

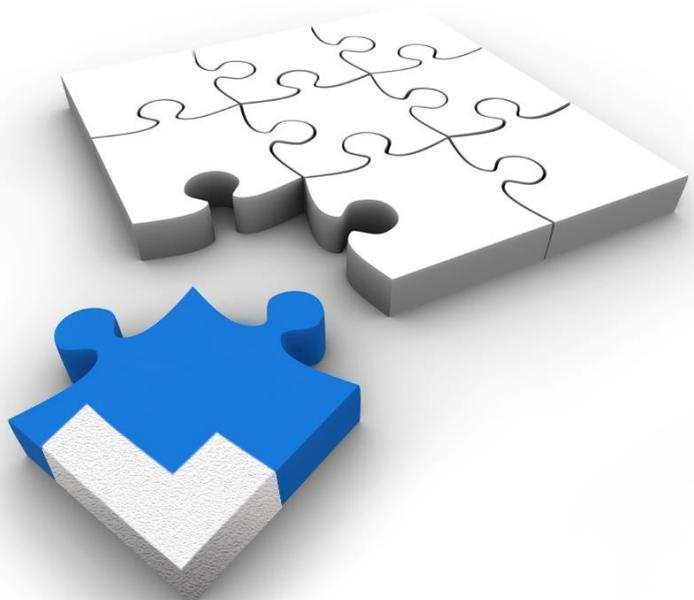
# ***logiREF-ZGPU-ZED***

***Xylon logicBRICKS™ Graphics Processing Unit (GPU) Reference Design for Xilinx® Zynq®-7000 All Programmable SoC based ZedBoard™ from Avnet Electronics Marketing***

## **User's Manual**

***Version: 2.02.a***

logiREF-ZGPU-ZED\_v2\_02\_a.docx





All rights reserved. This manual may not be reproduced or utilized without the prior written permission issued by Xylon.

Copyright © Xylon d.o.o. logicBRICKS™ is a registered Xylon trademark.

All other trademarks and registered trademarks are the property of their respective owners.

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.

<b>1</b>	<b>INTRODUCTION.....</b>	<b>5</b>
1.1	DESIGN DELIVERABLES.....	6
1.1.1	Hardware Design Files .....	6
1.1.2	Software .....	6
1.1.3	Binaries .....	7
1.2	USAGE MODES.....	7
1.2.1	Quick Evaluation with no HW and/or SW Changes .....	7
1.2.2	Develop Standalone and Linux Software, no HW Changes .....	7
1.2.3	Full GPU Customization, HW and SW Changes .....	7
1.3	XILINX DEVELOPMENT SOFTWARE.....	8
1.4	GRAPHICS DEMO PREVIEW .....	8
<b>2</b>	<b>LOGICBRICKS IP CORES .....</b>	<b>9</b>
2.1	ABOUT LOGICBRICKS IP LIBRARY .....	9
2.2	EVALUATION LOGICBRICKS IP CORES.....	10
2.3	LOGICBRICKS IP CORES USED IN THIS DESIGN .....	11
2.3.1	logiCVC-ML Compact Multilayer Video Controller.....	11
2.3.2	logiBITBLT Block Transfer 2D Graphics Accelerator.....	12
2.3.3	logi3D Scalable 3D Graphics Accelerator .....	12
2.3.4	logiCLK Programmable Clock Generator .....	13
2.4	LOGICBRICKS IP CORES FOR VIDEO PROCESSING.....	14
<b>3</b>	<b>GET AND INSTALL THE REFERENCE DESIGN.....</b>	<b>15</b>
3.1	REGISTRATION PROCESS.....	15
3.2	INSTALLATION PROCESS .....	17
3.2.1	Filesystem Permissions of the Installed Directory (Windows 7).....	19
3.3	DIRECTORY STRUCTURE.....	19
<b>4</b>	<b>GETTING LOGICBRICKS EVALUATION LICENSES.....</b>	<b>22</b>
<b>5</b>	<b>LOGIREF-ZGPU-ZED DESIGN .....</b>	<b>25</b>
5.1	DESIGN CUSTOMIZATION .....	26
5.2	MEMORY LAYOUT.....	28
5.3	RESTORING FULL SOC DESIGN FROM XYLON DELIVERABLES.....	29
<b>6</b>	<b>VIDEO OUTPUT CLOCKING .....</b>	<b>30</b>
6.1	LOGICVC-ML – STANDARD DISPLAY RESOLUTIONS AND PIXEL CLOCK.....	30
6.2	UTILITY CLOCK MODULE .....	31
6.3	LINUX FRAME BUFFER – CHANGING DISPLAY RESOLUTIONS .....	32
<b>7</b>	<b>QUICK START.....</b>	<b>34</b>
7.1	REQUIRED HARDWARE .....	34
7.2	SET UP THE ZEDBOARD FOR USE WITH PRECOMPILED LINUX DEMOS FROM THE SD CARD .....	34
7.3	RUNNING PRECOMPILED DEMOS FROM THE SD CARD IMAGE.....	36
7.3.1	BootUp Menu.....	36
7.3.2	Running 3D Demo Apps .....	37
7.3.3	Running QT Demo Apps.....	37
7.4	CHANGE THE DEMO APPLICATIONS OR DESIGN NEW APPLICATIONS FROM SCRATCH .....	38
7.4.1	Xilinx Development Software .....	38
7.4.2	Set Up Linux System Software Development Tools .....	38
7.4.3	Set Up git Tools.....	38
<b>8</b>	<b>SOFTWARE DOCUMENTATION .....</b>	<b>39</b>

8.1	SOFTWARE INSTRUCTIONS – STANDALONE SOFTWARE .....	39
8.2	SOFTWARE INSTRUCTIONS – LINUX SOFTWARE.....	39
<b>9</b>	<b>REVISION HISTORY .....</b>	<b>40</b>

## 1 INTRODUCTION

Xylon's logicBRICKS library of IP cores optimized for Xilinx All Programmable devices includes several graphics logicBRICKS IP cores for full range implementation of 2D and 3D Graphics Processing Units (GPU) on Xilinx Zynq-7000 All Programmable SoC and FPGAs. This user's manual describes Xylon's logiREF-ZGPU-ZED 2D and 3D GPU reference design for the ZedBoard, a community oriented development kit from Avnet Electronics Marketing. The ZedBoard is an ideal starter platform for designers and students who are interested in exploring and prototyping their application ideas for the new Zynq-7000 All Programmable SoC.



**Figure 1: The ZedBoard Development Kit Running Xylon's 3D Graphics Demo**

(Video clip: <http://www.logicbricks.com/logicBRICKS-IP-Library/Video-Galleries/logicBRICKS-Demos-ZedBoard-Clip.aspx>)

This free and pre-verified logicBRICKS reference design includes evaluation logicBRICKS IP cores and hardware design files prepared for Xilinx Vivado® Design Suite. It also includes the complete Linux OS image, software drivers, demo applications and documentation.

The offered evaluation hardware design is customizable. logicBRICKS IP cores can be setup through Vivado IP Integrator (IPI) to support only required graphics features required by the

application; from small and efficient display control that uses just a fraction of programmable logic in the smallest Z-7010 Zynq-7000 AP SoC device, up to the full 3D accelerated graphics engine.

The provided software drivers and libraries include standard Linux Framebuffer driver, Qt application framework and OpenGL® ES 1.1<sup>1</sup> API.

Standard software drivers enable software developers to work fast and efficiently with popular graphics libraries, widget toolkits and familiar development tools. Xylon also supplies bare-metal software drivers for non-OS use.

Aside from the logicBRICKS software support for the Linux OS, Xylon also provides software drivers for other popular operating systems running on the Zynq-7000 AP SoC: Android™, QNX® and Microsoft® Windows® Embedded Compact. A number of Xilinx partners who provide BSPs (Board Support Package) for different operating systems support Xylon logicBRICKS IP cores for graphics. To learn more about the available software support, please visit:

<http://www.logicbricks.com/logicBRICKS/Reference-logicBRICKS-Design/OS-IP-Core-Support.aspx>.

## 1.1 Design Deliverables

### 1.1.1 Hardware Design Files

- Configuration bitstream file for the programmable logic and the SDK export of the reference design that allows for an immediate start and software changes
- ZedBoard reference design prepared Vivado Design Suite
- Xylon evaluation logicBRICKS IP cores:
  - logiCVC-ML Compact Multilayer Video Controller
  - logiBITBLT Bit Block Transfer 2D Graphics Accelerator
  - logi3D Scalable 3D Graphics Accelerator
  - logiCLK Programmable Clock Generator

### 1.1.2 Software

- logicBRICKS standalone (bare-metal drivers) with driver examples
- Zynq FSBL sources and the Xilinx SDK project – custom version for standalone applications
- Linux Framebuffer driver for the logiCVC-ML IP core (display controller IP core)
- Qt5 XylonQPA plugin for 2D hardware acceleration (logiBITBLT 2D graphics accelerator IP core)
- HMI demo application that uses Qt application framework for GUI capabilities
- logi3D example sources and binaries (OpenGL ES 1.1<sup>1</sup> library for the logi3D IP core may be provided on request)

---

<sup>1</sup> Product is based on a published Khronos specification, and is expected to pass the Khronos Conformance Testing Process. Current conformance status can be found at [www.khronos.org/conformance](http://www.khronos.org/conformance).

### 1.1.3 Binaries

- Precompiled SD Card image for the fastest demo startup
- First Stage Bootloader (FSBL)
- standalone logiCVC-ML and logiBITBLT examples
- Linux binaries:
  - uboot, devicetree (dtb), root file system (uramdisk)
  - ulmage – kernel with the Framebuffer driver for the logiCVC-ML IP core
  - OpenGL ES 1.1 simple example and Xylon 3D demo

## 1.2 Usage Modes

The logiREF-ZGPU-ZED reference design can be used in different ways, which are listed in this paragraph and thoroughly explained throughout this document.

