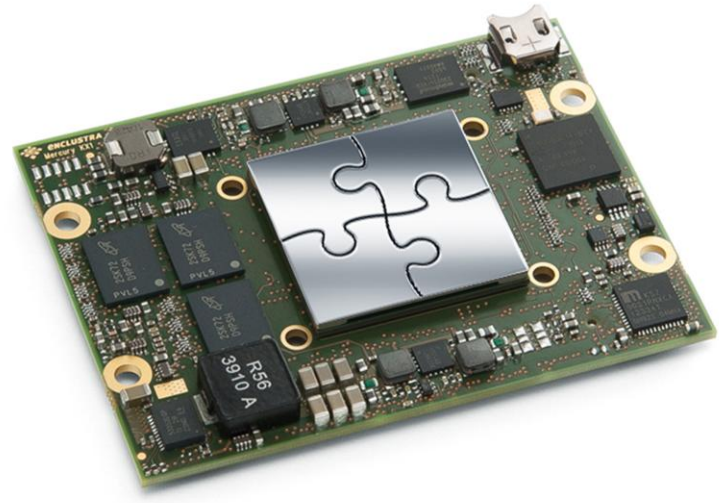
ENCLUSTRA FPGA SOLUTIONS

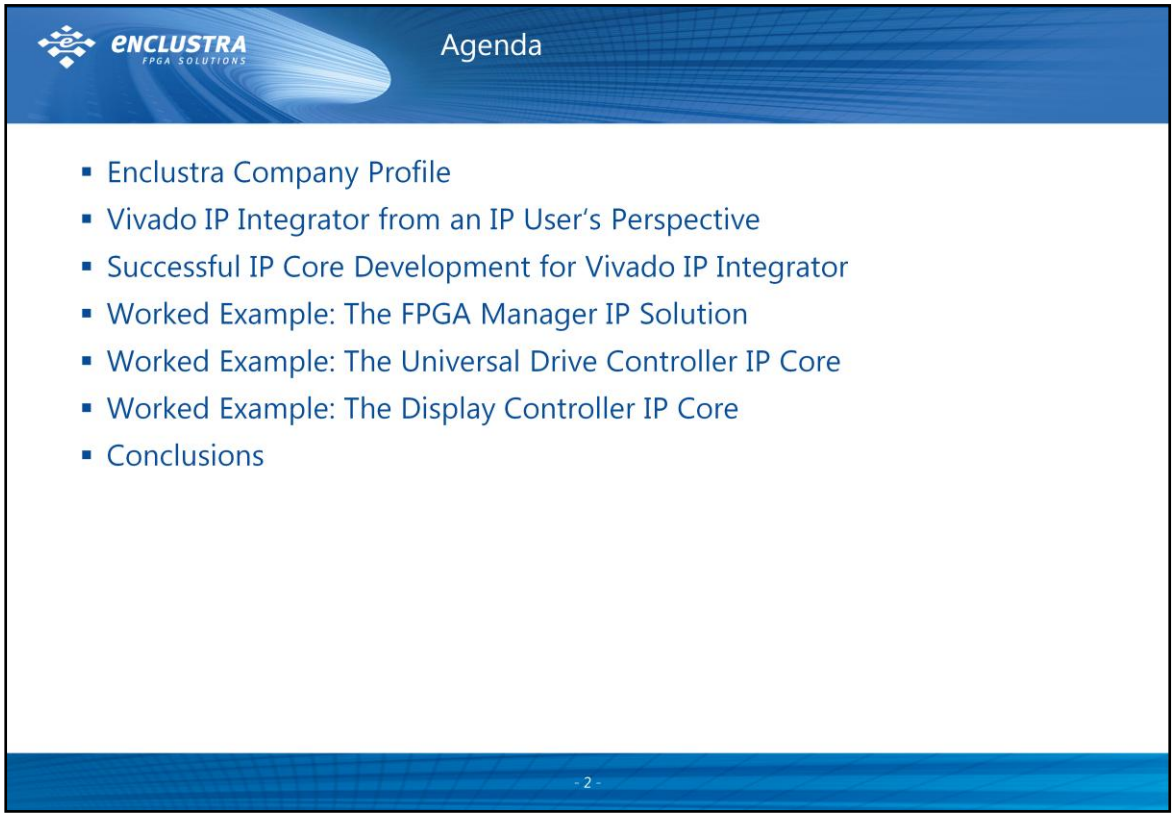
Successful Development of IP Cores for Vivado™ IP Integrator

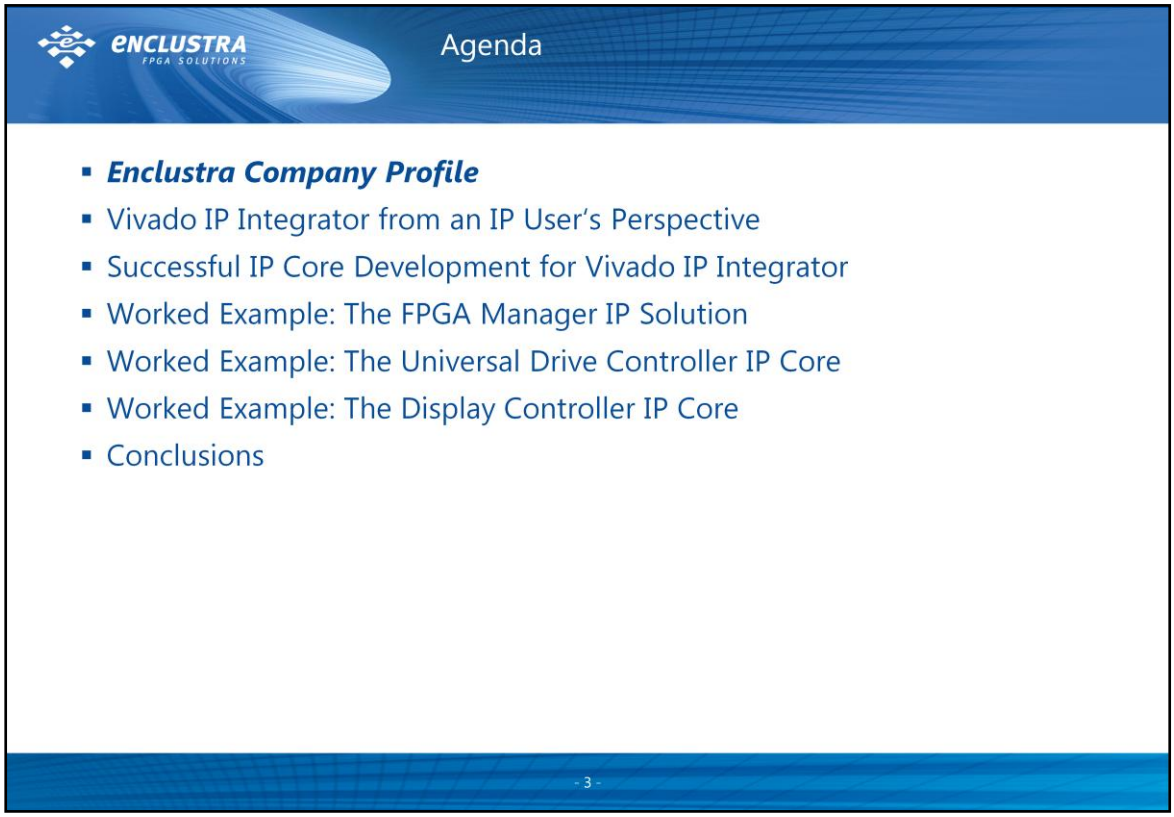


Club Vivado Users Group

Stuttgart, November 12, 2014

Martin Heimlicher
President & Founder
Enclustra GmbH


ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 2 -


ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 3 -

Short Profile

Enclustra is a dynamic, innovative and successful FPGA design service company located in Technopark Zurich, Switzerland.

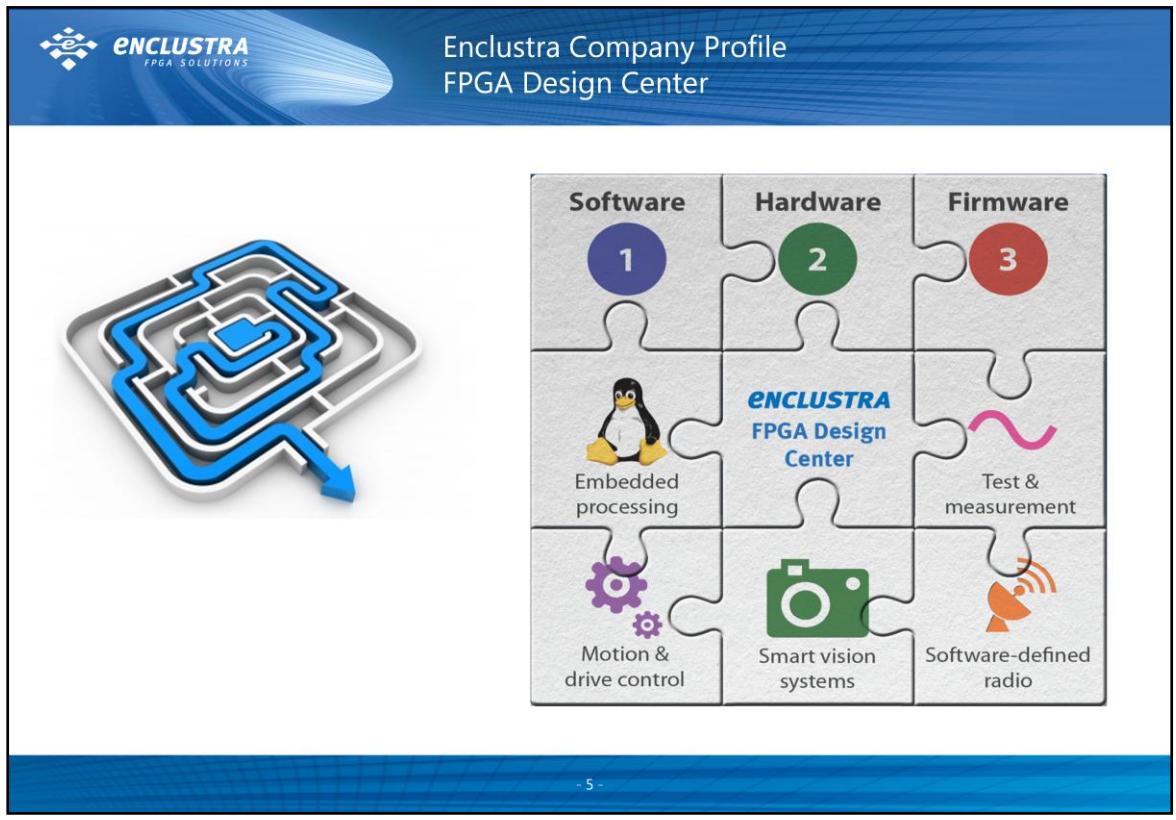
Our FPGA engineers have in-depth knowledge in various application areas like software defined radio, drive control, digital signal processing and data acquisition systems.

Investing in employee training and keeping critical knowledge up to date on a regular basis enables us to find ideal solutions at a minimal expense for our clients.

We're vendor-independent, and we're design service partners of Xilinx®, Altera® and Lattice Semiconductor – this close communication allows us to be forward-looking in our design process, and remain on the cutting edge of the most advanced FPGA technology.

Enclustra – we speak FPGA!

Enclustra GmbH
Technoparkstr. 1 – CH-8005 Zürich – Switzerland
Phone +41 43 343 39 43 – www.enclustra.com

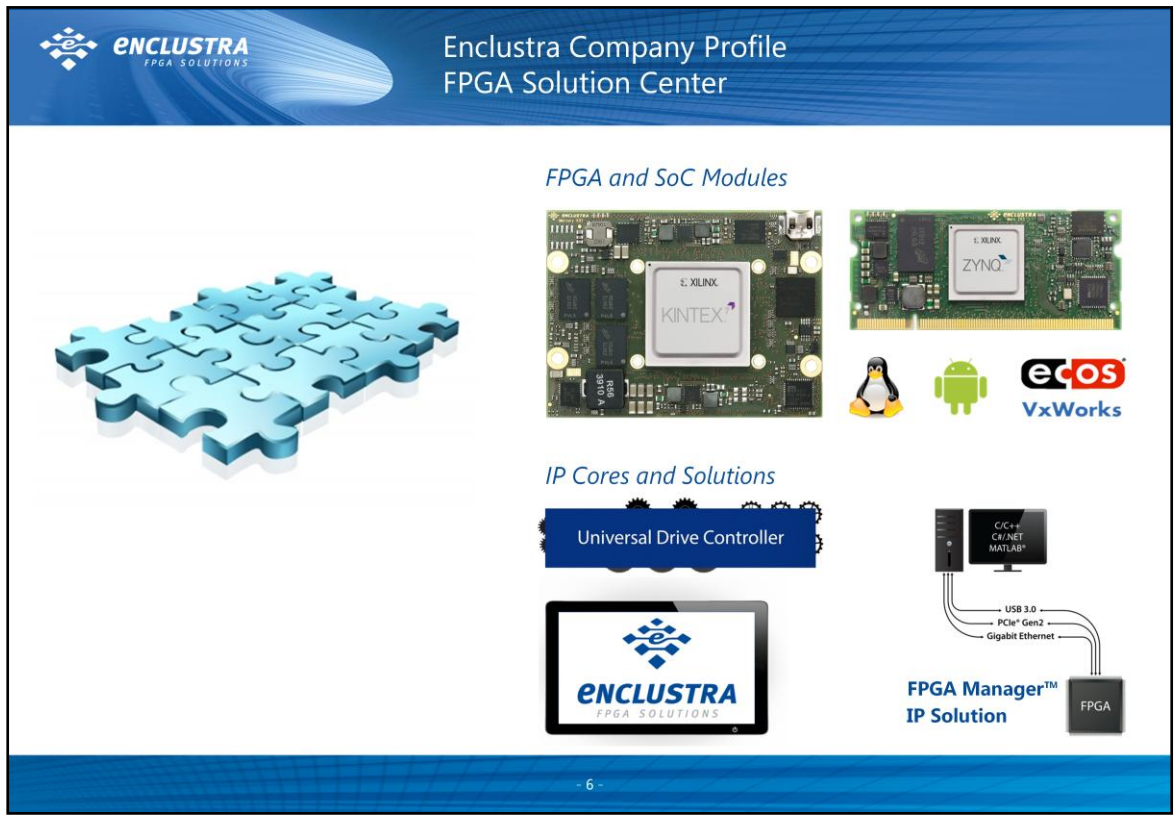


Short Profile

With the FPGA Design Center we accompany our customers on their way from an initial idea to a complete FPGA-based system.

Our design center offers design and support in all areas of FPGA-based system development, in a wide number of applications. High-speed hardware, HDL firmware, embedded software, real-time operating systems – our expertise covers every stage of the design process, from specification up to industrialization and manufacturing.

We have accumulated almost 100 person-years of FPGA hardware, firmware and software design experience and have immediate access to a comprehensive network of 3rd party consultants with specialized expertise for tasks like analog RF hardware design or thermal design. A lot of our past projects, particularly in the software defined radio and test & measurement areas, involved MATLAB and/or Simulink.



