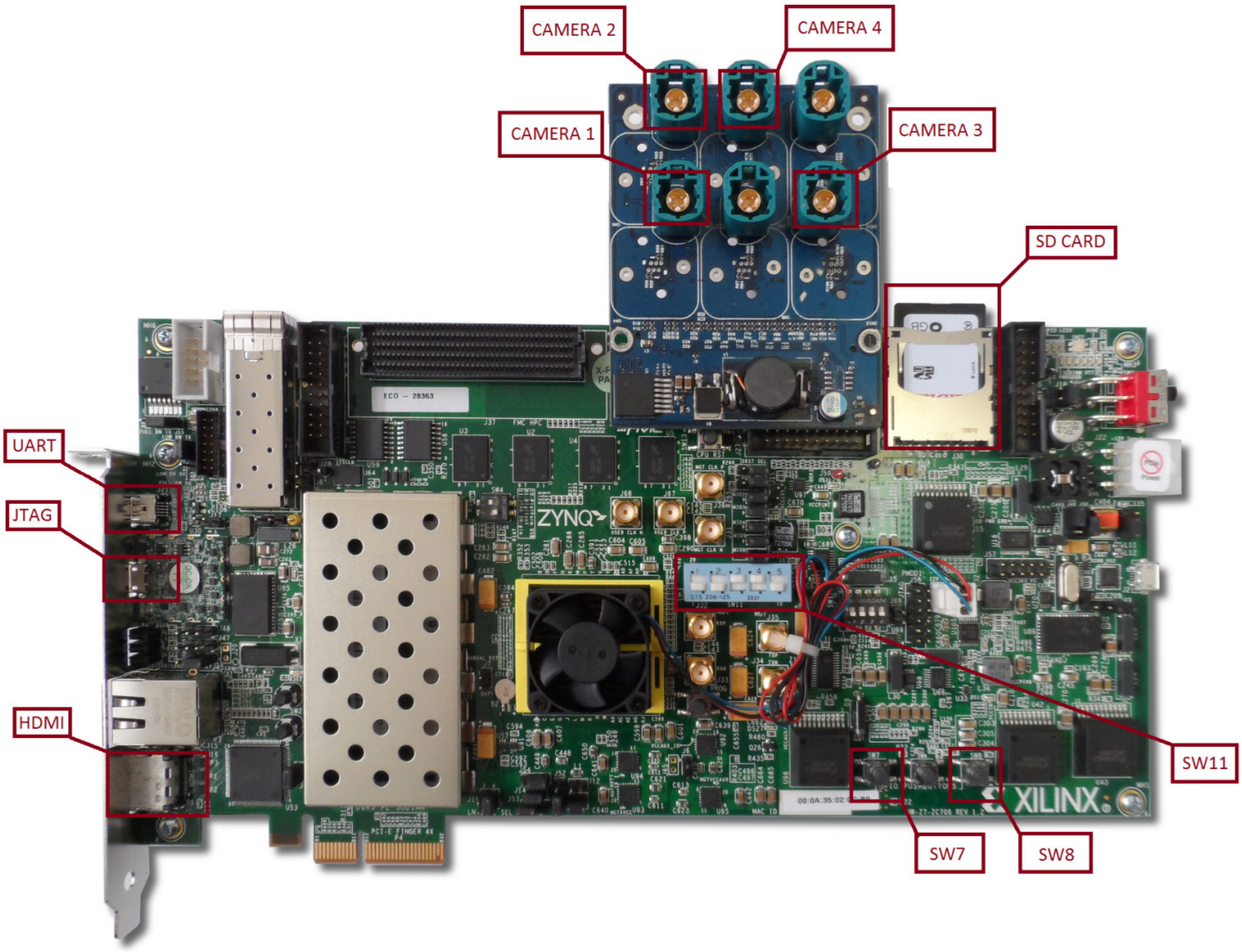


Figure 21: FOUR-CAM HW Setup

Optionally, you can use a serial terminal program (baud rate 115200 8N1) and the USB UART connection to the ZC706 board to monitor the system's operation.

For full explanation of the ZC706's features and settings, please check the documentation Xilinx UG954.

6.2 Demo controls

Start the FOUR-CAM demo design and the display output will show the video stream from one of the four attached video cameras. Change the displayed camera input by pressing the number, '1', '2', '3' or '4', on a keyboard, or by pressing left (SW7) or right (SW8) switch-buttons on the ZC706 board. Stop the application by pressing 'Q' or 'q' on a keyboard.