### 1.2.1 Quick Evaluation with no HW and/or SW Changes

- Download and install the logiREF-ZGPU-ZED reference design (see chapter 3 GET AND INSTALL THE REFERENCE DESIGN)
- Setup the demo hardware and use the provided SD card image to run precompiled demo applications (paragraph 7.2 Set Up the ZedBoard for Use with Precompiled Linux Demos From the SD Card)

### 1.2.2 Develop Standalone and Linux Software, no HW Changes

- Download and install the logiREF-ZGPU-ZED reference design (chapter 3 GET AND INSTALL THE REFERENCE DESIGN)
- Setup the demo hardware (paragraph 7.2 Set Up the ZedBoard for Use with Precompiled Linux Demos From the SD Card)
- Use the provided Zynq-7000 AP SoC as it is (binaries)
- Follow instructions for working with logicBRICKS stand-alone (bare-metal) or Linux drivers (please get the full instructions in the *start.html* file from your installation root directory)
- Develop software applications prior to the availability of the actual target system

### 1.2.3 Full GPU Customization, HW and SW Changes

- Download and install the logiREF-ZGPU-ZED reference design (chapter 3 GET AND INSTALL THE REFERENCE DESIGN)
- Setup the demo hardware (paragraph 7.2 Set Up the ZedBoard for Use with Precompiled Linux Demos From the SD Card)
- Obtain logicBRICKS evaluation licenses from Xylon (chapter 4 GETTING LOGICBRICKS EVALUATION LICENSES)
- Use the provided Zynq-7000 AP SoC to add or remove more logicBRICKS IP cores and/or third-party IP cores, or to change logicBRICKS IP settings through the GUI
- Implement new Zynq-7000 AP SoC design

- Develop software by following instructions listed in the *start.html* file from your installation root directory

### 1.3 Xilinx Development Software

The logiREF-ZGPU-ZED reference design and Xylon logicBRICKS IP cores are fully compatible with Vivado Design Suite 2014.4. Future design releases shall be synchronized with the newest Xilinx development tools.

Licensed users of the Xilinx tools can use their existing software installation for the logiREF-ZGPU-ZED evaluation.

### 1.4 Graphics Demo Preview

The logiREF-ZGPU-ZED reference design is functionally identical to the logiREF-ZGPU-ZC702 reference design prepared for Xilinx Zynq-7000 All Programmable SoC ZC702 Development Kit.

Please check Xylon's Video Gallery web pages (<http://www.logicbricks.com/logicBRICKS-IP-Library/Video-Galleries/logicBRICKS-Demos-Xilinx-ZC702-Video-Clip.aspx>) to preview the graphics demo applications provided with the logiREF-ZGPU-ZED installation for your ZedBoard development kit.



**Figure 2: Screenshots from Some Demos Provided with the Reference Design**

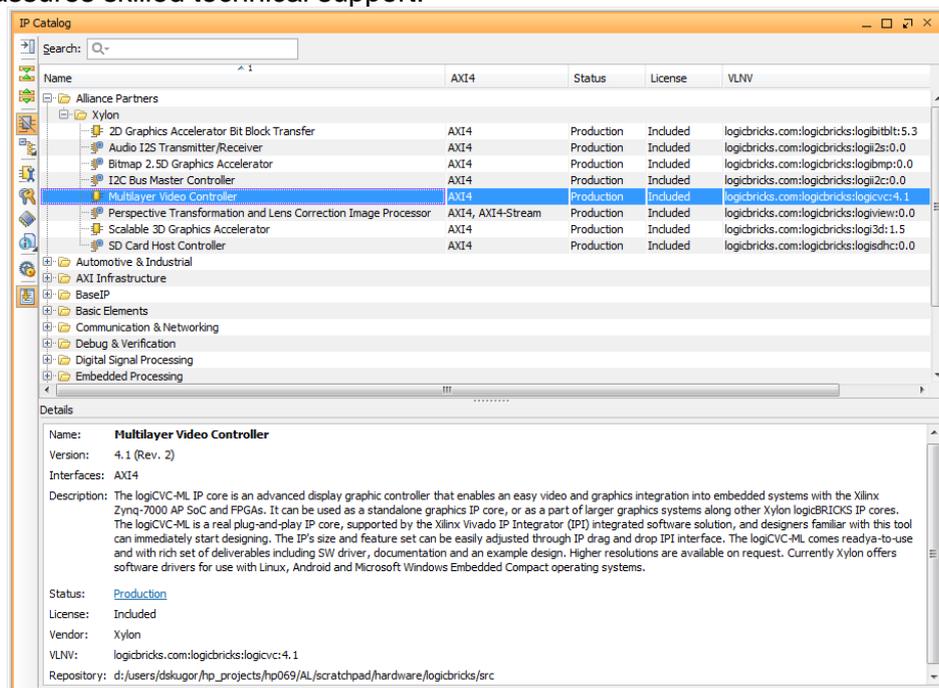
## 2 LOGICBRICKS IP CORES

### 2.1 About logicBRICKS IP Library

Xylon's logicBRICKS IP core library provides IP cores optimized for Xilinx FPGA and Zynq-7000 All Programmable SoC. 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:

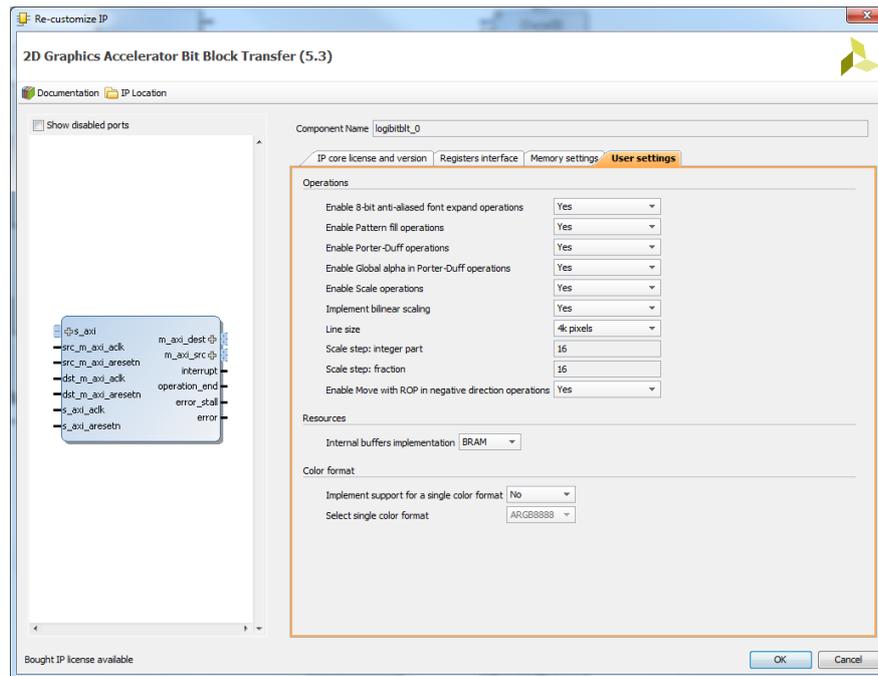
- Compatibility with the Xilinx Vivado and ISE Design Suites – logicBRICKS can be used in same ways as Xilinx IP cores and require no skills beyond general tools knowledge. logicBRICKS users can setup IP core feature sets and programmable logic utilization through Xilinx implementation tools' Graphical 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.



**Figure 3: logicBRICKS IP Cores Imported into the Vivado Catalog**

\* Some of the latest logicBRICKS IP cores are provided in the Vivado compatible version only. Please visit our web site, or contact Xylon to learn more about the tools compatibility of the specific logicBRICKS IP core.

The Figure 3 shows imported logicBRICKS IP cores into Vivado Design Suite, while the Figure 4 shows a typical logicBRICKS IP core's configuration GUI.



**Figure 4: Example of logicBRICKS IP Configuration GUI**

Click on the Documentation icon in the GUI opens the User's Manual of the logicBRICKS IP core!

## 2.2 Evaluation logicBRICKS IP Cores

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

- Import into the Xilinx ISE Platform Studio (XPS) and Vivado Design Suite
- IP parameterization through the tool GUI interface
- Bitstream generation
- If you need to simulate logicBRICKS IP cores, please contact Xylon

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
- 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

- Available for Xilinx Vivado Design Suite and ISE XPS implementation tools

More info: <http://www.logicbricks.com/Products/logiCVC-ML.aspx>

Datasheet: [http://www.logicbricks.com/Documentation/Datasheets/IP/logiCVC-ML\\_hds.pdf](http://www.logicbricks.com/Documentation/Datasheets/IP/logiCVC-ML_hds.pdf)

### 2.3.2 logiBITBLT Block Transfer 2D Graphics Accelerator



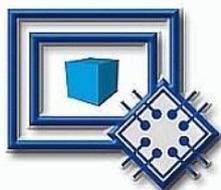
This 2D graphics accelerator speeds up the most common GUI operations and off-loads the processor. The logiBITBLT transfers graphics objects from one to another part of system's on-screen or off-screen video memory, and performs different operations during transfers, such as ROP2 raster operations, bitmap scaling (stretching) and flipping, Porter & Duff compositing rules or transparency.

- Supports Xilinx Zynq-7000 AP SoC and all Xilinx FPGA families
- Available SW drivers for Linux and Microsoft Windows Embedded Compact OS
- Supports move operations, in positive and negative direction
- Supports 16 different ROP2 operations
- Integrated bitmap flipping and optional up/down scaling
- Porter-Duff composition with/without global alpha
- Color-keyed transparency, source and destination
- Anti-aliased 8-bit font expansion
- Pattern fill with 8x8 pixels patterns
- Solid fill with any of the supported color formats
- Supported color formats: RGB8, ARGB8, RGB16, ARGB16, RGB24, and ARGB24
- Control of pixel alpha blending factors
- ARM AMBA AXI4 and AXI4-Lite bus compliant
- Memory layout configurable for big or little endianness
- Available for Xilinx Vivado Design Suite and ISE XPS implementation tools

More info: <http://www.logicbricks.com/Products/logiBITBLT.aspx>

Datasheet: [http://www.logicbricks.com/Documentation/Datasheets/IP/logiBITBLT\\_hds.pdf](http://www.logicbricks.com/Documentation/Datasheets/IP/logiBITBLT_hds.pdf)

### 2.3.3 logi3D Scalable 3D Graphics Accelerator



The logi3D Scalable 3D Graphics Accelerator IP core is a 3D Graphics Processing Unit (GPU) IP core developed for embedded systems based on the Xilinx Zynq-7000 All Programmable SoC.