FPGA and SoC Modules

We develop and sell our own FPGA and system-on-chip (SoC) modules, based on Xilinx and Altera devices, for our customers to integrate into their own systems.

9 different modules, in 2 different families, compatible with 6 different base boards – the diversity of our products allows the customer to select exactly the features and they need, down to a fine grain.

Mars Module Family

- SO-DIMM 67.6 x 30 mm, 96-108 user I/Os, 0-2 MGTs
- 1-2 Ethernet ports, 0-1 USB port, 3.3V single supply voltage
- Very low-priced SO-DIMM connectors produced for the laptop industry

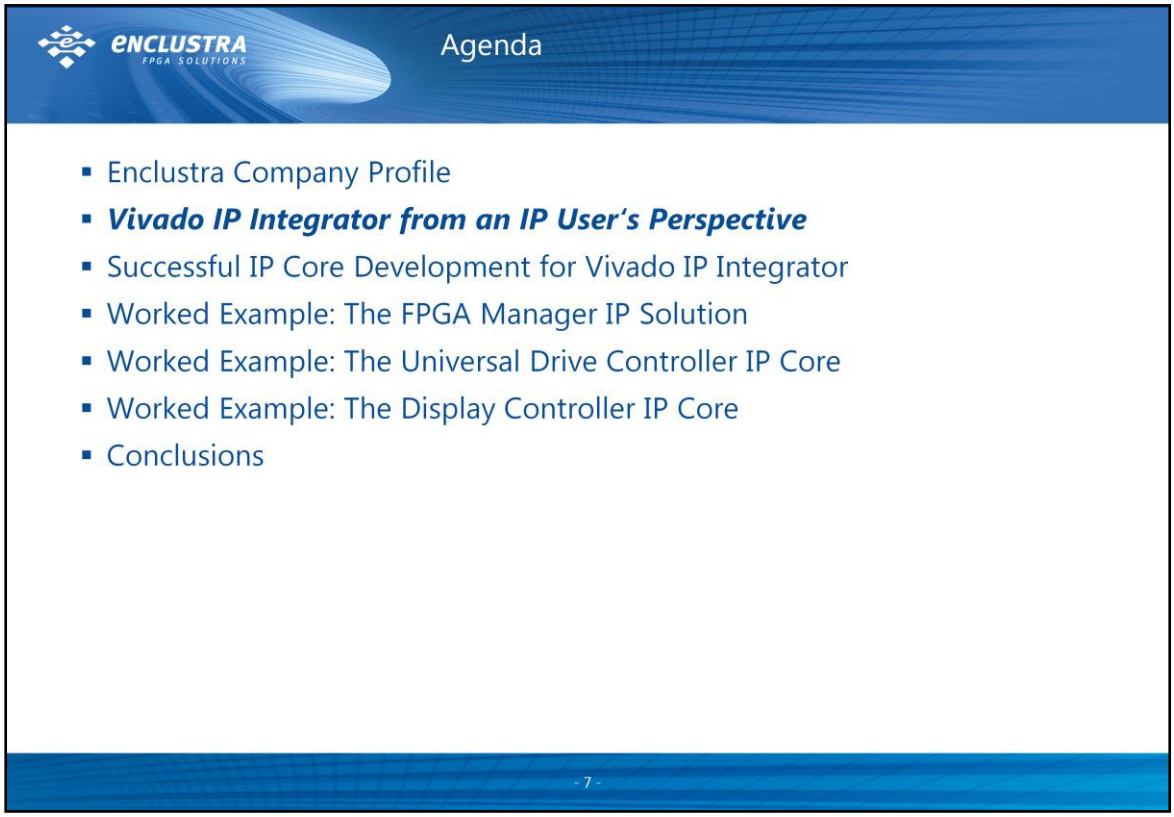
Mercury Module Family

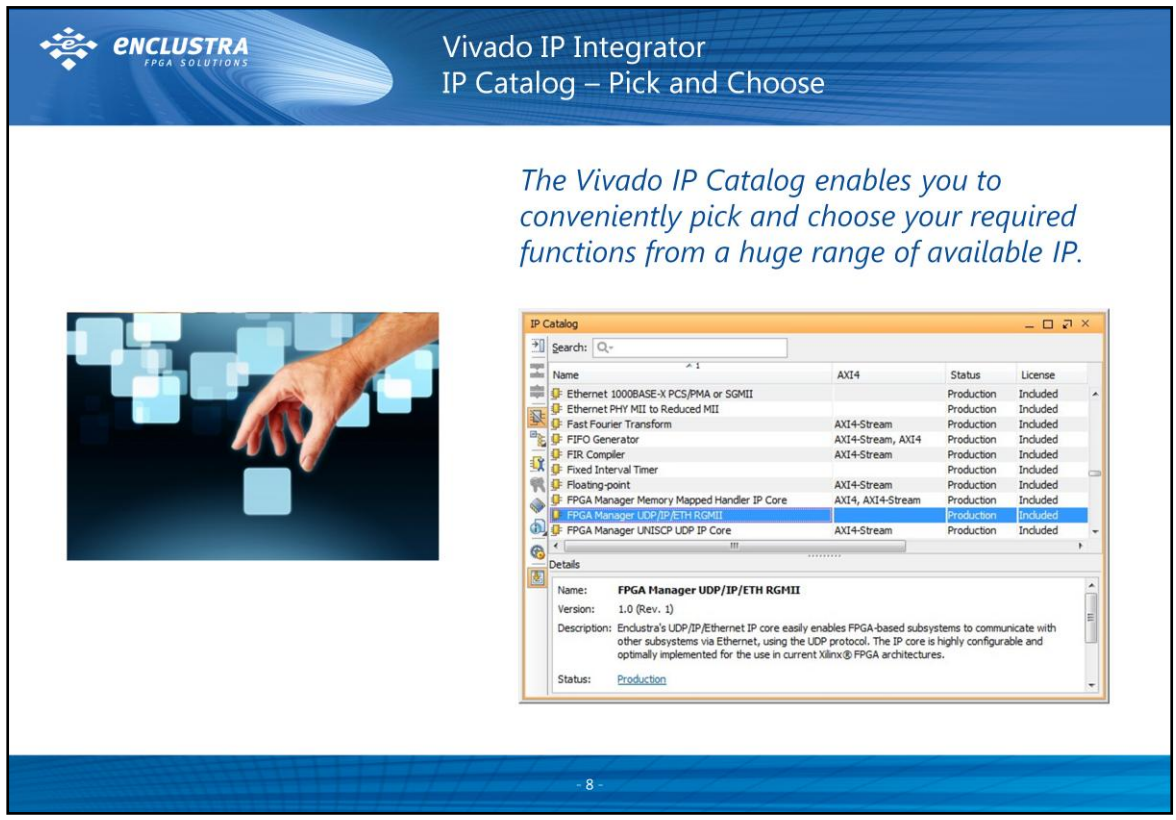
- 56 x 54 to 72 x 54 mm, 146-188 user I/Os, 0-8 MGTs
- 1-2 Ethernet ports, 1 USB port, 5-15V single supply voltage
- High-performance Hirose FX10 connectors

IP Cores and Solutions

Our FPGA-optimized IP cores and solutions enable quick, easy addition of desired functionality to any FPGA design, with minimal resource usage and minimal design cost.

- Universal Drive Controller – Advanced Velocity Estimator
- PROFINET IRT – UDP/IP Ethernet – Display Controller 2D
- FPGA Manager – Stream Buffer Controller

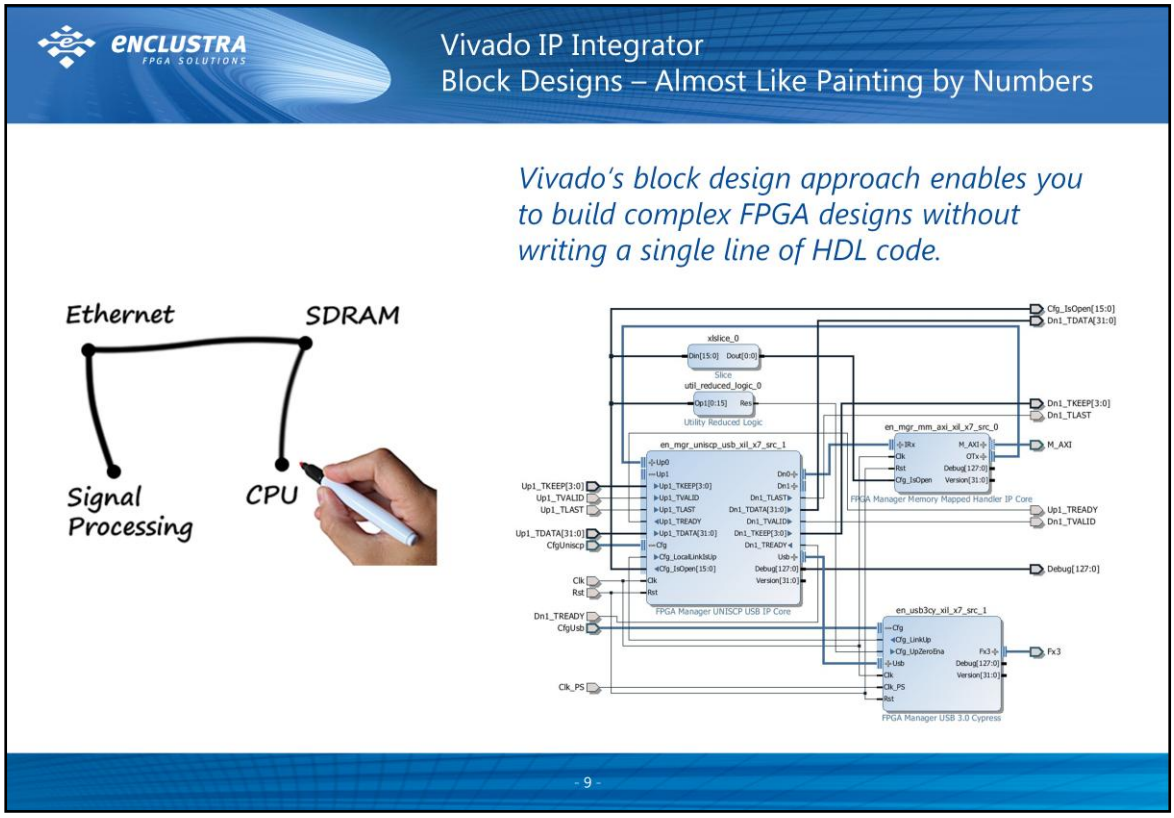

ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 7 -



Vivado IP Catalog

The Vivado IP Catalog is used to browse the IP that is available for a specific project. This IP can include Xilinx IP, third-party IP as well as IP previously developed by your company.

The Vivado IP Catalog provides filtered search over all available IP and displays the relevant information in a convenient way. Adding IP to your block design is simply done by double-clicking the IP and completing the IP configuration process.

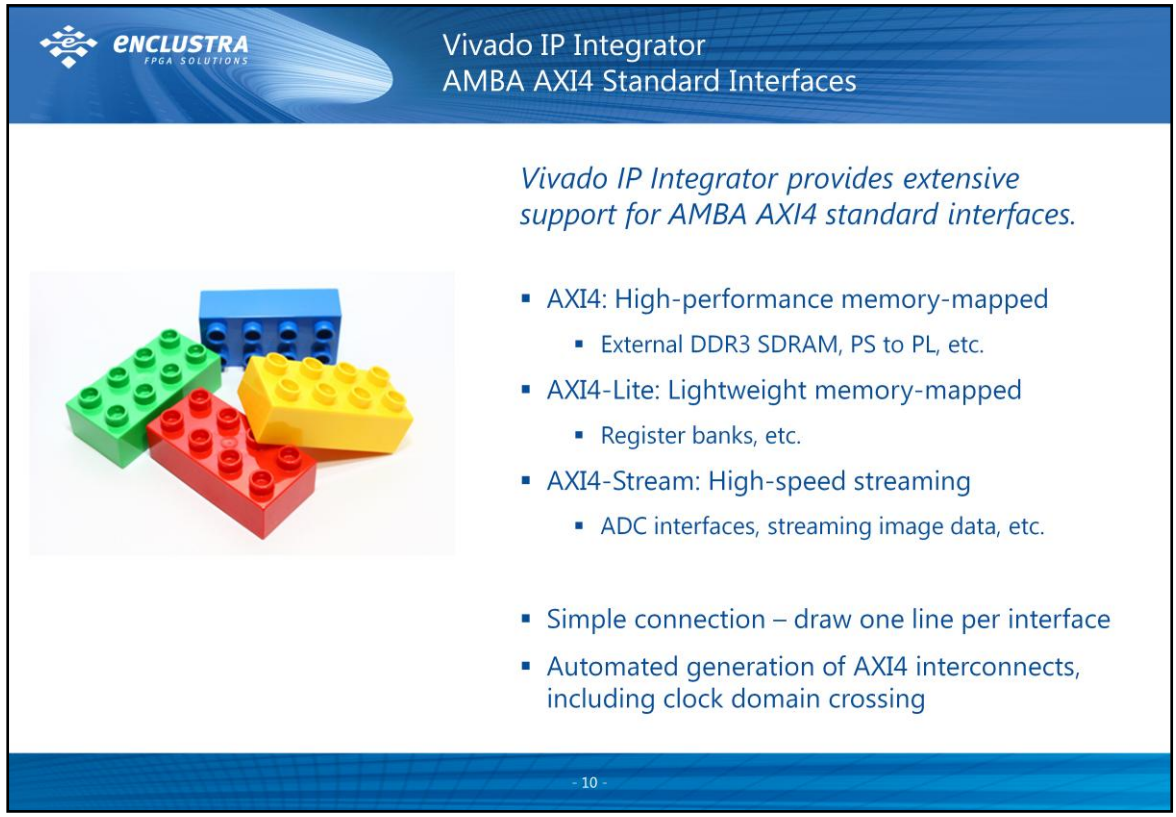


Block Designs

Block designs are the main design view used in Vivado IP Integrator. The graphical display of the system architecture provides a good overview while the capability of browsing through hierarchies enables the easy display of design details.

Automated Features

A lot of routine work that usually had to be done manually and diligently is now automated by Vivado IP Integrator. This includes automated connection of AXI4 interfaces, automated generation of system address maps and automated connection of external signals on known hardware, for example.



The AMBA AXI Standard

AXI is part of ARM AMBA, a family of micro controller buses first introduced in 1996. The first version of AXI was first included in AMBA 3.0, released in 2003. AMBA 4.0, released in 2010, includes the second major version of AXI, AXI4. There are three types of AXI4 interfaces:

AXI4

AXI4 is for memory-mapped interfaces and allows high throughput bursts of up to 256 data transfer cycles with just a single address phase.