Start the CAM-HDMI demo design. The attached PC monitor will show the HDMI video, if there is the HDMI video source connected to the HDMI video input on the FMC card. Otherwise, the display will show the video input from the attached video camera. Stop the application by pressing 'Q' or 'q' on a keyboard.

6.3 Change the Delivered Software

6.3.1 Xilinx Development Software

The logiADAK-VDF-SDSoC video design framework reference designs and Xylon logicBRICKS IP cores are fully compatible with Xilinx SDSoC Development Environment 2016.2. Future design releases shall be synchronized with the newest Xilinx development tools.

Licensed users of Xilinx tools can use their existing software installation for the logiADAK-VDF-SDSoC evaluation and modifications.

6.3.2 Set Up Linux System Software Development Tools

Set of ARM GNU tools are required to build the Linux software and applications. The complete tool chain for the Zynq-7000 All Programmable SoC can be obtained from the Xilinx ARM GNU Tools wiki page: http://wiki.xilinx.com/zynq-tools. Access to tools requires a valid, registered Xilinx user login name and password.

6.3.3 Set Up git Tools

Git is a free Source Code Management (SCM) tool for managing distributed version control and collaborative development of software. It provides the developer a local copy of the entire development project files and the very latest changes to the software.
Visit http://wiki.xilinx.com/using-git to get instructions how to use Xilinx git.

To get the latest version of Xylon logicBRICKS software drivers for Linux operating system, please visit Xylon's git: https://github.com/logicbricks.

6.3.4 Setting up the SDSoC workspace

All logiADAK-VDF-SDSoC software applications are delivered in the source code to enable users to do software customizations. This paragraph explains how to setup the Xilinx SDSoC environment for software customizations.

Quick steps required to set up the SDSoC workspace:

1. Open SDSoC SDK, select workspace: ***logiADAK-VDF-SDSoC_vX_Y_Z/software/SDSoC_workspace***

2. In the SDSoC SDK go to: *Project* and exclude *Build automatically* (optional, but recommended)
3. In the SDSoC SDK go to: *File → Import → General → Existing projects to workspace → Next,* in *Select a root directory* choose ***logiADAK-VDF-SDSoC_vX_Y_Z/software/SDSoC_workspace***, select all projects and click *Finish*
4. For optimized SW implementation it is recommended to set project build configuration to *SDRelease.* In Project explorer window -> Right click on the SDSoC project: *Build Configurations -> Set Active -> SDRelease*
5. If you want to use hardware acceleration, function *sds_sobel()* needs to be marked as [H]. This is done by default, so no change is needed.
6. If the *Build automatically* option has been disabled, build all imported applications manually.
7. If you want to run the sobel filter in SW, function *sds_sobel()* needs to be toogled to SW. In Project explorer window:
    - Browse to *src/sds_sobel.cpp* file. Expand content of the file by clicking arrow mark on left side (|>)
    - On *sds_sobel* function: right click and Toggle HW/SW (mark [H] disappears)
8. If the *Build automatically* option has been disabled, build the application manually.
9. Copy the sd_card folder content to the to the root folder of the SD card.
    - sd_card folder find in the application build generated folder. Build generated folder has the same name as the chosen build configuration and is located in the SDSoC workspace application folder (e.g. for the build configuration *SDRelease* and application *app_zc706_4cam_linux* the build generated folder is *app_zc706_4cam_linux/SDRelease*).

6.4 Software Instructions – Linux Software

Xylon provides the Linux Framebuffer driver. Zynq tool chain, Linux kernel and file system used for development and demonstrations of Xylon drivers are provisions of Xylon.

- Linux kernel building instructions and `dts` files can be found in *software\Linux\kernel* folder. Used Kernel version is Kernel 4.4.
- Running Linux applications with the ZC706 board setup for the precompiled SD card image. By pressing a number 1 to 4 user can switch the camera, if used application is *app_zc706_4cam_linux.*

7 REVISION HISTORY

Version	Date	Author	Approved by	Note
1.00.a	June 29th, 2016	A.Bogdanic	G.Galic	Initial
1.01.a	September 10th,2016	A.Bogdanic		Version of Vivado Development Tool changed to 2016.2. Added description of C-code based hardware accelerator implementation generated by the SDSoC tool.
2.00.a	November 21st,2016	A.Bogdanic		Added software description

8 REFERENCES

Table 5: List of References

Reference	Description
REF [1]	Xylon, logiADK-VDF (logiADK-VDF _v1_00_a.pdf)
REF [2]	Xilinx, UG1146 (ug1146-sdsoc_platforms_and_libraries.pdf)
REF [3]	Xilinx, UG1027 (ug1027-intro_to_sdsoc.pdf)