The IP is designed to support the OpenGL ES 1.1 API specifications – a royalty-free, cross-platform API for full-function 2D and 3D graphics on embedded systems – including consoles, phones, appliances and vehicles.

- Graphics Accelerator IP designed to support the OpenGL ES 1.1 API (Common Profile)
- Conformant to the AMBA AXI4 bus specifications from ARM
- Compatible with popular operating systems: Linux, Android and Microsoft Windows Embedded Compact
- FPGA resource-effective 3D acceleration

- ARM Cortex™ -A9 CPU Core with NEON™ runs the geometry engine and optimizes the IP's size
- The logi3D can be used with different CPUs
- Hardware implemented 3D graphics algorithms:
  - Occlusion culling
  - Gouraud shading
  - MIP-MAP level of the texture per pixel
  - Texture filtering: point sampling, bilinear filtering and trilinear filtering
  - Fog function per vertex
  - Alpha Blending
  - Full Screen Anti-aliasing
- Parametrical VHDL design that allows tuning of slice consumption and features set
- Available for Xilinx Vivado Design Suite and ISE XPS implementation tools

More info: <http://www.logicbricks.com/Products/logi3D.aspx>

Datasheet: [http://www.logicbricks.com/Documentation/Datasheets/IP/logi3D\\_hds.pdf](http://www.logicbricks.com/Documentation/Datasheets/IP/logi3D_hds.pdf)

### 2.3.4 logiCLK Programmable Clock Generator



The logiCLK is a programmable clock generator IP core featuring twelve independent and fully configurable clock outputs. While six clock outputs can be fixed by generic parameters prior to the implementation, the other six clock outputs can be either fixed by generics or dynamically reconfigured in a working device. The Dynamic Reconfiguration Port (DRP) interface gives system designers the ability to change the clock frequency and other clock parameters while the design is running by mean of a set of PLL registers.

- Supports Xilinx Zynq-7000 All Programmable SoC, 7 series and Spartan®-6 FPGAs
- Provides 12 independent clock outputs that can be configured by generic parameters:
  - 6 outputs can be dynamically configured through the DRP interface
  - 6 outputs can be configured by generics only
- Input clock frequency range (depends on the used device's speed grade):
  - Spartan-6: 19 – 540 MHz
  - 7 series: 19 – 1066 MHz
- Output clocks frequency range:
  - Spartan-6: 3.125 – 400 MHz
  - 7 series: 6.25 – 741 MHz
- Configurable ARM AMBA AXI4-Lite and CoreConnect PLBv46 compliant registers interface
- Software support for Linux and Microsoft Windows Embedded Compact operating systems
- Available for Xilinx Vivado Design Suite and ISE XPS implementation tools

More info: <http://www.logicbricks.com/Products/logiCLK.aspx>

Datasheet: [http://www.logicbricks.com/Documentation/Datasheets/IP/logiCLK\\_hds.pdf](http://www.logicbricks.com/Documentation/Datasheets/IP/logiCLK_hds.pdf)

## 2.4 logicBRICKS IP Cores for Video Processing

Xylon offers several logicBRICKS IP cores for video processing on Xilinx Zynq-7000 All Programmable SoC and FPGA programmable devices, which can be used as extensions to Xylon 2D and 3D graphics engines, or as key IP cores for video-only embedded applications.

All logicBRICKS IP cores support ARM AMBA AXI4 on-chip bus and can be easily mixed together, or with Xilinx and third-party IP cores.

### 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](http://www.logicbricks.com/Documentation/Datasheets/IP/logiVIEW_hds.pdf)

### logiWIN Versatile Video Input



Enables easy implementation of video frame grabbers. Input video can be decoded, real-time scaled, de-interlaced, cropped, anti-aliased, positioned on the screen, ...

More info: <http://www.logicbricks.com/Products/logiWIN.aspx>

Datasheet: [http://www.logicbricks.com/Documentation/Datasheets/IP/logiWIN\\_hds.pdf](http://www.logicbricks.com/Documentation/Datasheets/IP/logiWIN_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](http://www.logicbricks.com/Documentation/Datasheets/IP/logiISP_hds.pdf)

## 3 GET AND INSTALL THE REFERENCE DESIGN

Xylon offers several logicBRICKS reference designs for different hardware platforms. Short descriptions of all Xylon logicBRICKS reference designs can be found at:

<http://www.logicbricks.com/logicBRICKS/Reference-logicBRICKS-Design.aspx>.

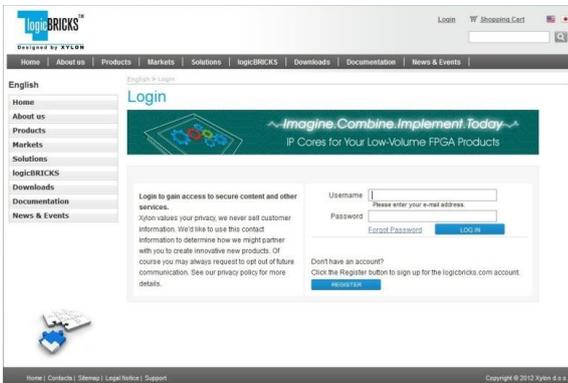
A quick access to specific reference design is also possible through the main downloads navigation page:

<http://www.logicbricks.com/logicBRICKS/Reference-logicBRICKS-Design/Xylon-Reference-Designs-Navigation-Page.aspx>.

Only registered logicBRICKS users can download logicBRICKS reference designs. Unregistered users will be re-directed to the User Login page. The download link is automatically sent by an e-mail, which means that the registration process requires access to the e-mail account. Xylon reference logicBRICKS designs can be downloaded as cross-platform Java JAR self-extracting installers.

### 3.1 Registration Process

Registration is very quick and simple. If you experience any trouble during the registration process, please contact Xylon Technical Support Service – [support@logicbricks.com](mailto:support@logicbricks.com).



**Figure 5: Registration Process – Step 1**

**Step 1**

If you are the registered logicBRICKS user, please type-in your Username and Password. Unregistered users should click on the Register button, which will open the registration form.



**Figure 6: Registration Process – Step 2**

**Step 2**

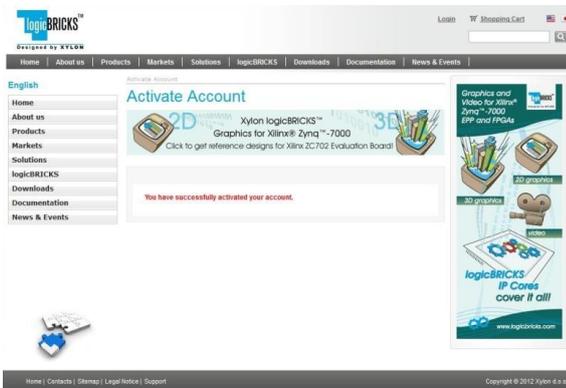
Unregistered users should fill-in the registration form from the Figure 6 . Please take care on required form's fields. Your Username is an actual e-mail account used for communication with Xylon logicBRICKS. Xylon accepts only valid company e-mail accounts.



**Figure 7: Registration Process – Step 3**

**Step 3**

As soon as your registration form gets accepted by Xylon, you get a confirmation message. Please check your e-mail to find a link that activates your logicBRICKS account. If you do not get the confirmation message in several minutes, please check your Spam Filter or Junk Mail directory. If you have not received the confirmation message, please contact Xylon support.



**Figure 8: Registration Process – Step 4**

### Step 4

Click on the logicBRICKS web account activation link in the received e-mail, and you will get the confirmation status message. Please login to proceed.



**Figure 9: Registration Process – Step 5**

### Step 5

As soon as you select an appropriate logicBRICKS reference design and installer for your operating system from the Downloads Navigation Page (link below), you will get an e-mail with the download link for the selected reference design installation.

<http://www.logicbricks.com/logicBRICKS/Reference-logicBRICKS-Design/Xylon-Reference-Designs-Navigation-Page.aspx>

## 3.2 Installation Process

Installation process is quick and easy. Each logicBRICKS reference design 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 self-installing executable to unpack and install the reference design on your PC.

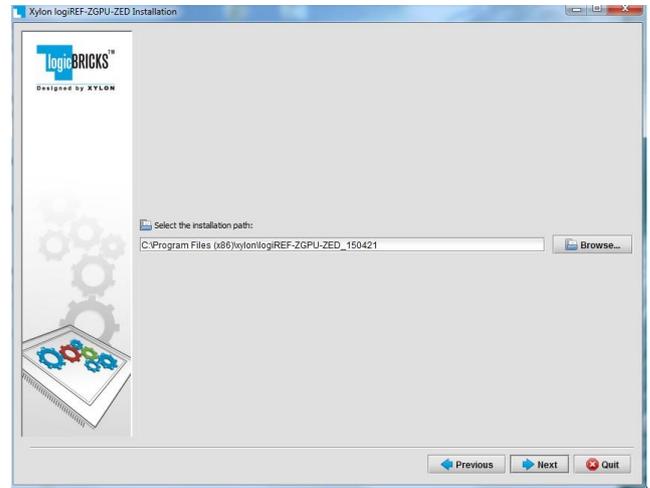
At the beginning, you will be requested to accept the reference design evaluation license – Figure 10. For installation in Linux OS, please follow instructions:

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

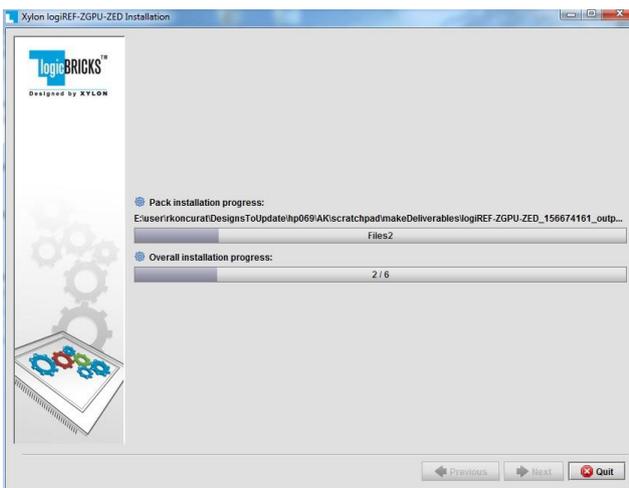
If you agree with the conditions from the evaluation license, click NEXT and select the installation path for your logicBRICKS reference design (Figure 11). The installation process takes several minutes. It generates the directory structure described in the paragraph 3.3 Directory Structure.