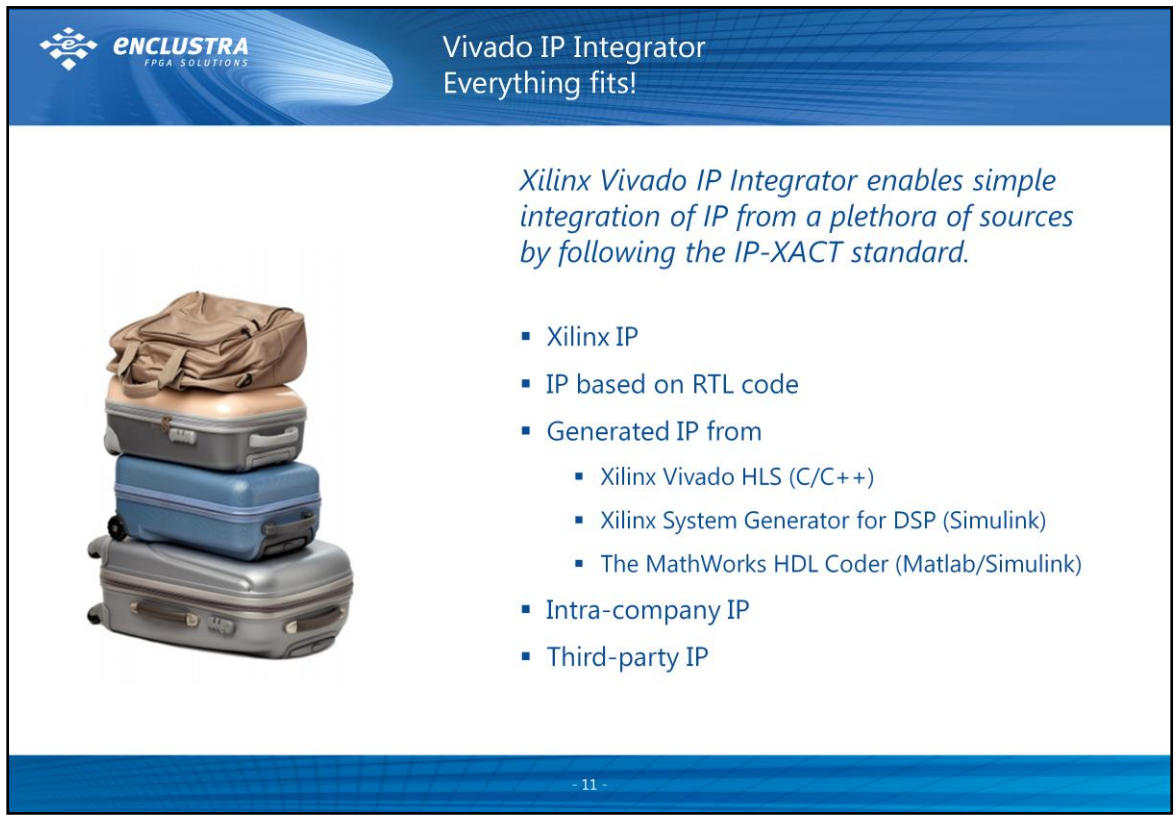
AXI4-Lite

AXI4-Lite is a light-weight, single transaction memory-mapped interface. It has a small logic footprint and is a simple interface to work with both in design and usage.

AXI4-Stream

AXI4-Stream removes the requirement for an address phase altogether and allows unlimited data burst size. AXI4-Stream interfaces and transfers do not have address phases and are therefore not considered to be memory-mapped.

(Source: ug1037-vivado-axi-reference-guide.pdf, Xilinx Inc., April 2, 2014)

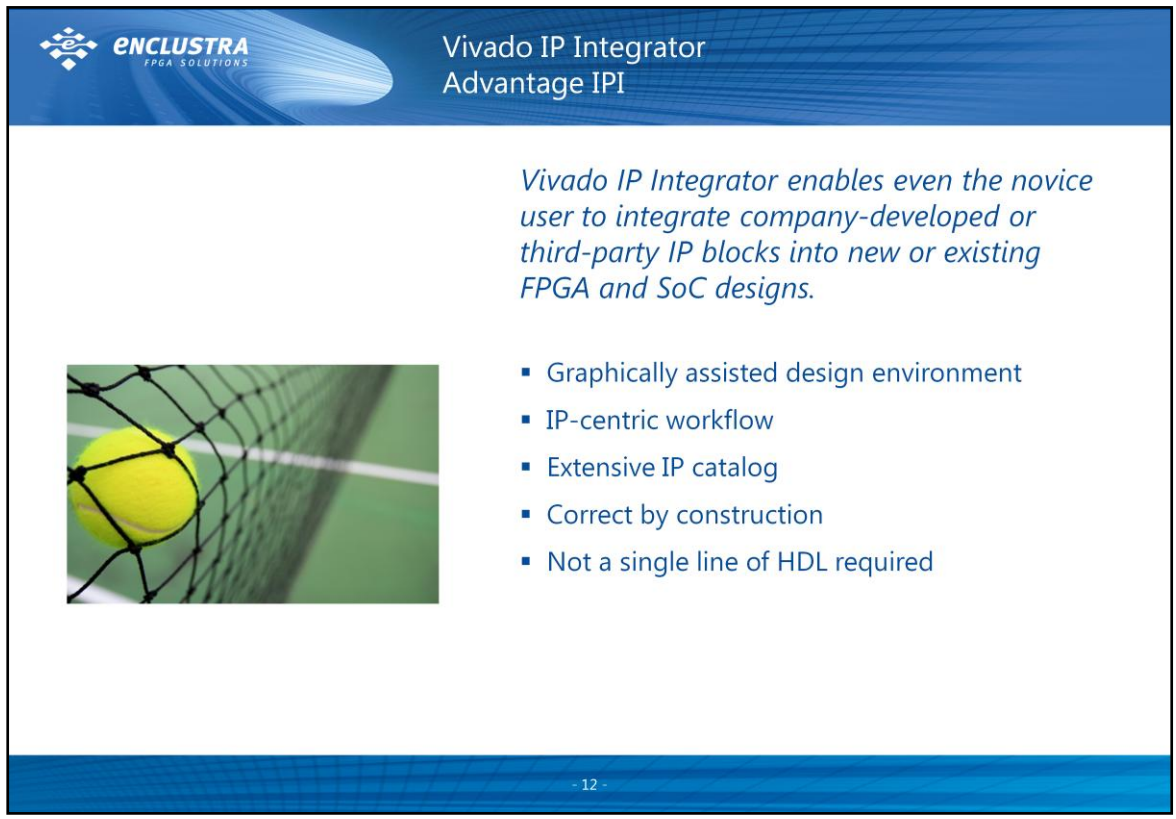


The IP-XACT Standard (IEEE-1685)

IP-XACT is an XML format that defines and describes electronic components and their designs. IP-XACT was created by the SPIRIT Consortium as a standard to enable automated configuration and integration through tools.

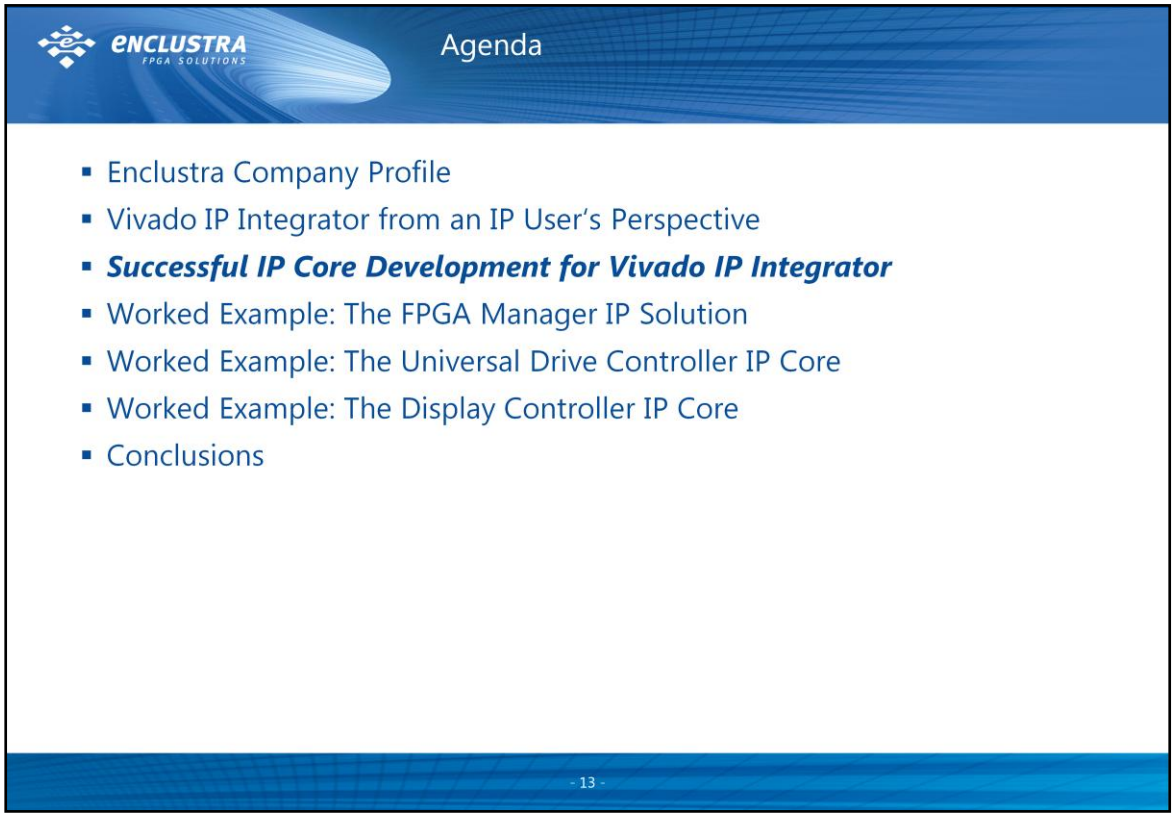
The goals of the standard are:

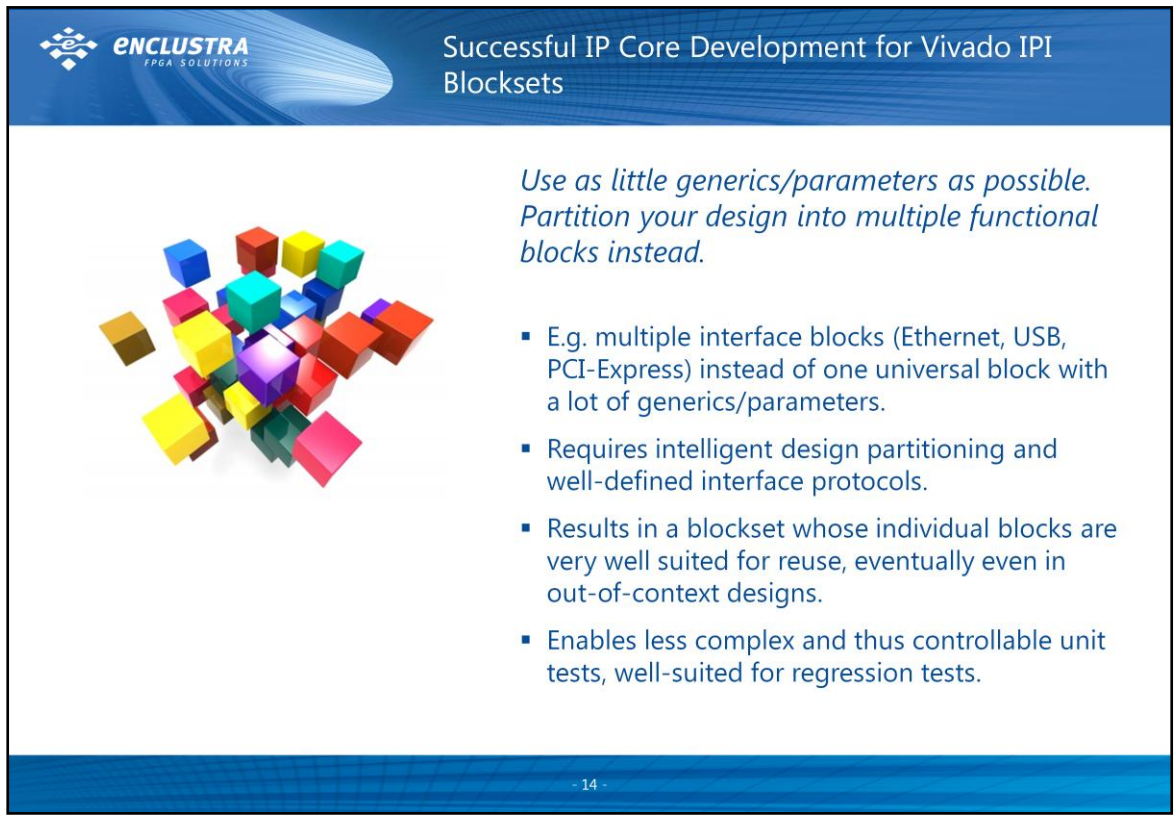
- to ensure delivery of compatible component descriptions from multiple component vendors,
- to enable exchanging complex component libraries between electronic design automation (EDA) tools for SoC design (design environments)
- to describe configurable components using metadata, and
- to enable the provision of EDA vendor-neutral scripts for component creation and configuration (generators, configurators).



A new Approach

Vivado IP Integrator with its graphically assisted, IP-centric workflow enables a whole new group of system designers to successfully build FPGA and SoC designs. The need for extensive HDL knowledge is reduced and the software-centric HLS tools enable even C and C++ to be implemented in FPGA logic.


ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 13 -



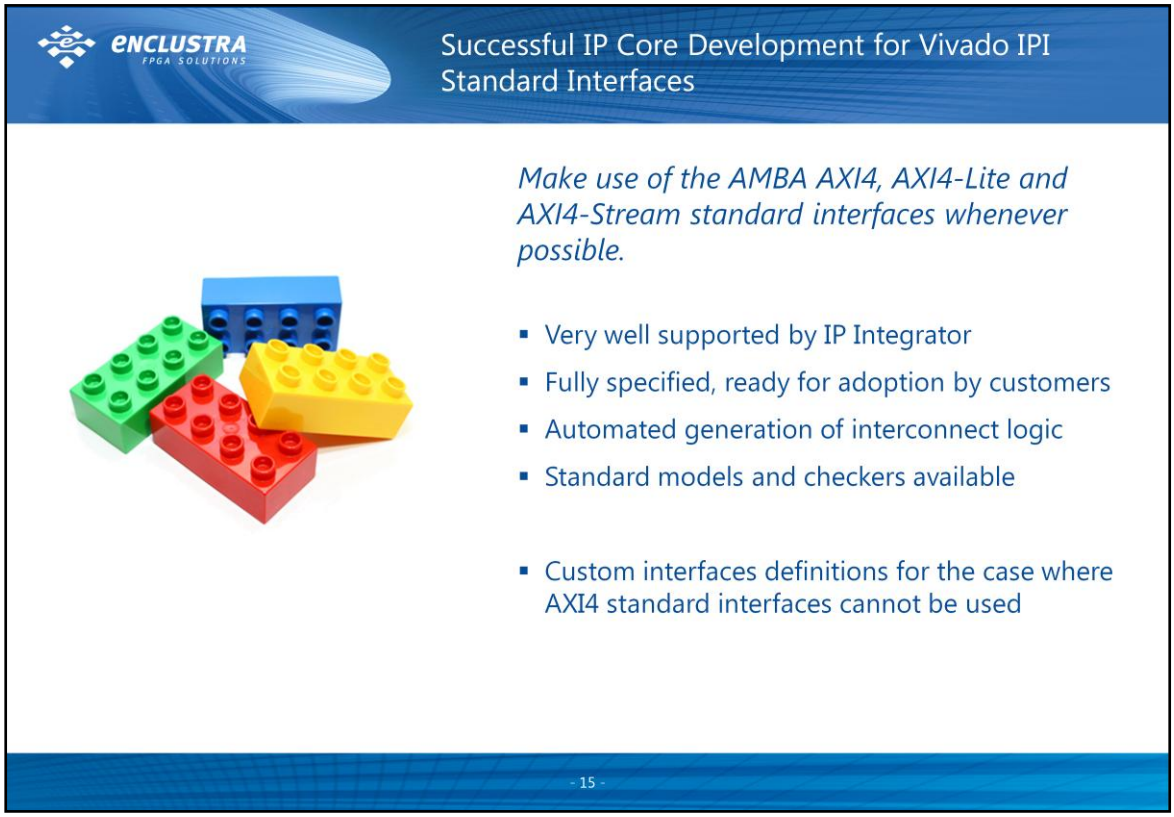
The Case for Block Sets

Partitioning the IP functionality into smaller blocks, resulting in a block set, provides multiple advantages over a single all-in-one IP block.

Single all-in-one IP blocks are usually complex and mix functionalities of different technologies and layers. As a result of this, they usually require a big number of generics/parameters to be configured by the user, with the majority of the possible generic/parameter combinations being invalid. Such IP blocks are thus very complicated to use and require very complex unit tests for a way too big number of generic/parameter combinations.

When properly partitioned however, the same functionality can be achieved by more simple IP blocks that concentrate on one technology and layer per block. This makes the blocks easier understandable by the user, requires less generics/parameters and enables less complex unit tests.

In addition to this, reuse of these blocks is much more possible than for big, complex IP blocks and new functionality can be added to block sets without modifying the existing blocks.



Partitioning

The possibility to conveniently use AXI4 standard interfaces between individual IP blocks is strongly dependent on system partitioning.

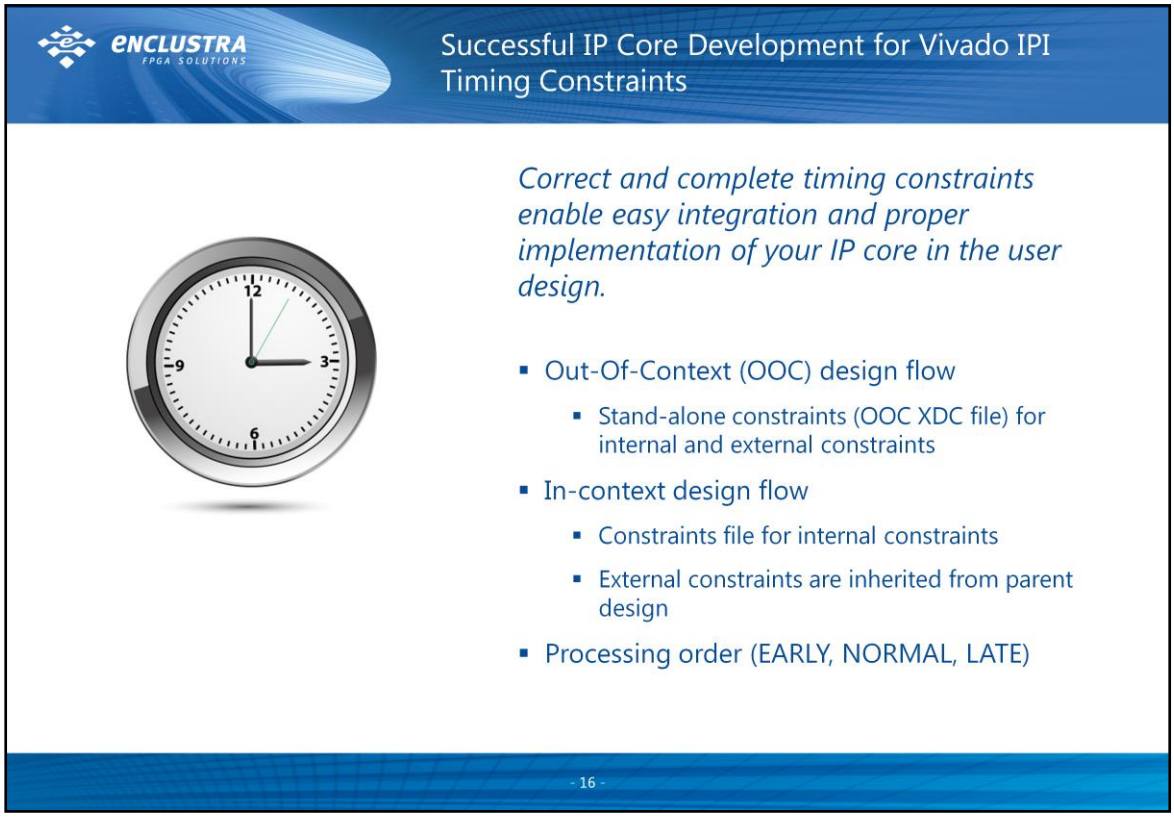
In general, it is recommended to introduce IP block boundaries at places where the resulting interface is going to be as simple as possible and the interface requirements are as low as possible. It may then only need minor design modifications, if any at all, to transform the resulting interface into an AXI4 standard interface.

User Signals

In some cases, an interface resulting from design partitioning requires some additional side-band information in addition to the AXI4 standard interface. This problem can, in most of the cases, be solved by making use of the USER signals defined in the AXI4 standard.

Custom Interfaces

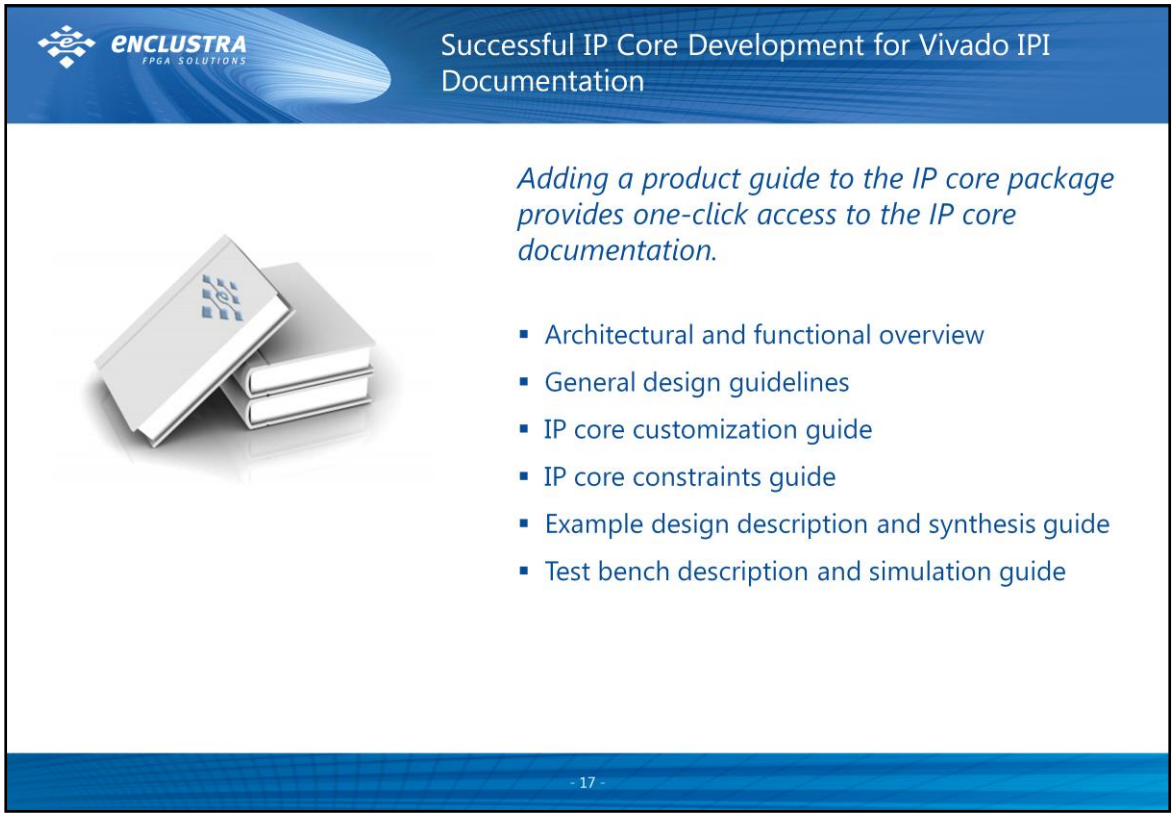
If the use of AXI4 standard interfaces is not appropriate, Vivado IPI provides the possibility to define custom interfaces, which represent a group of arbitrary signals. This mechanism allows to handle custom interfaces very similar to AXI4 standard interfaces.



Timing Constraints are Vital

Although one can come across many FPGA designs without proper (i.e. correct and complete) timing constraints, this is a no-go for successful FPGA and IP design.

Every FPGA design, and thus every IP core, should be fully constrained. This means that no unconstrained paths are to be shown in the timing reports.

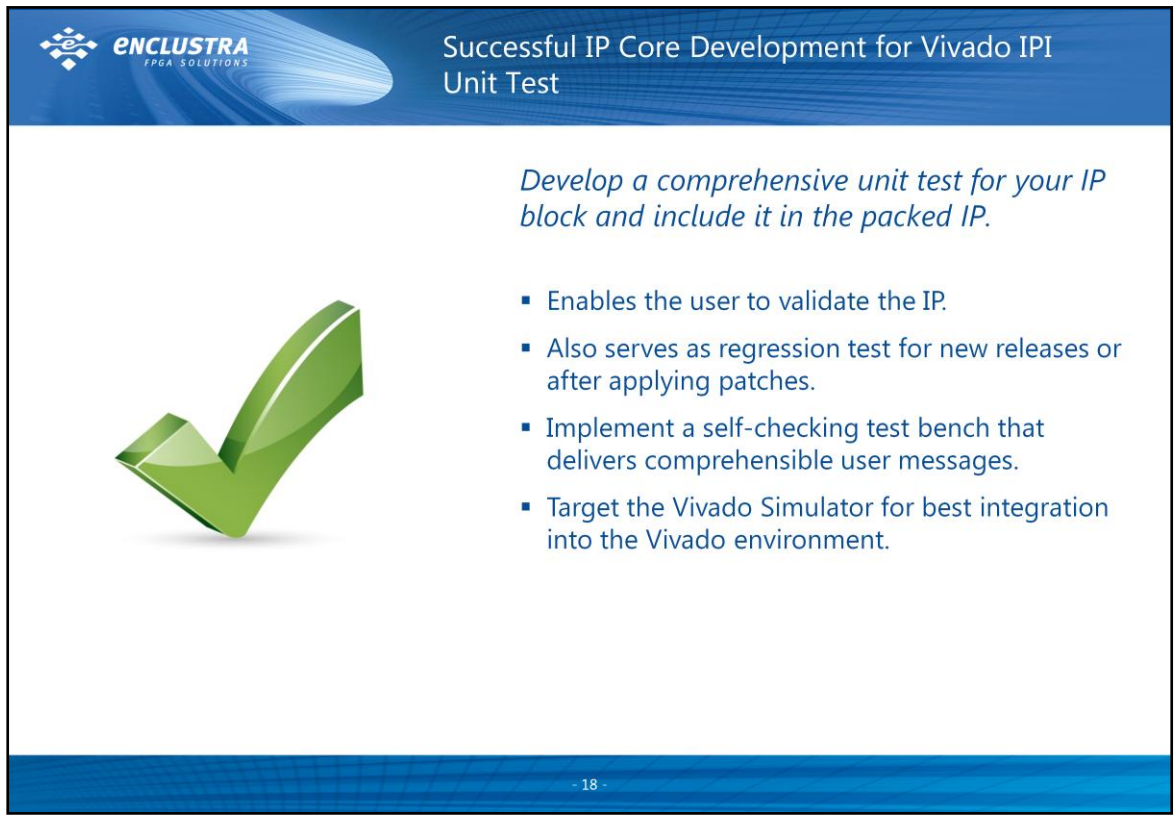


As close as it gets

Having the IP product guide only a single click away saves the user a lot of searching and the IP designer a lot of question – and both of them a lot of time.

Revisioning

Also, having the documentation packed together with the IP core, it is never a question whether the user has got the matching document revision for the IP core revision currently in use.



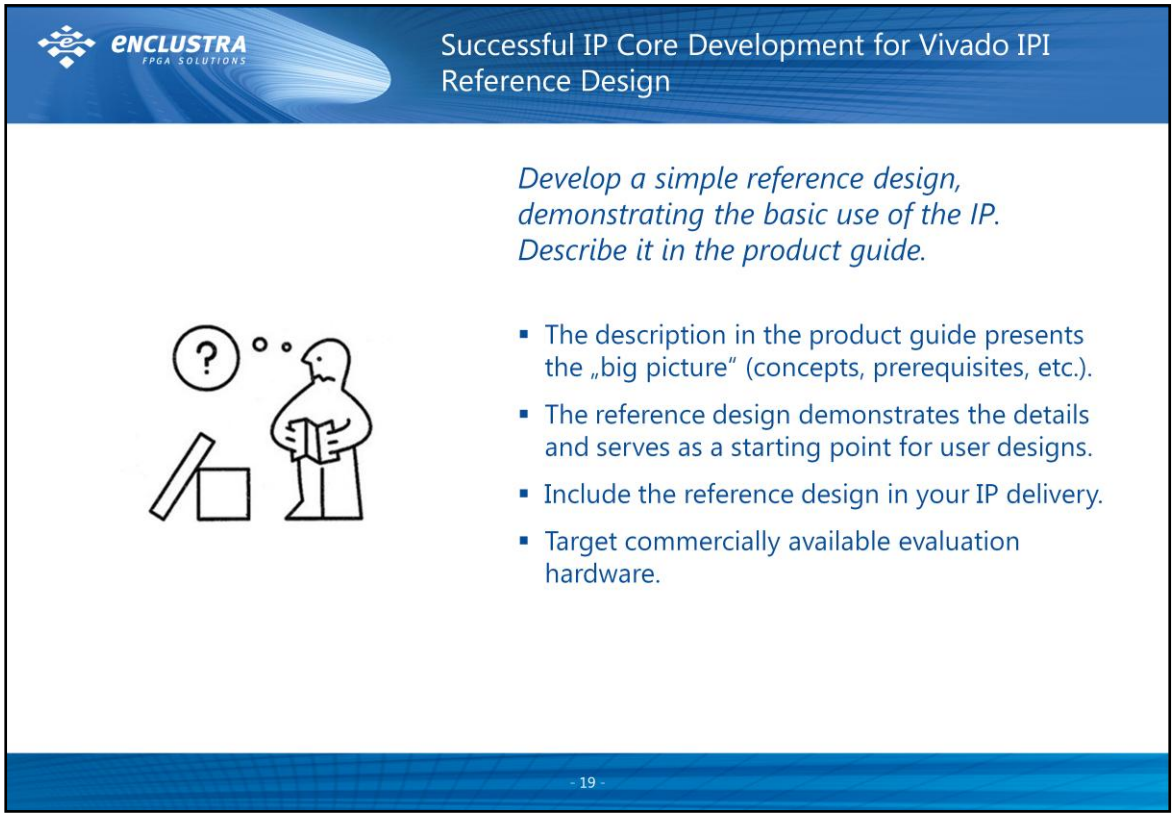
Regression Tests are Crucial

Also successful IP core design does not go without bug fixes, feature extensions and releasing new IP core revisions.