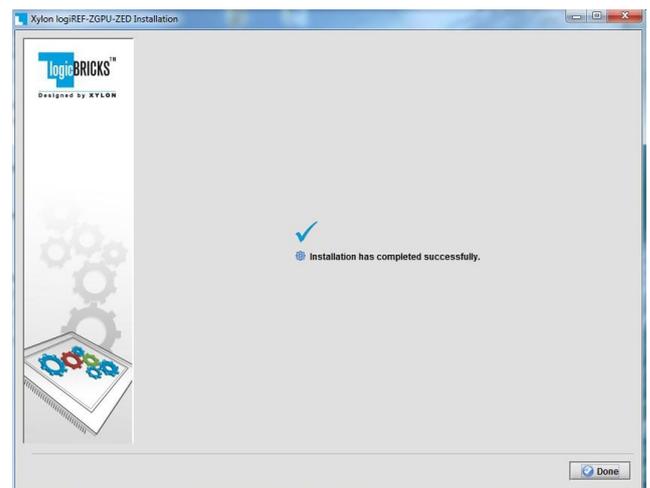
**Figure 10: Installation Process – Step 1**



**Figure 11: Installation Process – Step 2**



**Figure 12: Installation Process – Step 3**



**Figure 13: Installation Process – Step 4**

### 3.2.1 Filesystem Permissions of the Installed Directory (Windows 7)

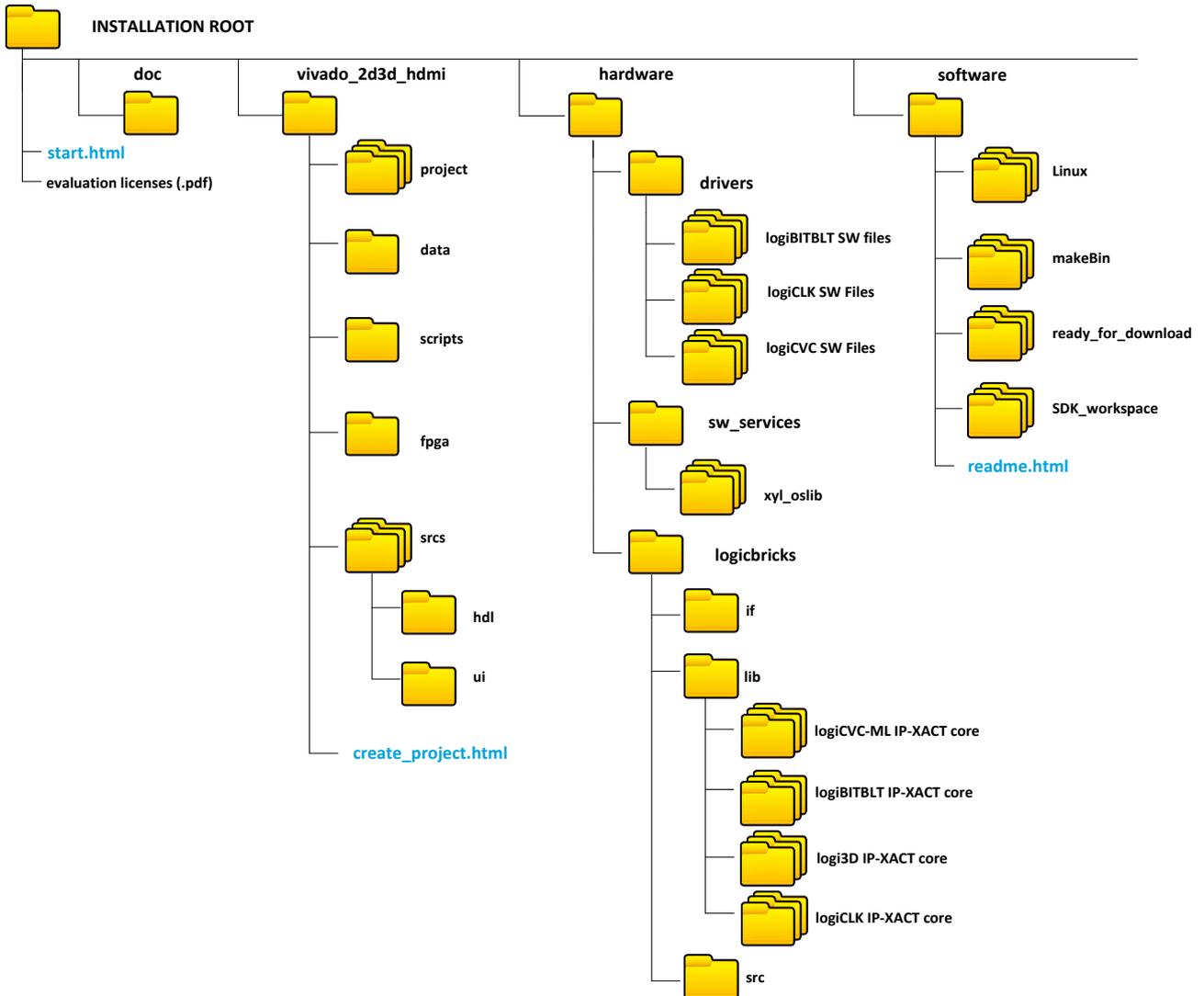
The reference design installed in the default path `C:\Program Files\xylon` will inherit read-only filesystem permissions from the parent directory. This will block you in opening the hardware project file in Xilinx Vivado tools. Therefore it is necessary to change the filesystem permissions for the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

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

### 3.3 Directory Structure

Figure 14 gives a top level view of the directories and files included with the logiREF-ZGPU-ZED reference design for the ZedBoard development kit. Table 1 explains the purpose of directories.

Please use the `start.html` file located in the installation root directory as a jump-start navigation page for exploring the reference design.



**Figure 14: Directory Structure**

Directory	Purpose
INSTALLATION ROOT	This directory contains the <i>start.html</i> page – the jump-start navigation page through the reference design.
doc	Project documentation.
vivado_2d3d_hdmi	This directory contains the complete Vivado project and files necessary for regenerating project from TCL scripts.
data	Design constraints files (XDC).
project	Vivado project related directories.
srcs	Block design GUI script and HDL wrappers.
scripts	TCL scripts to create block design from scratch.
fpga	ZedBoard reference design bitstream.
create_project.html	Instructions for building Vivado project from scratch.
hardware	
drivers	Standalone (bare-metal) drivers for logicBRICKS IP cores with documentation and examples.
logicbricks/lib	Xylon custom IP core interfaces (bus definitions).
logicbricks/src	Evaluation logicBRICKS IP-XACT cores (zip archives).
	Evaluation logicBRICKS IP-XACT extracted IP cores. IP cores' User's Manuals are stored in <i>doc</i> subdirectories.
sw_services	<i>xyl_oslib</i> – Xylon OS abstraction library for Xilinx Xilkernel embedded kernel – use in standalone (non-OS) applications.
software	
readme.html	Navigation page through the software files and instructions for building binaries.
Linux/kernel	Linux kernel and device tree configuration files.
Linux/libraries	Qt5 XylonQPA plugin for 2D acceleration, Qt application HDMI demo, logi3D demo.
makeBin	Utility script for creating boot.bin file.
ready_for_download	Prepared binaries ready for download.
SDK_workspace	Xilinx SDK workspace folder for building bare-metal applications.

**Table 1: Explanation of Directories in logiREF-ZGPU-ZED Reference Design**

## 4 GETTING LOGICBRICKS EVALUATION LICENSES

Please note that the logiREF-ZGPU-ZED reference design installation provides you with everything needed to run the provided demo applications or to use/change the provided software source code. However, if you wish to make any changes on the hardware design files, such as to remove, add or reconfigure some of the provided IP cores, you have to obtain evaluation IP licenses from Xylon.

The following pages describe the procedure for getting and licensing evaluation logicBRICKS IP cores that takes several minutes to complete. If you experience any trouble during this process, please contact Xylon Technical Support Service – [support@logicbricks.com](mailto:support@logicbricks.com).

You must be logged in to the Xylon website using your logicBRICKS user name and password to get an access to evaluation logicBRICKS IP cores. Unregistered users will be re-directed to the User Login page. Paragraph 3.1 Registration Process explains this simple registration procedure.

**Step 1** – Logged in users get the “My logicBRICKS” tab in the main [www.logicbricks.com](http://www.logicbricks.com) navigation menu. Click on it, and you will be directed to your main web page for communication with Xylon logicBRICKS – Figure 15. Please select the “Request Eval IP Core” tab in the left menu.



Figure 15: Step 1 – My logicBRICKS Navigation Page

**Step 2** – Select the evaluation logicBRICKS IP core and click on “Obtain evaluation license key” link (Figure 16). If you are entitled to get the evaluation logicBRICKS IP core, you will be immediately asked your Ethernet MAC ID number or Sun Host ID – as described in the Step 3 (Figure 19).

If the evaluation logicBRICKS IP cores’ list looks differently from the one shown on Figure 16, for example as the list presented by the Figure 17, please fill in and submit the request form (Figure 18), and allow us some time to process your request. Scroll down to get to the request form.

For instructions how to find your Ethernet MAC or host ID, please visit:

<http://www.logicbricks.com/Documentation/Article.aspx?articleID=KBA-01186-M0JXKD>.