All the above processes can only be properly handled, if a regression test suite for each IP core is available and maintained. Tests for validating bug fixes shall be integrated into the regression test suite, the same is true for tests for verifying feature extensions.

Make the Tests available to the User

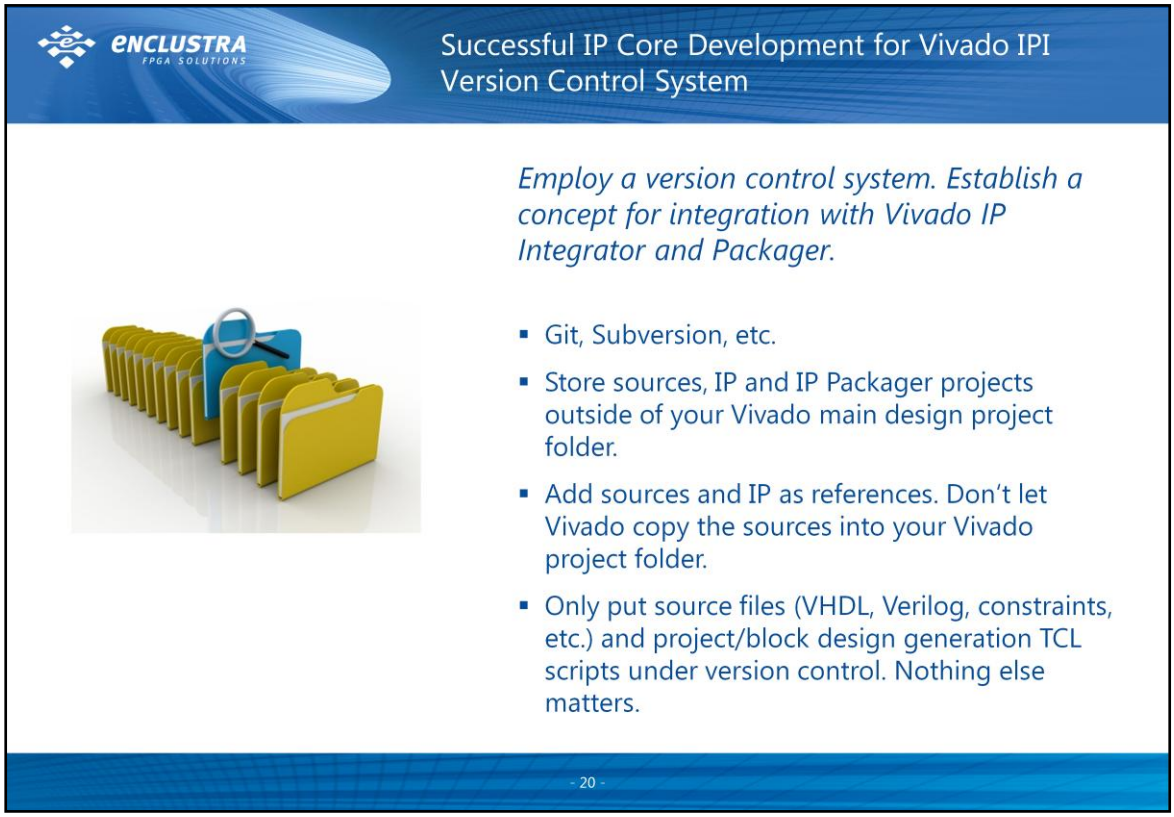
Making the tests – or a subset of the tests – available to the user allows the user to validate his copy of the IP in his context, which helps to build up confidence in the IP solution at first contact, most probably during the evaluation phase. It also helps in resolving potential problems, because the user can do a first error analysis using the unit test.



The Value of Reference Designs

The "big picture" is often easily transported to the user. But when it comes to implementing a user design incorporating a more or less complex IP core, getting every tiny detail right is often a challenge. Especially if the design does not work in the first attempt, finding the cause can be extremely difficult.

Exactly in this case, a simple reference design demonstrating the details can make the major difference and serve as a starting point that is known to be working. By targeting commercially available evaluation hardware, every user has the chance to get the reference design running within minutes.



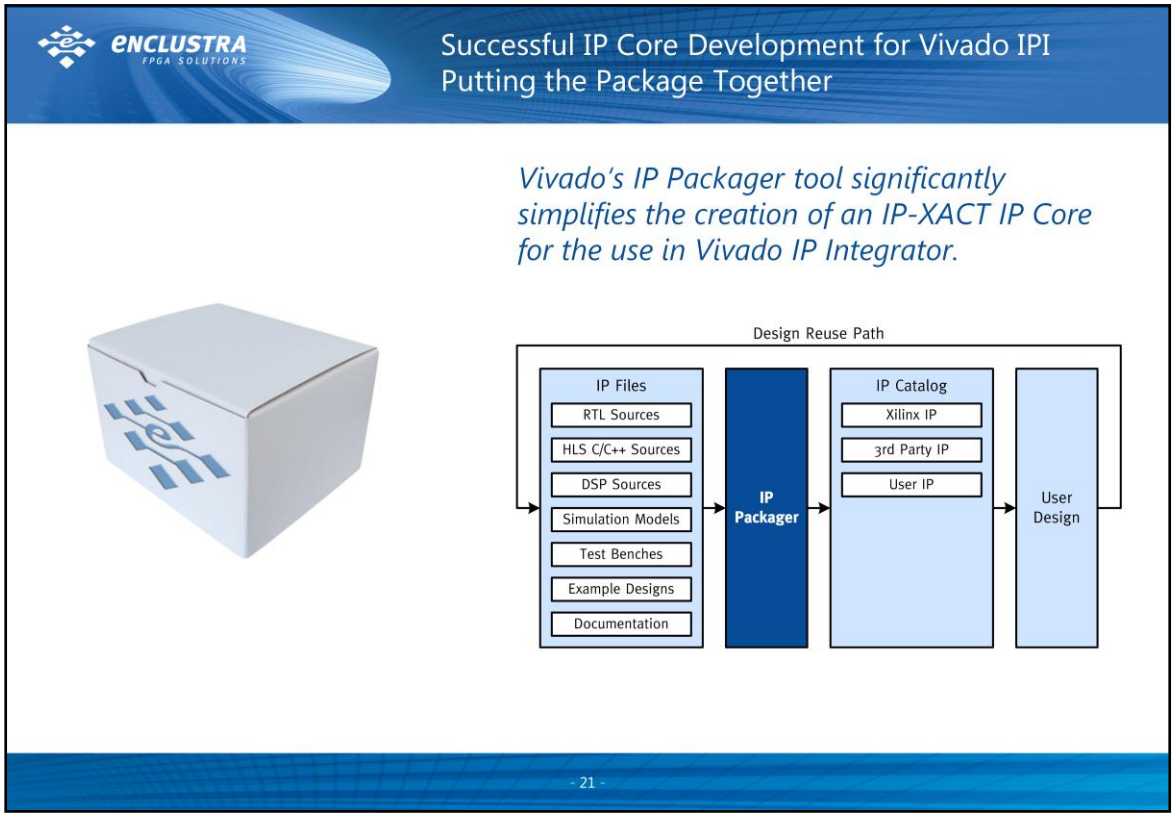
Version Control

Version control allows you to revert files back to a previous state, revert the entire project back to a previous state, compare changes over time, see who last modified something that might be causing a problem, who introduced an issue and when, and more. Using a version control system also generally means that if you screw things up or lose files, you can easily recover. In addition, you get all this for very little overhead.

Integration with Vivado

It does basically not matter, what version control system you are using. Vivado however has a few restrictions that need to be respected in order to successfully collaborate with any kind of version control system:

- Store sources, IP and IP Packager projects outside of your Vivado main design project folder. Always. Otherwise you will not be able to regenerate your project.
- Add sources and IP as references. Don't let Vivado copy the sources into your Vivado project folder. Always. Otherwise you will not be able to regenerate your project.
- Only put source files (VHDL, Verilog, constraints, etc.) and project/block design generation TCL scripts under version control. Regenerate your project and block designs by sourcing these TCL scripts after clone/checkout.

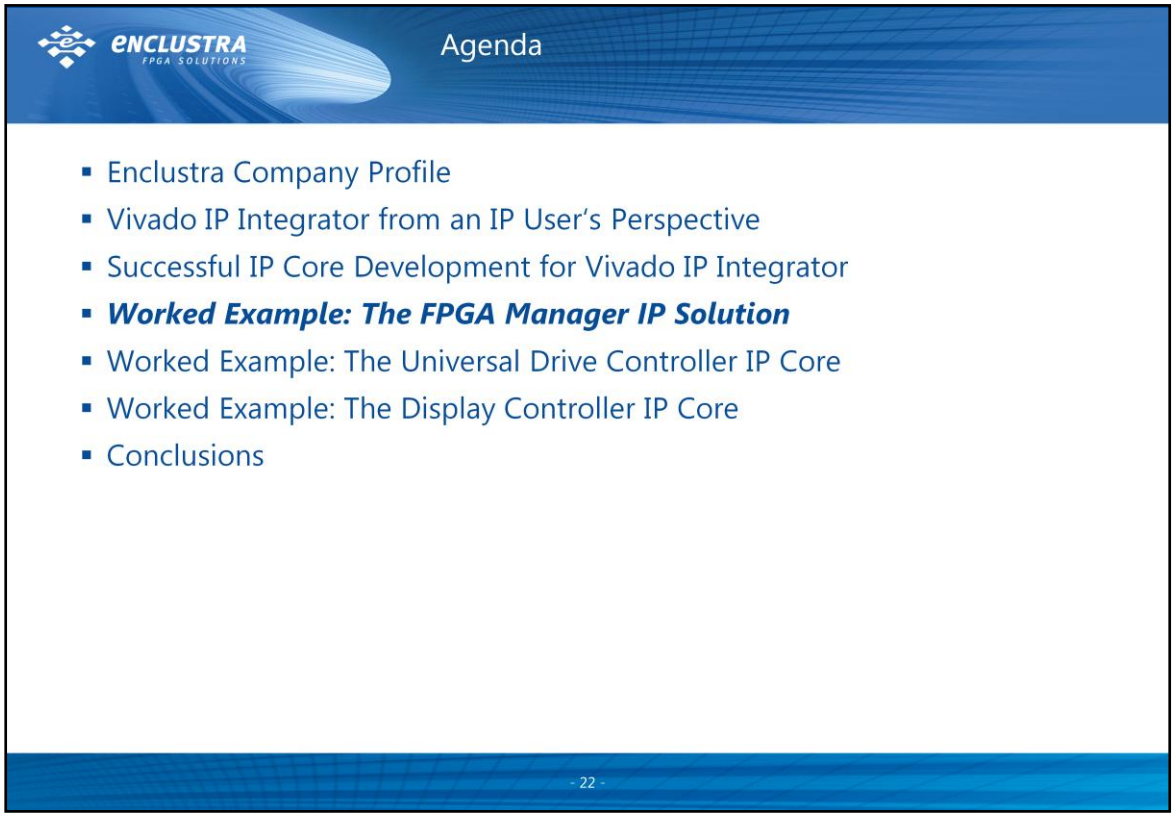


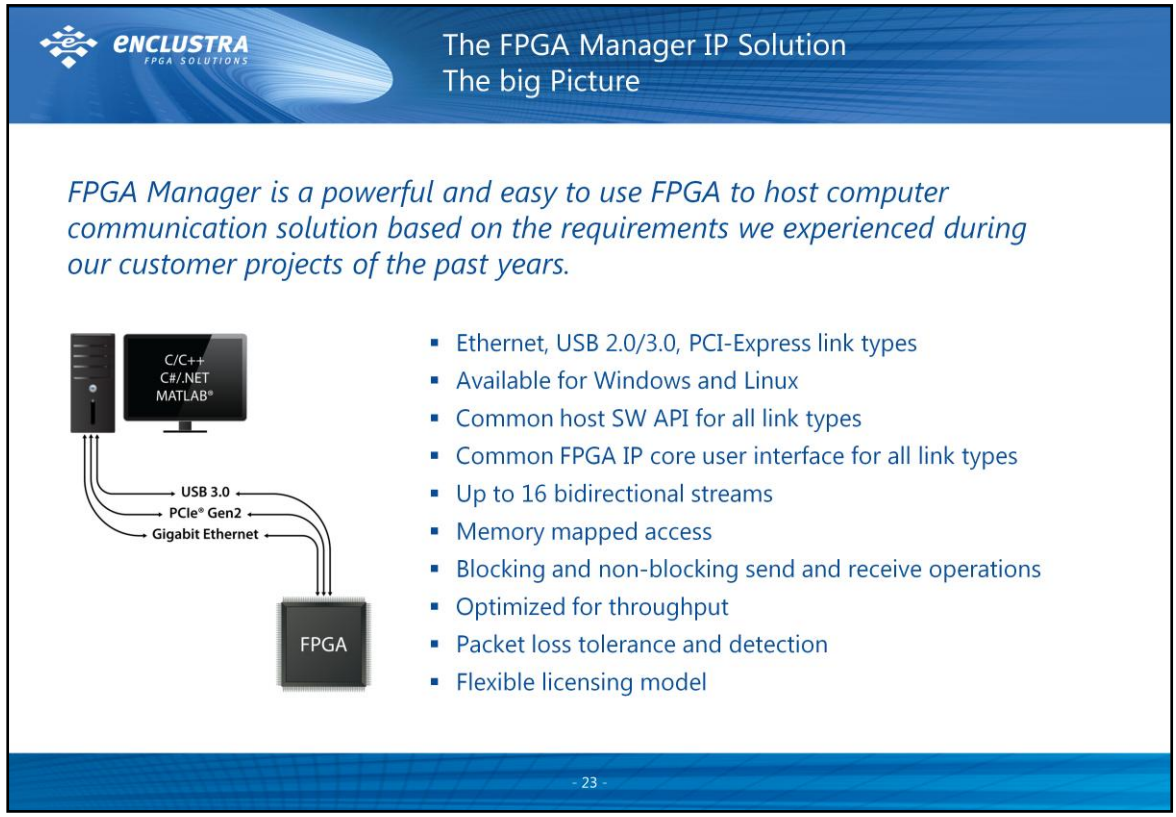
Vivado and Design Reuse

Vivado's IP Packager / IP Integrator environment strongly simplifies and encourages design reuse.

Technically, any Vivado design can be packaged at any stage of the design flow and any Vivado design can thus be deployed as an IP core.

However, this requires thorough knowledge of the capabilities and restrictions of the Vivado IP Packager / IP Integrator environment.


ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 22 -

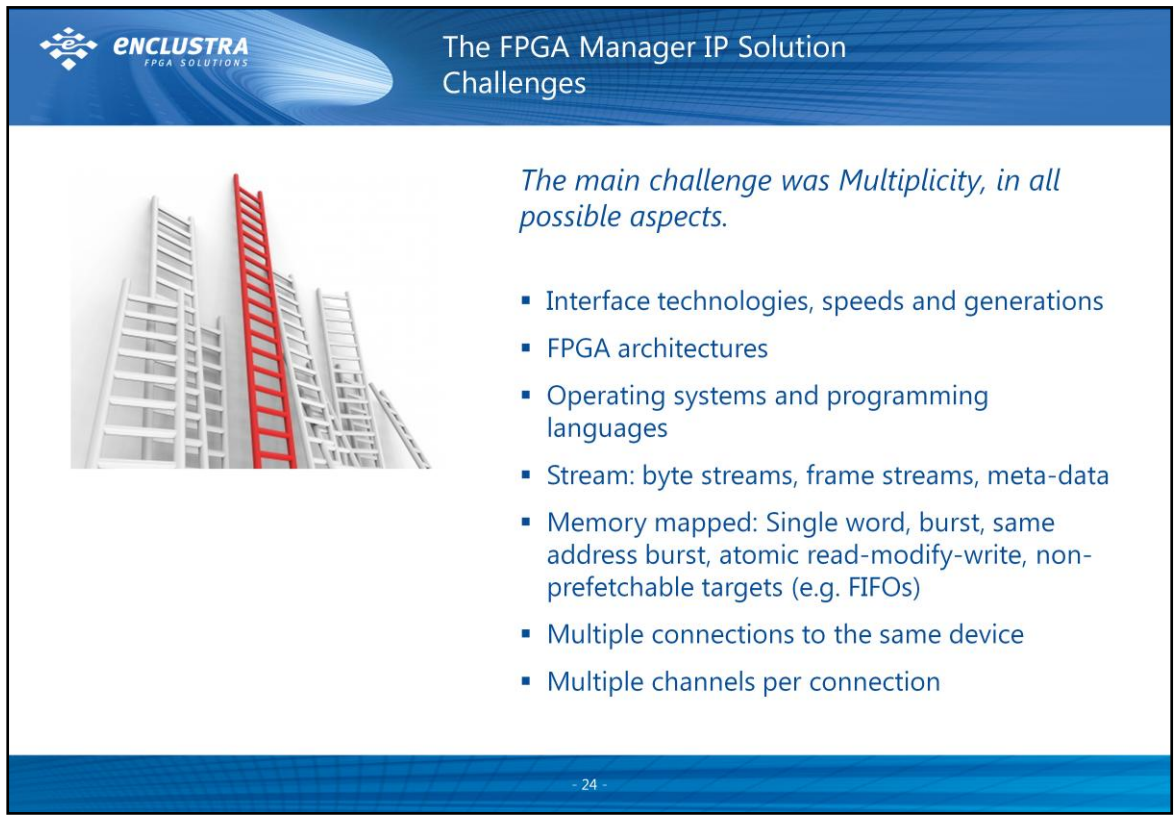


Background

The majority of our past projects required an FPGA to communicate with some kind of host computer in one or another way. Some of the requirements have been common to most of the projects (e.g. streaming and memory mapped data transfers), while others turned out to be diametrically opposed (e.g. operating systems or link types). FPGA Manager is an attempt to accommodate all of these requirements in a single communication solution, commercially available from Enclustra.

Short Description

Enclustra's FPGA Manager solution allows for easy and efficient data transfer between a host and a FPGA over different interface standards like USB 2.0/3.0, Gigabit Ethernet and PCI-Express. The solution includes a host software library (DLL), a suitable IP core for the FPGA and device controller firmware, if necessary. Device controllers are required for the USB 2.0 and USB 3.0 link types. The user host application can communicate with the FPGA through a simple API consisting of simple read/write data commands hiding the complexity of the underlying protocols. Streaming and memory-mapped accesses are supported.

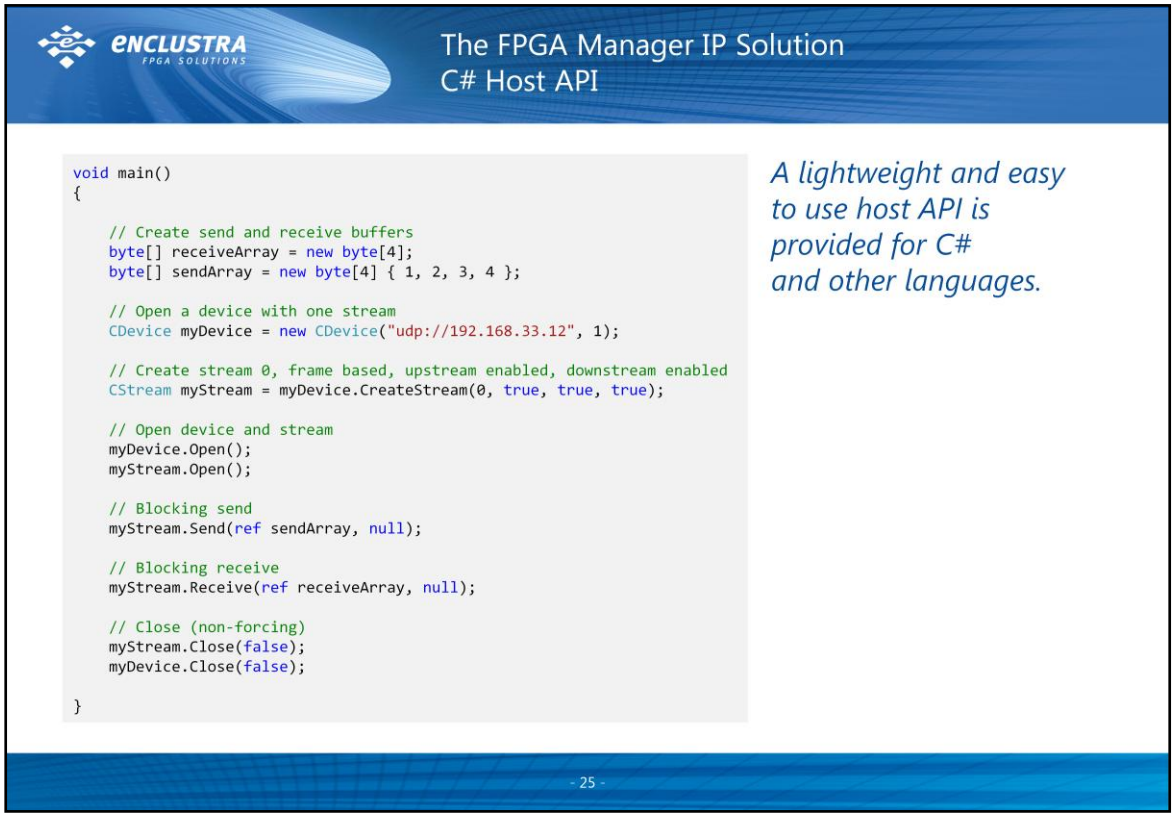


Challenges

During the development of a generic FPGA to host communication solution, a plethora of challenges are faced. Some of the challenges have nearly no impact on the partitioning and high-level tools used, while others have a great influence on how the design should be partitioned and on how the features of the high-level tools are used best.

The first class of challenges contains performance challenges like bandwidth vs. latency trade-offs, operation throughput and CPU utilization optimizations. Also device enumeration, error handling and other system management requirements belong to this class.

The challenges that heavily influence the partitioning and also the best use of tool features include all kind of multiplicity and multiplexing requirements as described in the slide above. Also design security and compatibility are major issues in this class of challenges.



Example Code

The example above shows a very simple C# script for setting up FPGA Manager communication between an FPGA and a host computer.

Portability

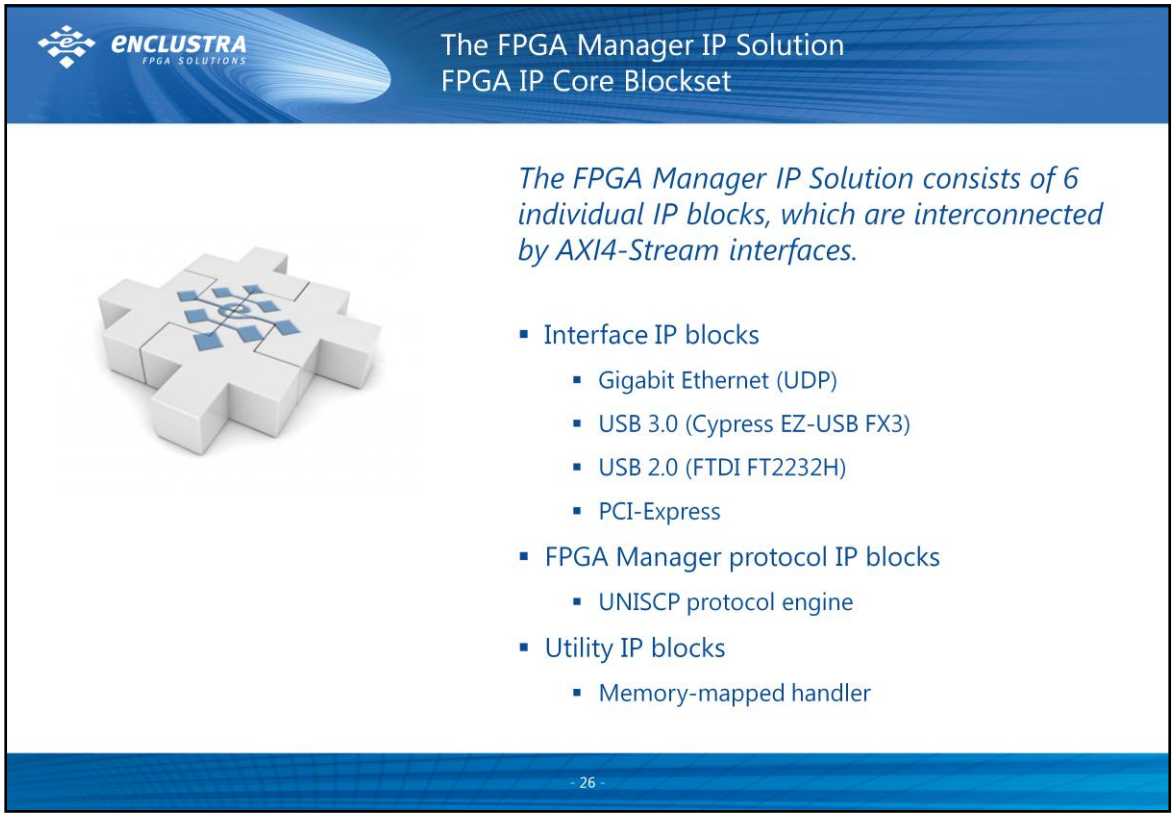
Other languages than MATLAB, such as C, C++ and C#, are supported with a very similar API, which makes the migration between different host programming languages as easy as it gets.

FPGA Manager and MATLAB

MATLAB is a de facto industry standard tool for signal processing and data analysis, almost every engineer working in these areas is familiar with it. MATLAB not only provides numerous toolboxes and commands for processing and analyzing data, but also contains a very powerful plotting engine. In a traditional FPGA-based signal processing project, MATLAB is used for a) implementing a behavioral model of the processing block and b) building up a validation and verification framework around the processing block model.

With the ability of FPGA Manager to provide direct access to data on the FPGA from MATLAB, the MATLAB infrastructure developed during the modeling and design phase can directly be reused for the development, validation and verification of the associated FPGA implementation. Using MATLAB together with FPGA Manager results in much more streamlined development, validation and verification flows when compared to file-based data exchange, e.g. via .csv files.

MATLAB is often used during the development and prototyping phases where flexible data processing and analysis capabilities are most important. When it comes to user interfaces for production systems however, other programming languages like C/C++ or C#/.NET are more common. Since FPGA Manager provides the same host API and IP core interfaces for all of these languages, the migration to the production system programming language goes without touching the FPGA implementation and is as simple as it probably gets.



FPGA Manager Block Set

The FPGA Manager block set consist of three IP block subsets:

Interface IP Blocks

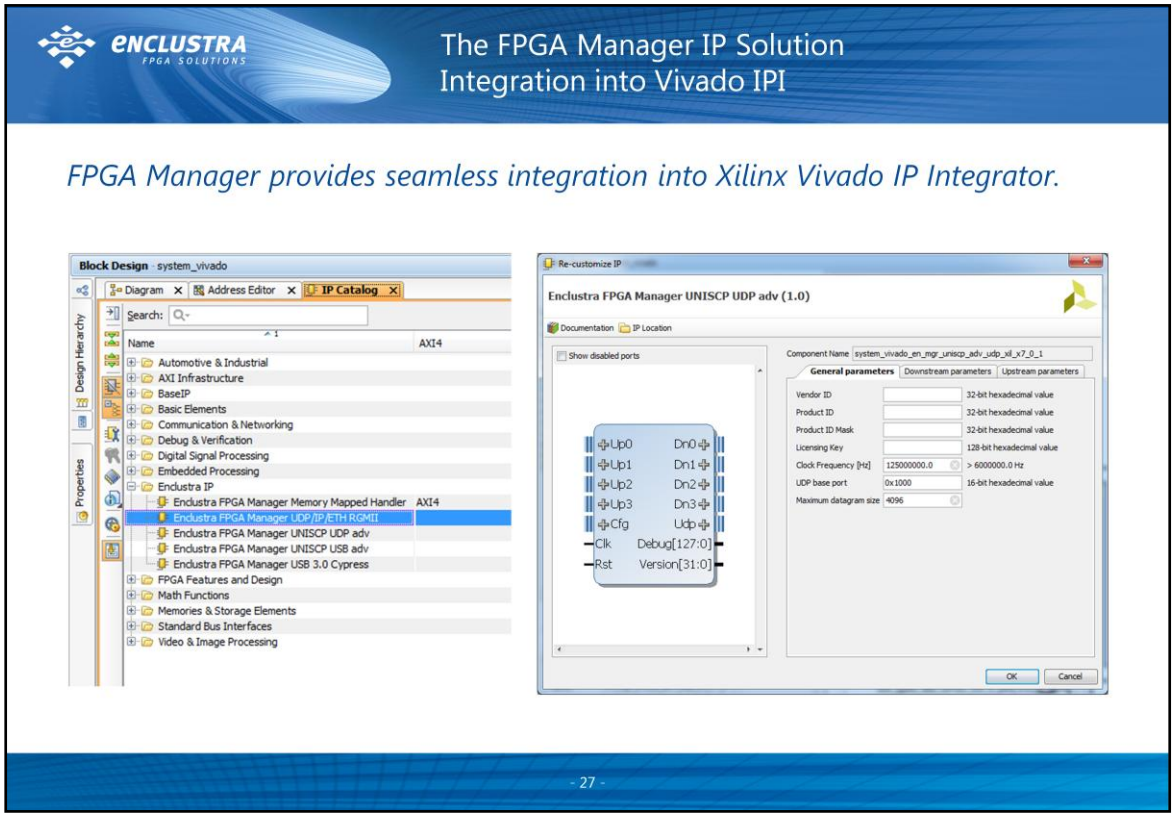
These blocks convert the FPGA Manager internal up- and downstream AXI4-Stream interfaces into the interface technology dependent interfaces. All blocks provide exactly the same FPGA Manager internal interfaces, so they can be exchanged very easily and without affecting the remaining IP blocks in any way.

FPGA Manager Protocol IP Blocks

These blocks process the FPGA Manager internal communication protocol, called UNISCP.

Utility IP Blocks

These blocks provide utility functions like the conversion of UNISCP streams into memory-mapped accesses and vice versa.