**Figure 16: Step 2 – Selecting logicBRICKS IP Core for Licensing**

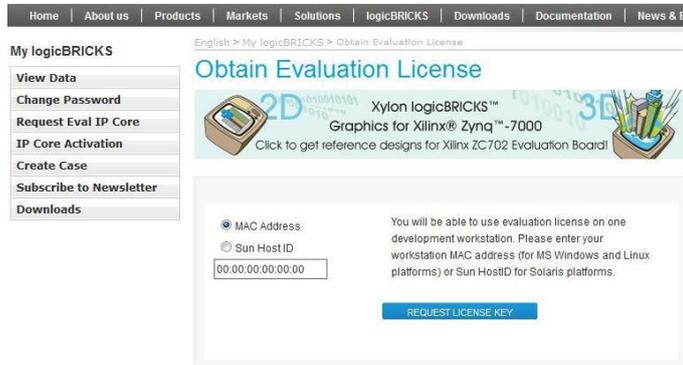


**Figure 17: Step 2 – A List of Already Activated logicBRICKS IP Licenses**



**Figure 18: Step 1 – Licensing logicBRICKS Evaluation IP Cores**

**Step 3** – Evaluation logicBRICKS IP licenses are tied to your Ethernet MAC address or Sun Host ID (Figure 19), and can be used on a single working station only. Fill in this address and click on the **“Request License Key”** button. You should get the confirmation message (Figure 20). If you do not get the confirmation message, please contact Xylon technical support – [support@logicbricks.com](mailto:support@logicbricks.com).

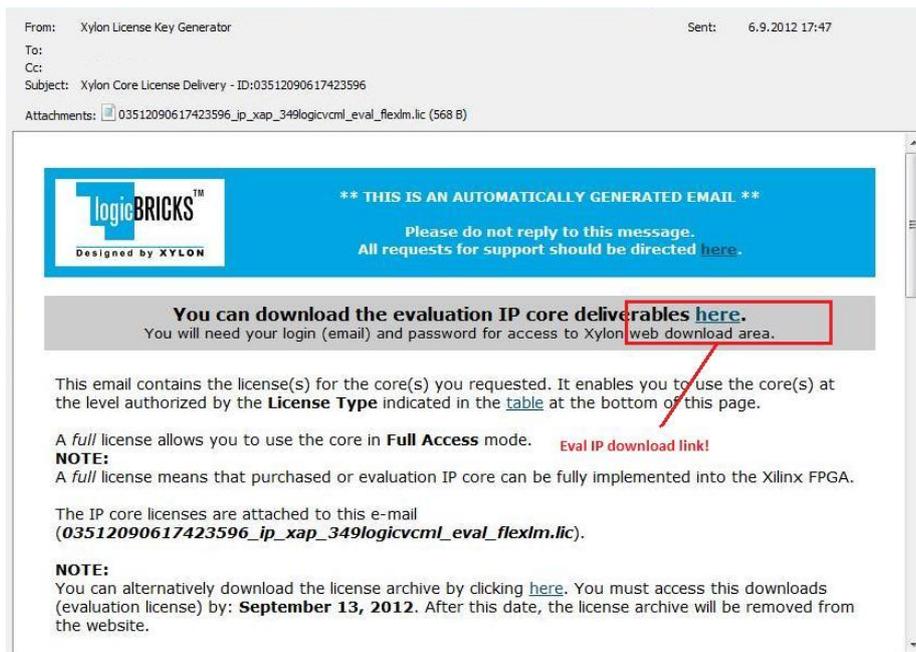


**Figure 19: Step 3 – Licensing logicBRICKS Evaluation IP Cores**



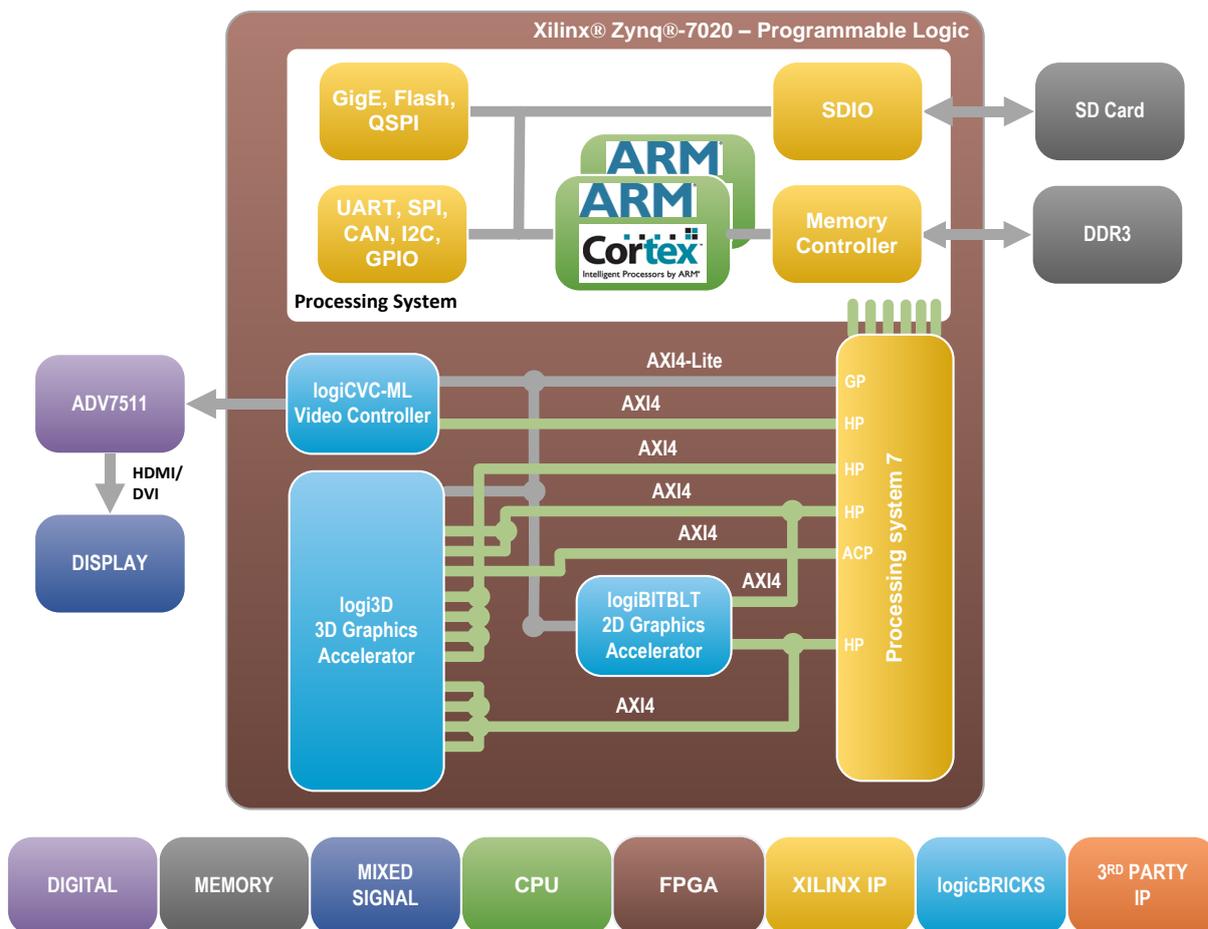
**Figure 20: Step 3 – Confirmation Message**

**Step 4** – You will get an e-mail with the license key (file) and full instructions for setting up the license key and downloading the logicBRICKS IP core. Please follow the provided instructions.



**Figure 21: Step 4 – E-mail with logicBRICKS License and Download Instructions**

## 5 LOGIREF-ZGPU-ZED DESIGN



**Figure 22: logiREF-ZGPU-ZED Reference Design Block Diagram**

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

Figure 22 shows the simplified block diagram of the logiREF-ZGPU-ZED reference design. Design clocking structures are not shown. Please read the chapter 6 VIDEO OUTPUT CLOCKING and explore the design files to understand the clocking structure.

Graphics features like bitmaps operations, alpha blending, overlays, rotations, scaling, LCD display control, 3D rendering, etc., are supported by three separated logicBRICKS IP cores.

The logiCVC-ML Compact Multilayer Video Controller IP core is an advanced display graphics controller for LCD and CRT displays, which enables an easy video and graphics integration into embedded systems based on Xilinx programmable technology. Though its main function is to provide flexible display control, it also includes a level of hardware acceleration: alpha blending, panning, buffering of multiple frames, etc. The logiCVC-ML IP core can directly drive a common PC monitor

through the ADV7511 – High-Definition Multimedia Interface (HDMI®) transmitter available on the ZedBoard.

The logiREF-ZGPU-ZED GPU reference design includes the 2D graphics accelerator that off-loads the ARM dual-core Cortex-A9 Core processing system and increases graphics performance. The logiBITBLT IP core supports: fast copying, moving, up and down scaling, image flipping, alpha blending and Porter & Duff compositing operations between different graphics objects.

For 3D graphic acceleration there is the logi3D Scalable 3D Graphics Accelerator IP core designed to support the OpenGL ES 1.1 API – a royalty-free, cross-platform API for full-function 2D and 3D graphics on embedded systems – including consoles, phones, appliances and vehicles.

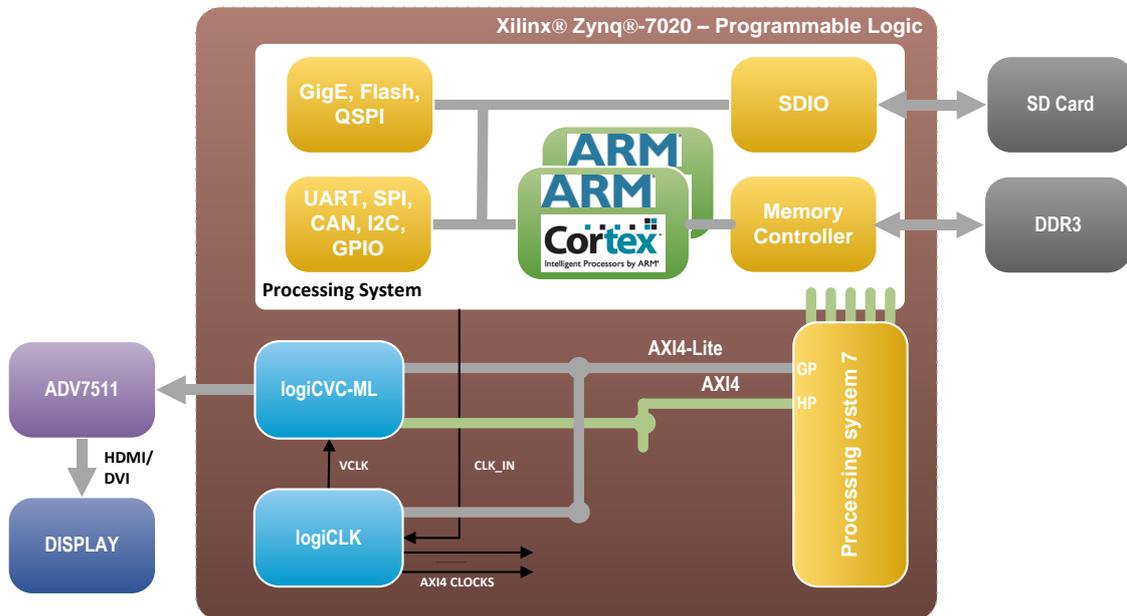
The memory subsystem is an essential part of any graphics based system. It must ensure enough storage space for GUI elements and application code and a fast interface to assure enough memory bandwidth for a flicker-free display output. The ZedBoard includes two 16-bit DDR3 memories connected as one 512MB 32-bit memory module. The memory is connected to the hard memory controller in the Zynq-7000 AP SoC Processor Subsystem (PS).

## 5.1 Design Customization

The provided reference design can be customized in different ways. Please note that any changes in the provided reference design require evaluation IP licenses for logicBRICKS IP cores. The licensing process is described in the chapter 4 GETTING LOGICBRICKS EVALUATION LICENSES.

Possible design changes include:

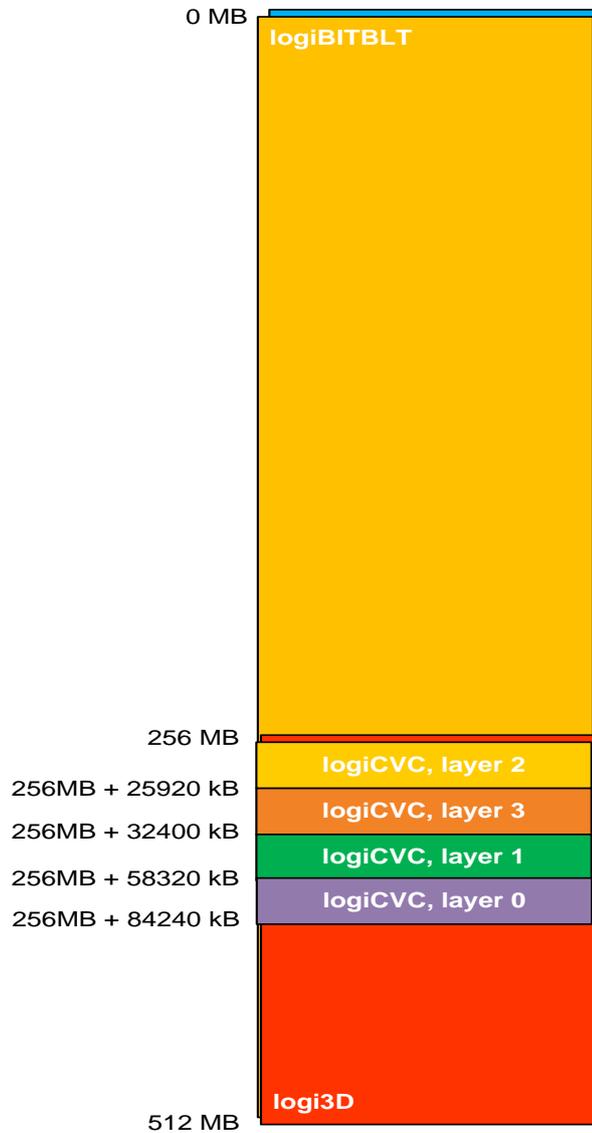
- Change logicBRICKS IP settings, i.e. change number of graphics layers controlled by the logiCVC-ML display controller IP core
- Remove some logicBRICKS IP cores, i.e. remove all graphics accelerators and use only the logiCVC-ML display controller IP core, or remove the 3D acceleration and work with the 2D graphics only, etc.
- Add more instances of logicBRICKS IP cores, i.e. add second logiCVC-ML IP core and drive two displays with different graphics content, add logiWIN IP core and process the video...
- Add your own or third-party IP cores to various combinations of logicBRICKS IP cores
- ...



**Figure 23: A Minimal Zynq-7000 AP SoC Display Controller**

Figure 23 shows an example architecture featuring only the logiCVC-ML display controller IP core. Such a configuration provides no graphics acceleration in the programmable logic and all graphic contents must be fully drawn by the Processing System (PS). The consumption of programmable logic resources is minimal. The Figure 27 shows a clocking structure detail – please see the chapter 6 VIDEO OUTPUT CLOCKING.

## 5.2 Memory Layout



**Figure 24: logiREF-ZGPU-ZED Memory Layout**

IP Core	Memory Access [MB]	Memory Stride [pixels]	Display Memory [HxV pixels]
logiCVC	256 – 511	2048	up to 1920x1080
logiBITBLT	0 – 511	2048	-
logi3D	256 – 511	2048	-

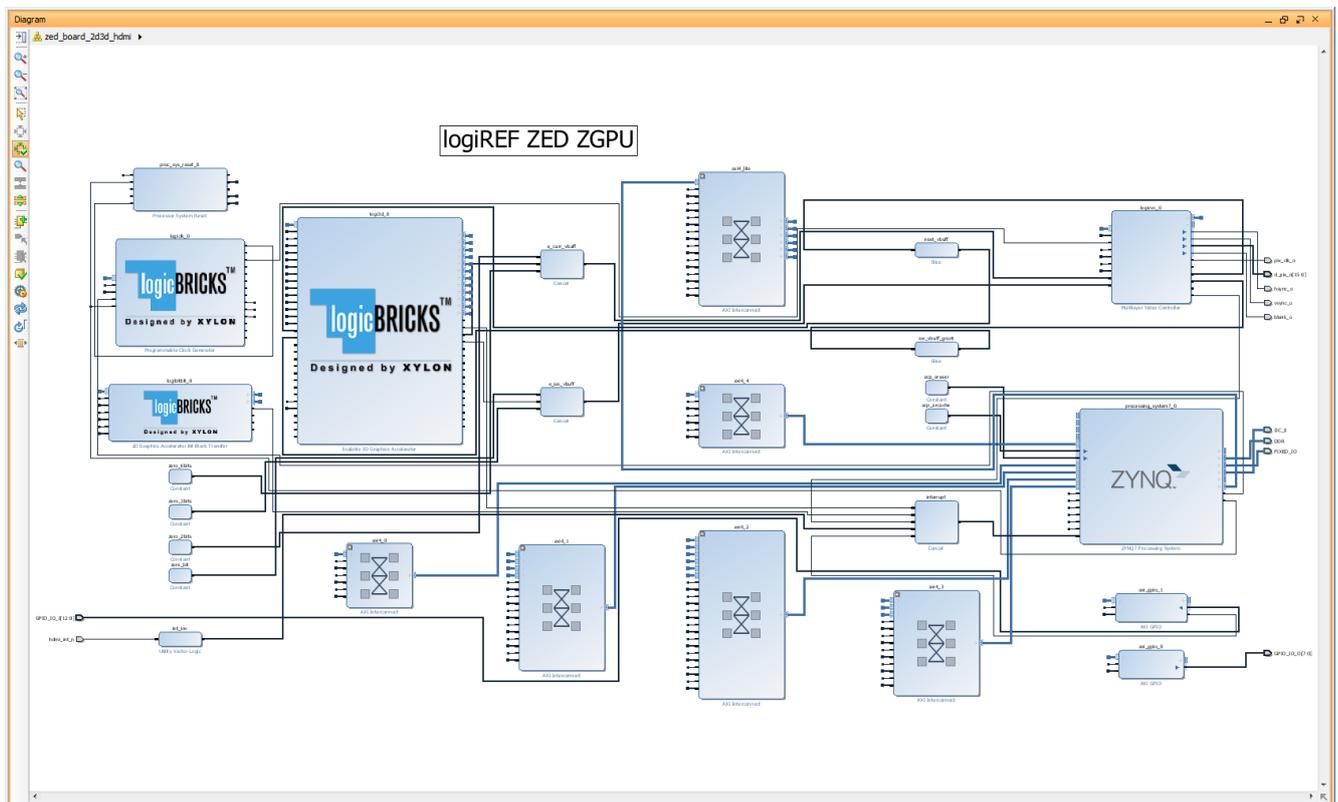
**Table 2: logicBRICKS IP Cores' Memory Addressing**

### 5.3 Restoring Full SoC Design from Xylon Deliverables

Xylon provides all necessary design files and TCL scripts to enable full Vivado Design Suite 2014.4 project restore. Full guidelines can be found in the `vivado_2d3d_hdmi/create_project.html` file.

The `vivado_2d3d_hdmi.xpr` file from your installation directory is the Vivado Design Suite project file that opens the project. Open this file with the Vivado and explore the design (Figure 25).

In order to re-implement or change the provided reference design, please go to Xylon's web site [www.logicbricks.com](http://www.logicbricks.com) and acquire evaluation licenses for the logicBRICKS IP cores (see chapter 4 GETTING LOGICBRICKS EVALUATION LICENSES).



**Figure 25: Vivado Block Design Diagram – logiREF-ZGPU-ZED Project**

To access logicBRICKS IP core user's manuals, double-click on the specific IP core to open the GUI, and click on the Documentation icon to open the document.

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

## 6 VIDEO OUTPUT CLOCKING

Xylon's standard logiCVC-ML Compact Multilayer Video Controller IP core supports display resolutions up to 2048 x 2048. For information about support for higher display resolutions, please contact Xylon at [info@logicbricks.com](mailto:info@logicbricks.com).