Graphically assisted Design Environments

Today's FPGA design tools provide graphically assisted design environments which allows even the novice user to integrate company-developed or third-party IP blocks into new or existing FPGA designs. The successful use of the technology requires a carefully chosen concept when developing IP cores. Instead of specifying dozens of textual generic parameters, a flexible IP core should be divided into a set of blocks that can then be interconnected graphically by an FPGA developer in order to obtain the features required for his target application.

FPGA Manager Block Set

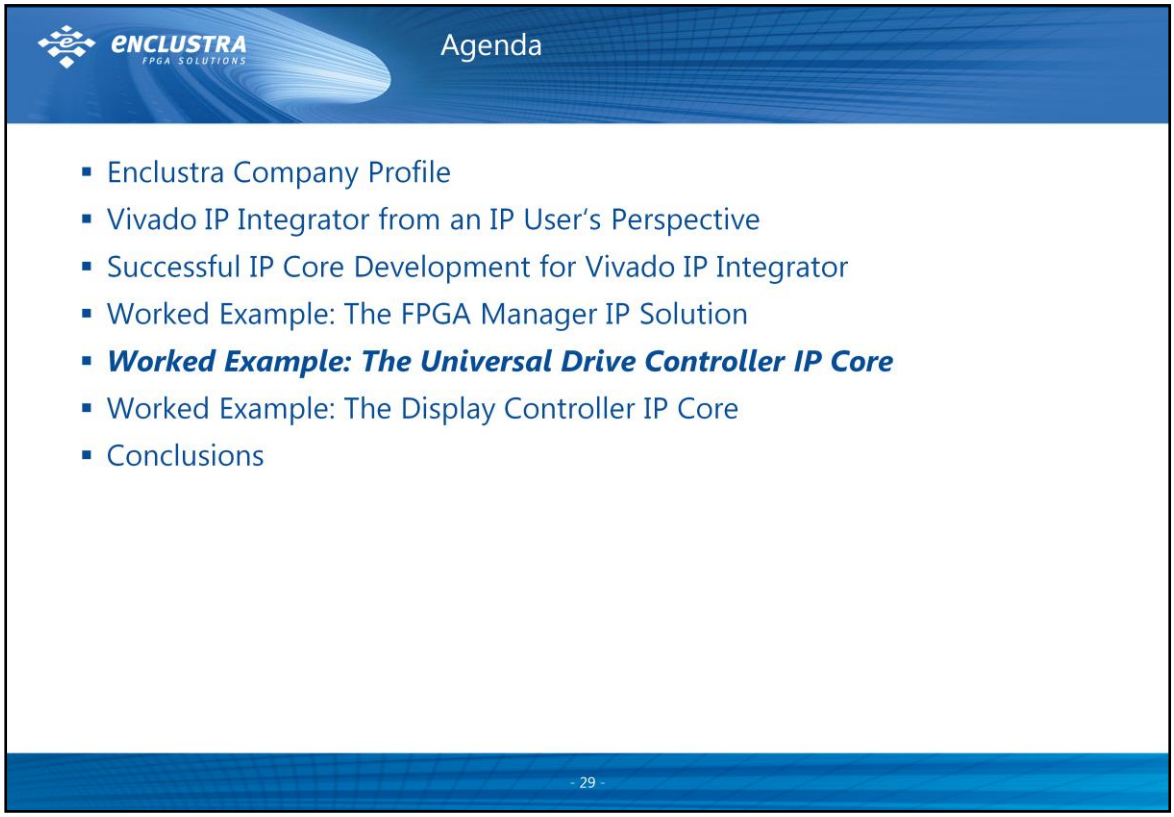
The FPGA portion of Enclustra's FPGA Manager IP solution is basically a set of 6 IP cores, which together provide PCI-Express, USB 3.0 and Gigabit Ethernet streaming capabilities for up to 16 data streams. The use of AXI4 memory-mapped and AXI4-Stream interfaces for interconnecting these IP cores is fully supported by the Xilinx Vivado and Altera QSYS tools.

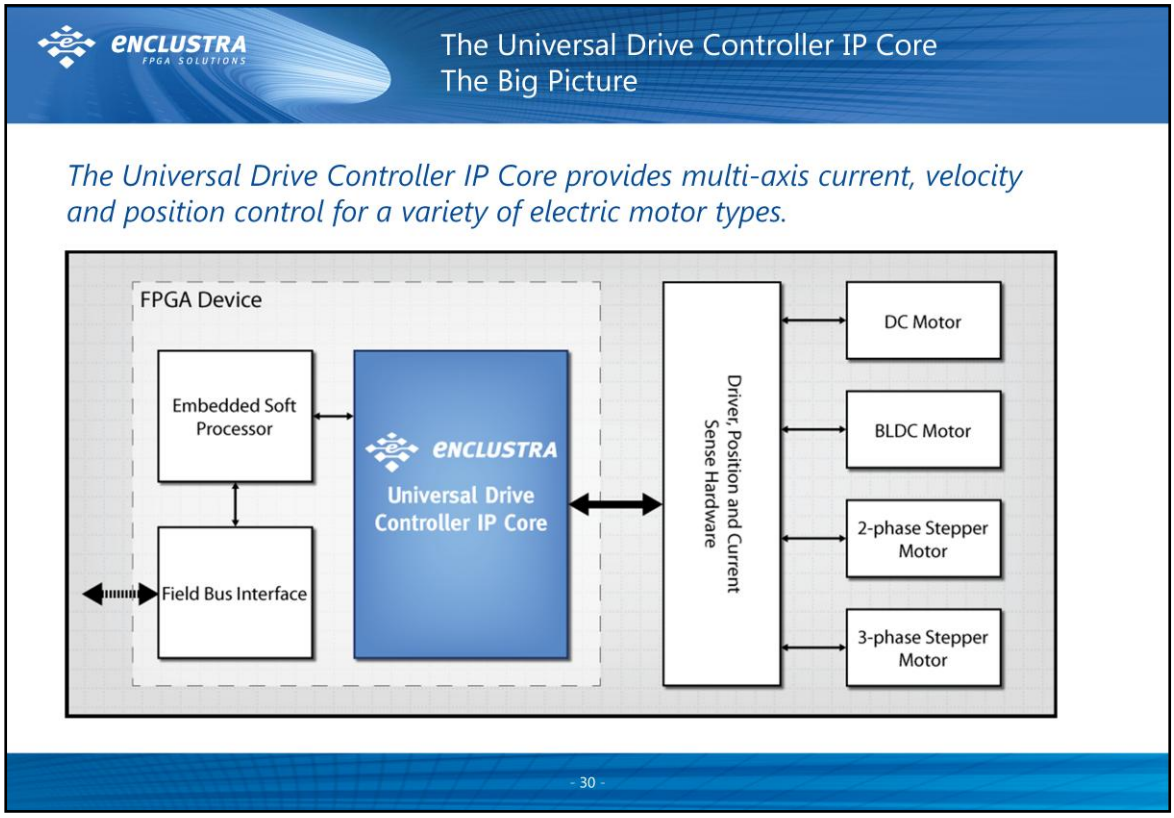
Integration with generated HDL Code

Since generated HDL code can also be deployed as a suitable IP core for the most common FPGA design tools, the integration of FPGA Manager with generated HDL code can be accomplished without writing a single line of HDL code.

Open to the Future

Due to the modular block set architecture of the FPGA Manager IP solution, new features like additional link types can be added to the overall solution by just adding new blocks to the existing block set. The major advantage of this approach is that new features can be added without touching existing blocks, which minimizes the verification effort of the overall solution.

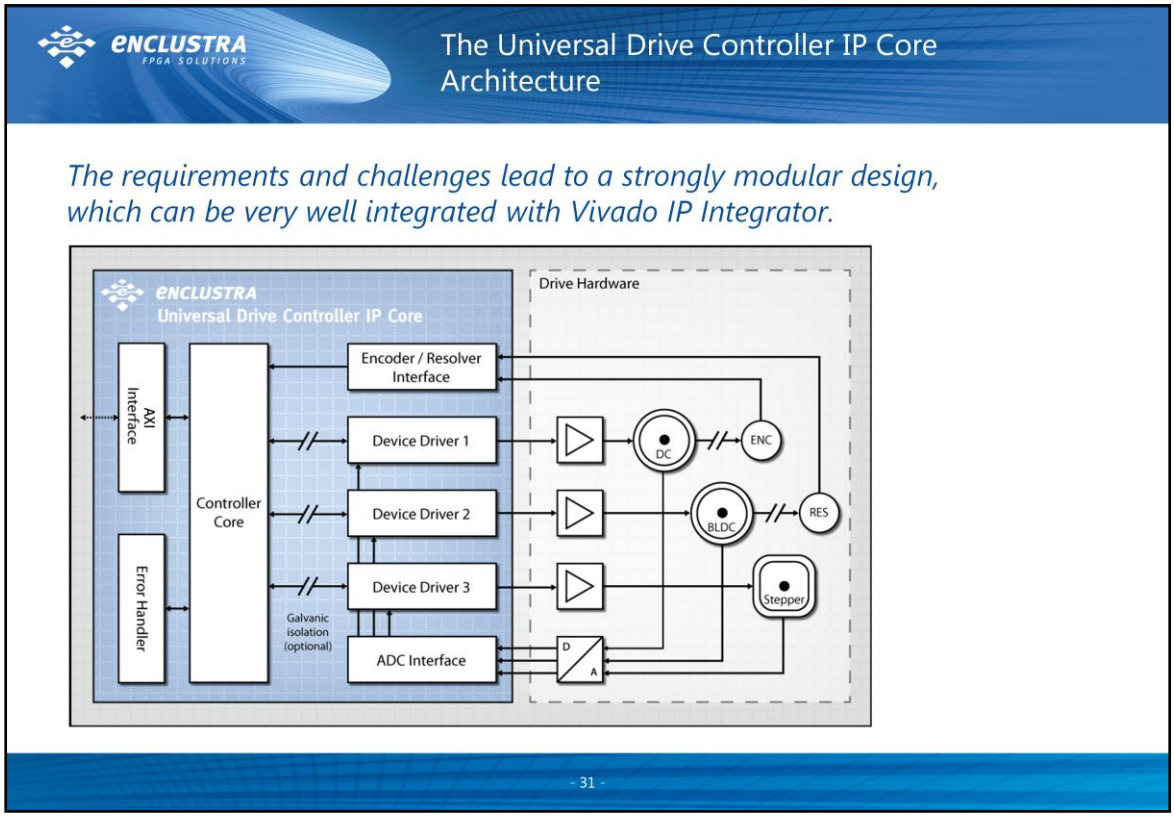

ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 29 -



Short Description

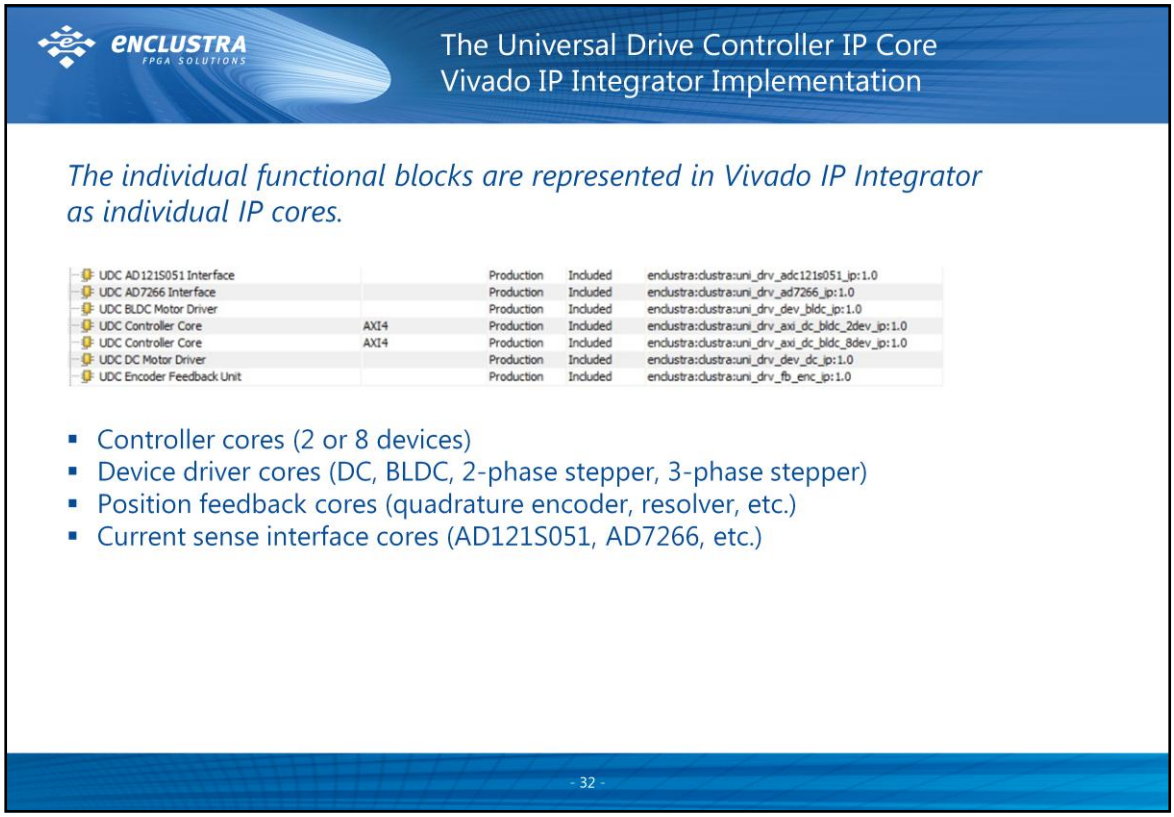
The Enclustra Universal Drive Controller IP Core enables the easy addition of drive control capabilities to existing or future FPGA designs. There is no need for an extra drive controller chip that would consume precious PCB space and unnecessarily extend the project BOM.

Selecting Enclustra's Universal Drive Controller IP Core for the drive control needs of future projects will significantly reduce time to market as well as the overall system cost.



Architecture

The highly modular approach enables the user to build exactly the system he needs – for example a system with two BLDC and one stepper motor, sharing a multi-channel ADC device for current sense and providing full position control functionality for all three axes.