The logiREF-ZGPU-ZED reference design demonstrates the logiCVC-ML IP core implemented in Zynq-7000 AP SoC programmable logic. The logiCVC-ML display controller drives inputs into an Analog Devices ADV7511 HDMI Transmitter, which is configured to display the 1080p60 (resolution 1920x1080, 16-bit, YCbCr 4:2:2) image on a regular PC monitor. The ADV7511 HDMI Transmitter driver is integrated with the Xylon Framebuffer software driver for the Linux OS. ADV7511 HDMI Transmitter driver is also provided for the bare-metal applications – see the `software\SDK_workspace\zed_board_init` application.

Xylon logicBRICKS IP cores and provided software can be used in many different hardware setups, and with many different display types. Therefore, in order to be able to fully utilize the graphics provided with the reference design for the ZedBoard and to properly use logicBRICKS products in other hardware setups, designers should understand the video clocking scheme implemented in the logiREF-ZGPU-ZED reference design.

### 6.1 logiCVC-ML – Standard Display Resolutions and Pixel Clock

For full information about setting up the display interface controlled by the logiCVC-ML Compact Multilayer Video Controller IP core, please refer to the logiCVC-ML User's Manual. This chapter focuses on the pixel clock generation and control, since it depends on the overall system's architecture to a great extent.

Table 3 shows required pixel clock's frequencies for several popular display resolutions. Properly implemented display interface must respect the expected display signals' timings, which are based on the requested pixel clock. Wrong pixel clock causes wrong timings on the display interface and, as a consequence, wrong or missing picture on the display. It is visible from the table that graphic controller must be able to source different pixel clocks in order to support multiple display resolutions.

Resolution (@60Hz)	Pixel Clock (MHz)
VGA (640x480)	25.25
480p (720x480)	27
WVGA (854x480)	32
SVGA (800x600)	40
XGA (1024x768)	65
WXGA (1280x768)	68.25
HD 720p (1280x720)	74.25
HD 1080i (1920x1080)	74.25
SXGA (1280x1024)	108
HD1080p (1920x1080)	148.5

**Table 3: Pixel Clock – Common Video Resolutions**



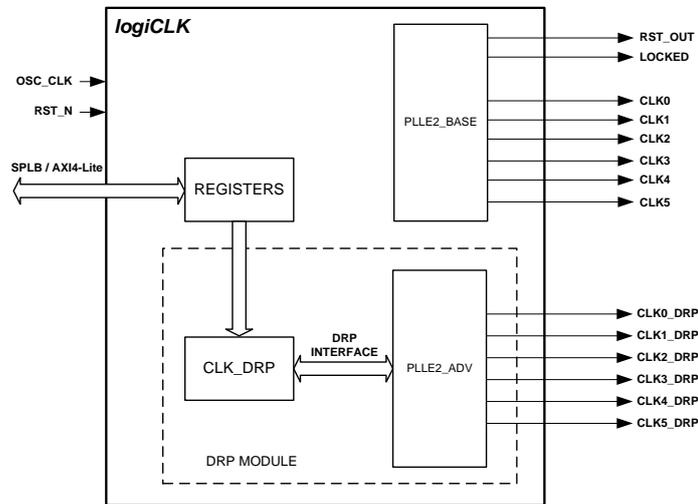


Figure 27: logiCLK Architecture

The logiCLK clock generator IP core is designed to provide frequency synthesis, clock network de-skew, and jitter reduction. It has twelve independent fully configurable clock outputs. Six clock outputs are dynamically reconfigurable by mean of the Dynamic Reconfiguration Port (DRP). The DRP gives the system designer access to the configuration and status registers of the PLL, and behaves like a set of memory-mapped registers.

One dynamically configurable clock output is connected directly to the logiCVC-ML IP core's VCLK input port. It means that the VCLK clock frequency can be changed while design is running, and consequently the pixel clock PIX\_CLK dynamically changes.

When the Xylon Linux Framebuffer driver is configured to use the logiCLK as a pixel clock generator, the required pixel clock frequency is automatically adjusted for a particular video resolution through common clock framework connected to logiCLK CCF Linux device driver. Desired pixel clock frequency is determined from the video mode and set accordingly.

To learn more about this IP core, please read the datasheet:

[http://www.logicbricks.com/Documentation/Datasheets/IP/logiCLK\\_hds.pdf](http://www.logicbricks.com/Documentation/Datasheets/IP/logiCLK_hds.pdf).

## 6.3 Linux Frame Buffer – Changing Display Resolutions

Linux Framebuffer is a standard Linux driver that abstracts the graphics hardware and allows application software to access it through a well-defined interface. Software designers can use it with no need to know anything about the underlying hardware (IP cores) in Xilinx Zynq-7000 All Programmable SoC or FPGA device. The Linux Framebuffer delivered with the logiREF-ZGPU-ZED reference design is adopted by Xylon to fully support the logiCVC-ML display controller IP core. Xylon Framebuffer driver is located in Xilinx Linux Kernel github repository Latest Framebuffer driver version is provided in this installation, for instructions how to configure driver and build the Linux kernel, please check `software/Linux/kernel/readme.txt`, or download User's manual document:

<http://www.logicbricks.com/Documentation/Datasheets/SW/Xylon-Linux-FrameBuffer.pdf>.

Note that Xylon provides Device Tree Source (`dts`) file with IP core configuration information specific to this reference design. When using the Linux kernel with this reference design, user must use the Xylon `dts/dtb` files, located in the `software/Linux/kernel/linux-xlnx-v2014.2` directory, instead of the ones provided by Xilinx. Figure 28 shows an excerpt from Xylon `dts` file.

```
logicvc_0: logicvc@40030000 {
    compatible = "xylon,logicvc-4.00.a";
    reg = <0x40030000 0x6000>;
    interrupt-parent = <&ps7_scugic_0>;
    interrupts = <0 57 0>;
    background-layer-bits-per-pixel = <32>;
    background-layer-type = "rgb";
    hsync-active-low;
    vsync-active-low;
    size-position;
    pixel-stride = <2048>;
    layer_0 {
        address = <0x138F4000>;
        buffer-offset = <1080>;
        bits-per-pixel = <32>;
        type = "rgb";
        transparency = "pixel";
    };
    layer_1 {
        address = <0x11FA4000>;
        buffer-offset = <1080>;
        bits-per-pixel = <32>;
        type = "rgb";
        transparency = "layer";
    };
    layer_2 {
        address = <0x10000000>;
        buffer-offset = <1080>;
        bits-per-pixel = <32>;
        type = "rgb";
        transparency = "layer";
    };
    layer_3 {
        address = <0x11950000>;
```

**Figure 28: Video Mode Definitions – An Excerpt from the Linux `.dts`**

## 7 QUICK START

### 7.1 Required Hardware

A full evaluation\* of the provided reference design requires:

- ZedBoard development kit\*\* from Avnet Electronics Marketing
- HDMI or HDMI to DVI video cable (for 1024x768 capable monitor)
- SD card (min 256MB)
- **optional:** A keyboard (USB Micro-B cable), or a control serial link (USB UART or Ethernet) between PC and the ZedBoard
- **optional:** USB Micro-B cable, USB hub, mouse and keyboard
- **optional:** USB Micro-B cable for debug UART
- **optional:** Ethernet cable for Telnet connection
- **optional:** Xilinx JTAG Parallel cable USB for standalone application development

\* The majority of provided demo applications can be controlled by on-board push-buttons and do not require keyboard and mouse connections

\*\*The reference design has been tested on the ZedBoard, Rev. C

### 7.2 Set Up the ZedBoard for Use with Precompiled Linux Demos from the SD Card

Xylon provides Linux, Qt and 3D demo binaries in the `software/ready-for-download/linux_sd` directory of the delivery. If you want to run prepared demos, copy the contents of the `linux_sd` directory to the root directory on the FAT32 formatted SD card, setup the ZedBoard (Figure 29), plug-in the SD card with precompiled demos and run.

Note: there should be no `linux_sd` directory on the SD card, but only the contents of that directory.

The precompiled Linux graphics demo applications can be launched and controlled by on-board push-buttons or mouse and keyboard combination. Qt graphics demos require use of the mouse to interact with the application and cannot be fully controlled by the push-buttons.

Set up your ZedBoard as shown on the Figure 29 (Table 4):

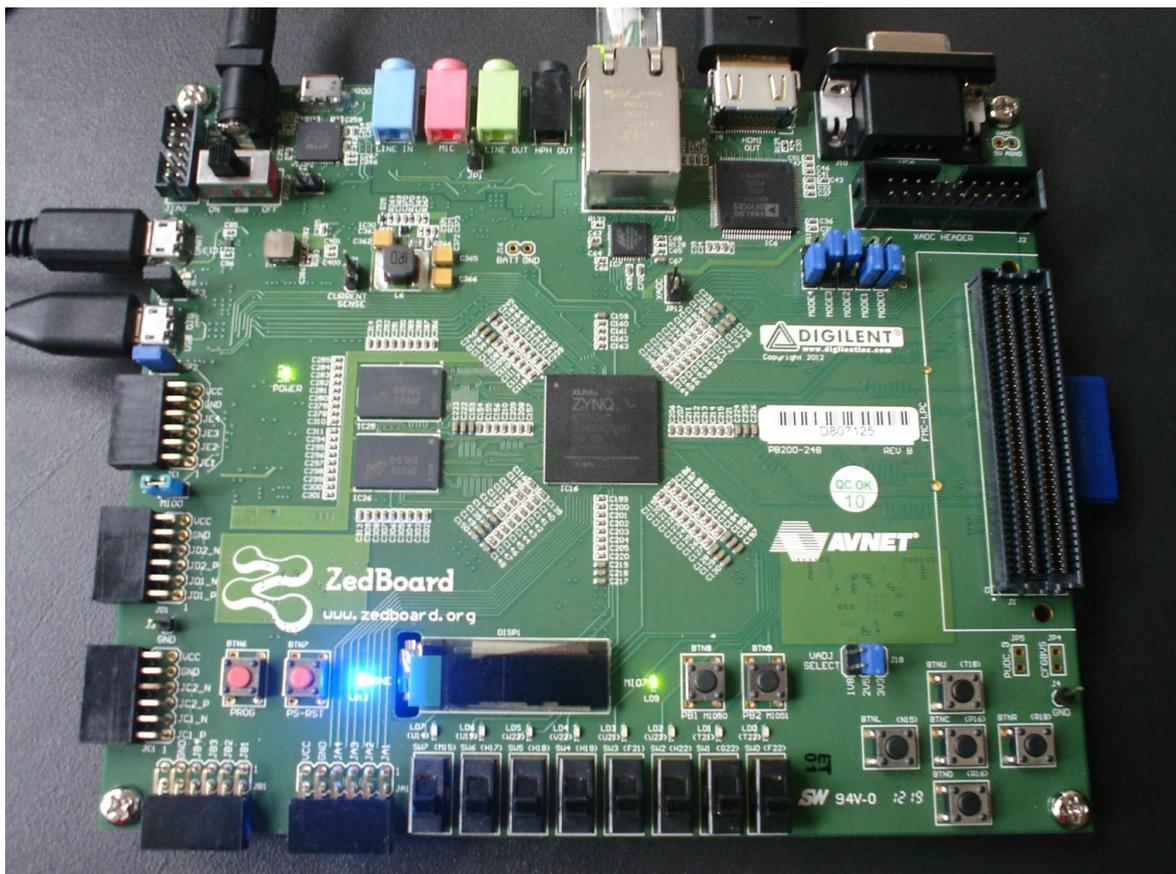
- set the jumpers (Table 4)
- insert the SD card in J12 slot
- connect the power supply and the video cable (HDMI) and optional control devices: serial cable to the USB UART or Ethernet cable
- optionally connect mouse and keyboard to the USB (OTG)

Jumpers settings for the SD boot mode:

Jumper	Name	Setting
JP7	MODE0	GND
JP8	MODE1	GND
JP9	MODE2	3V3
JP10	MODE3	3V3
JP11	MODE4	GND

Jumper	Name	Setting
JP6	MI00	Short
JP2	Vbus 5V enable	Short

**Table 4: Jumpers Set Up for Booting from the SD Card**



**Figure 29: The ZedBoard (Rev. C) Setup for the logicBRICKS Reference Design**

For full explanation of the ZedBoard's features and settings, please check the documentation provided with the kit and visit <http://zedboard.org/support/documentation/1521>.

## 7.3 Running Precompiled Demos from the SD Card Image

To quickly start precompiled Linux demos, make sure that you have the SD card with the precompiled image plugged in the board's slot, and all jumpers setup as described in the previous paragraph.

To control the precompiled demos, you can use:

- four user push-buttons (out of five available) placed in the shape of a cross on the ZedBoard (presuming that HDMI port is located on the North side; use North and South buttons for iterations through examples; West button for starting the selected example; East button for exiting the example)
- the keyboard connected to the micro-USB port
  - in graphical menu use – up and down arrows for iteration over examples; enter key for starting selected example; Q letter key for exiting the example
  - in console (outside of graphical menu) – use whole keyboard to write commands directly to the screen console
- the mouse connected to the micro-USB port for interacting with Qt examples
- serial terminal program (baud rate 115200 8N1) and USB UART connection to the ZedBoard
- telnet connection and Ethernet connection with the ZedBoard

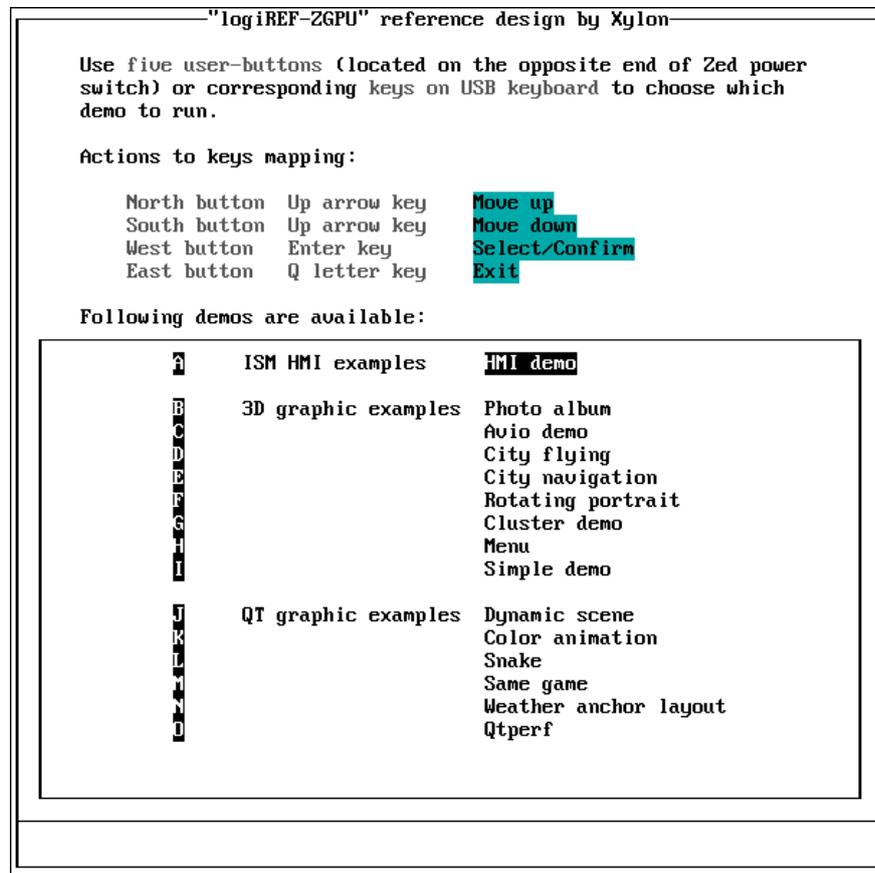
### 7.3.1 BootUp Menu

Your ZedBoard will eventually boot up to the graphical menu from which you can start different demo applications or switch to the Linux terminal. To enter the graphical menu from the Linux console run the following command in the / (root of the filesystem) Linux directory:

```
source .profile
```

If you want to prevent ZedBoard from entering graphical menu on the startup, please rename `config_profile` file on the SD card to any other name. Please do not rename or remove `init.sh` file from the SD card since it performs startup initializations required for applications to run properly.

Please note that graphical menu shows up only on the console on the monitor attached to your ZedBoard kit. If you control the ZedBoard kit by a PC through a serial UART link, the graphical menu is not available.



**Figure 30: logiREF-ZGPU-ZED Boot-Up Graphics Menu**

### 7.3.2 Running 3D Demo Apps

In the screen console, type in:

```
/mnt/3dDemo.elf -a # starts Photo album demo
/mnt/3dDemo.elf -b # starts Avionics demo
/mnt/3dDemo.elf -c # starts City flying demo
/mnt/3dDemo.elf -f # starts Auto Instrument Cluster demo
```

These commands start the demo. Control demos by keys A or D (left-right), E (select), and Q (quit). In some demos you can also use keys W, X, R and F. CTRL+C key combination stops the demo.

### 7.3.3 Running QT Demo Apps

In the screen console, type in:

```
/opt/qt/examples/declarative/demos/snake/snake $ABB #runs with xylongpa
/opt/qt/examples/declarative/demos/snake/snake $AFB #runs with linuxfb
```

```
Note: $ABB and $AFB are defined in config_profile as:  
export ABB="-platform xylonqpa:fb="/dev/fb3" -plugin evdevmouse -plugin  
evdevkeyboard"  
export AFB="-platform linuxfb:fb="/dev/fb3" -plugin evdevmouse -plugin  
evdevkeyboard"
```

## 7.4 Change the Demo Applications or Design New Applications from Scratch

### 7.4.1 Xilinx Development Software

The logiREF-ZGPU-ZED reference design and Xylon logicBRICKS IP cores are fully compatible with Xilinx development tools – Vivado Design Suite 2014.4. 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 logiREF-ZGPU-ZED evaluation and modifications.

### 7.4.2 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.

### 7.4.3 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>.

## 8 SOFTWARE DOCUMENTATION

Please use the `start.html` file, which is located in your logiREF-ZGPU-ZED installation directory (section 4 Software Documentation), or open directly `software/readme.html` file to find relevant documentation for using the logiREF-ZGPU-ZED software deliverables. This file contains links to software documents and instructions related to:

- Standalone (Bare-Metal) software
- Linux software

### 8.1 Software Instructions – Standalone Software

- FSBL instructions
- Standalone software drivers (code and documentation) and examples
- `zed_board_init` application, HDMI initialization and pixel clock setting
- Building standalone applications
- Running standalone applications with the ZedBoard setup for standalone applications

### 8.2 Software Instructions – Linux Software

Xylon provides the Linux Framebuffer driver, Qt5 XylonQPA plugin, Qt HMI example and OpenGL ES 1.1 library/driver for Linux. Zynq toolchain, Linux kernel and file system used for development and demonstrations of Xylon drivers are provisions of Xylon.

- Linux Kernel building instructions, and `dts` files
- Qt5 XylonQPA plugin; general information and building instructions
- Xylon 3D graphics acceleration library; binaries, instructions for building 3D applications, code examples
- Running Linux applications with the ZedBoard setup for the precompiled SD card image

## 9 REVISION HISTORY

Version	Date	Author	Approved by	Note
1.00.a	June 15 <sup>th</sup> , 2012	Z. Šafaržik	Z Šafaržik	Initial
1.00.b	September 3 <sup>rd</sup> , 2012	G. Galić	G. Galić	Added description of the registration, installation and evaluation procedures, directory structure's description
1.00.c	September 19 <sup>th</sup> , 2012	G. Pantar	G. Pantar	SW descriptions updated
2.01.a	May 05 <sup>th</sup> , 2014	D. Škugor	R. Končurat	Changes introduced regarding reference design porting to Vivado User's manual corrections performed
2.01.b	July 24 <sup>th</sup> , 2014	Z. Plesivčak	R. Končurat	Added chapter dealing with read-only permissions of the install directory. Changed instructions on copying Linux binaries to the SD card. Other minor updates and fixes.
2.02.a	April 8 <sup>th</sup> , 2015	D. Škugor	R. Končurat	Changes regarding to updated reference design IP core versions and Vivado version. Removed DirectFB reference. Added XylonQPA reference. Other minor updates and fixes.