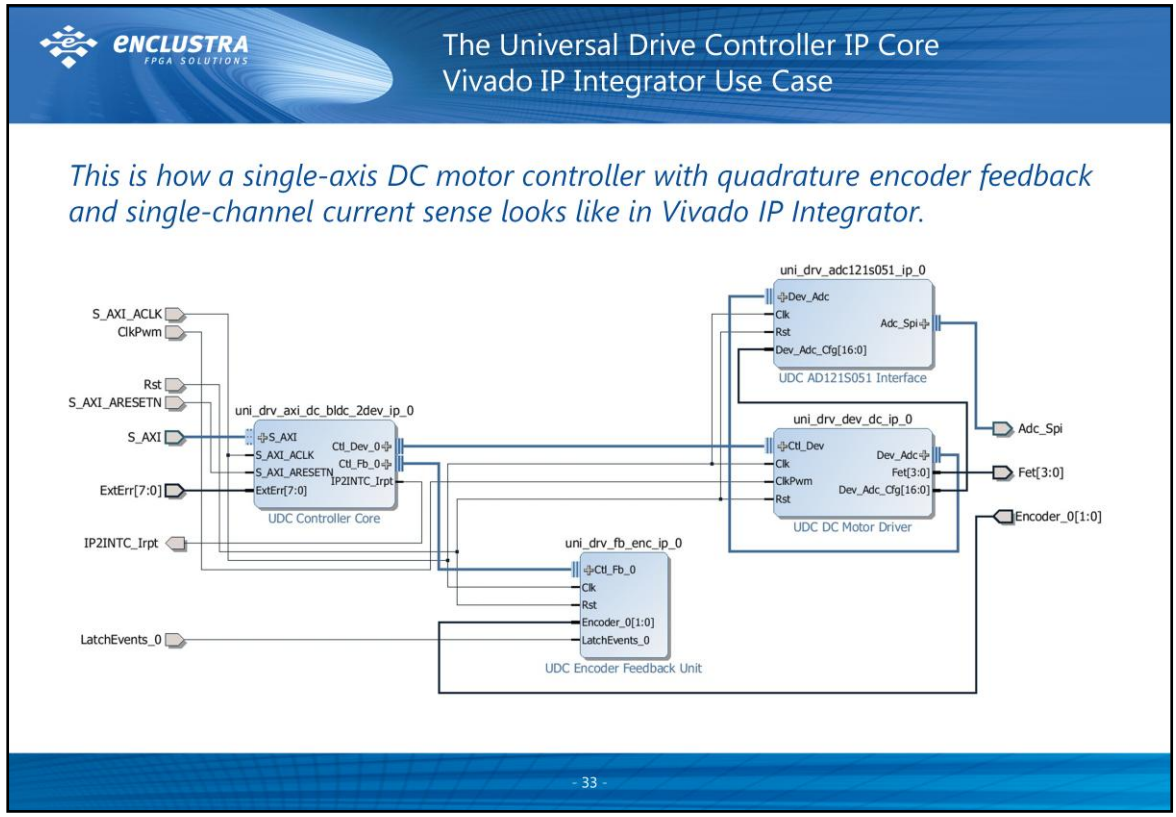


Modular Construction System

The Universal Drive Controller IP Core is basically a modular construction system for drive control applications.

The central component is the controller core, which provides the TDM structures for calculating the cascaded PID controllers for the current, velocity and position control loops for up to 8 axis. The complete control loop timings are also controlled by the controller core.

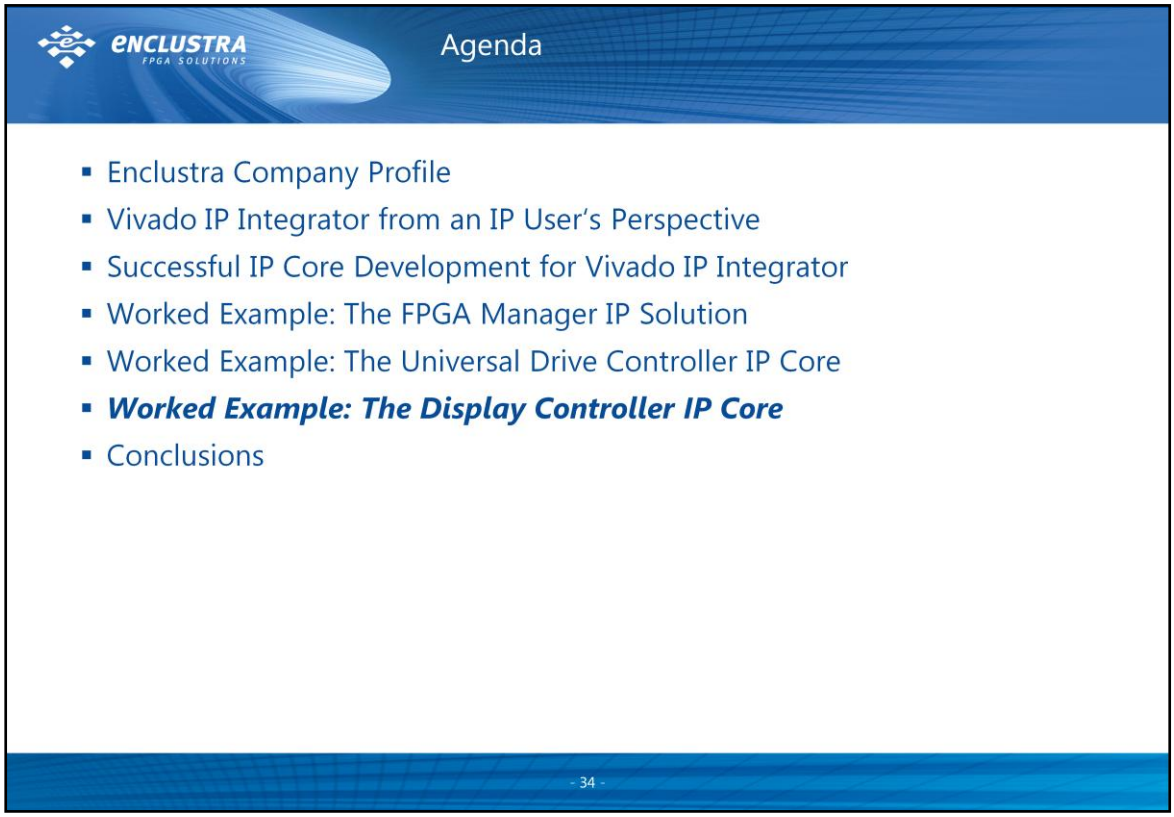
The individual motor types (DC, BLDC, 2-phase stepper, 3-phase stepper) and feedback devices (encoder, resolver, ADCs) are controlled by individual device driver cores with unified interfaces to the central controller core.

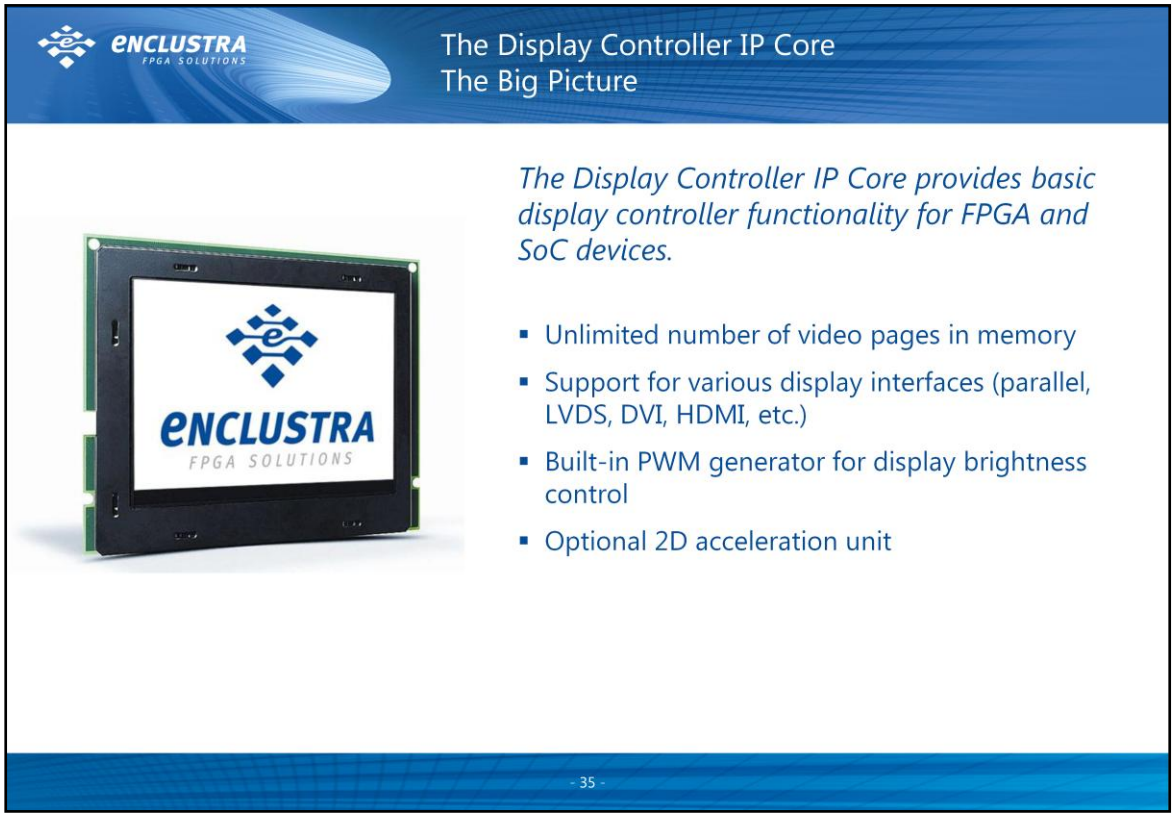


Example Design

The slide above shows an example design for a position-controlled DC motor with quadrature encoder feedback and single-channel current sense – a typical servo drive.

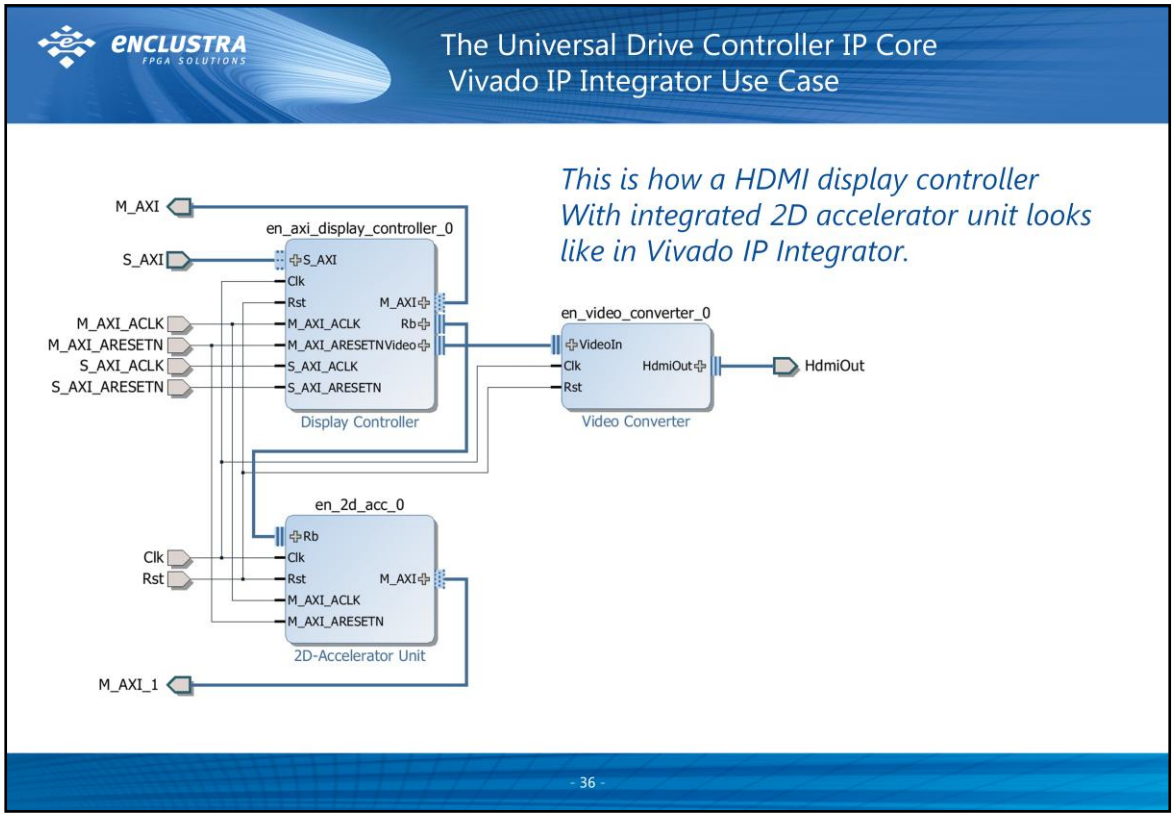
For the ease of use, the interface signals between the central controller core and the device driver cores have been grouped into custom interfaces. This allows the user to connect such interfaces with just one (bus) wire, just as it is possible for AXI4 interfaces.


ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 34 -



Short Description

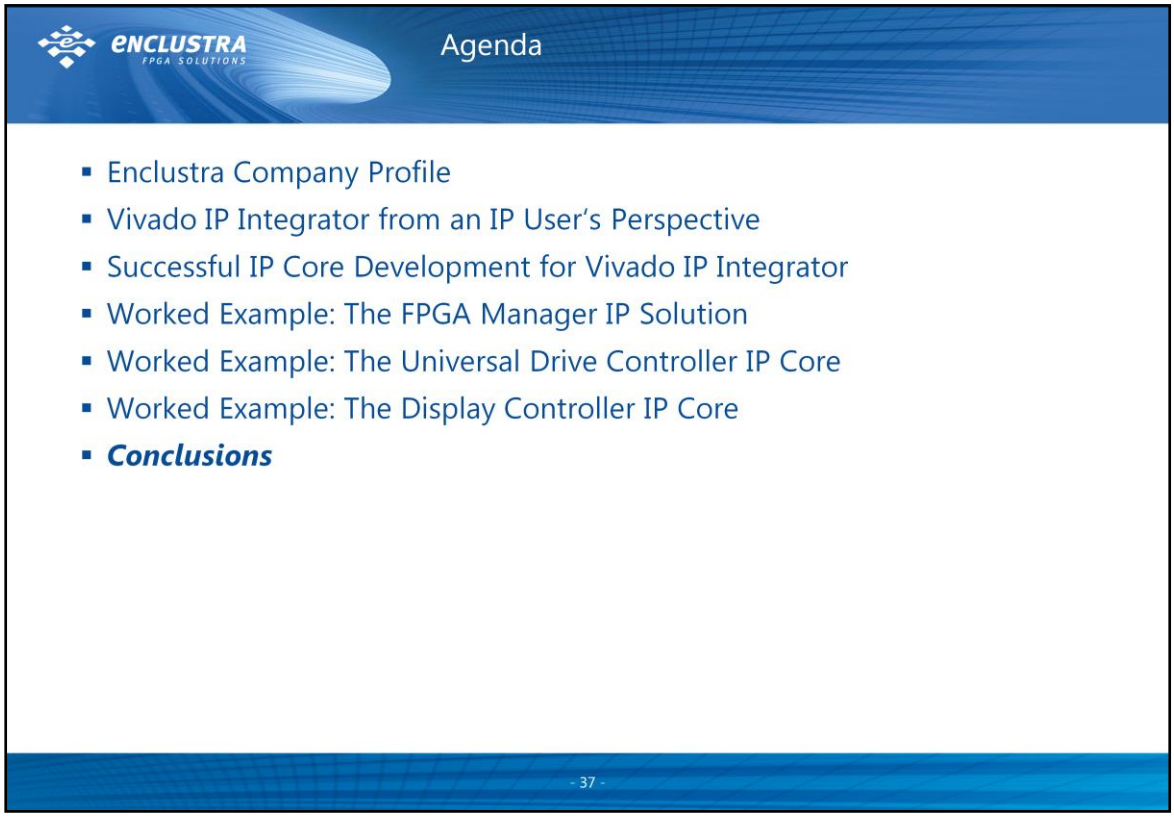
The Display Controller IP Core enables the easy addition of displays to existing or future FPGA and SoC designs. It allows the system designer to focus on the main application instead of dealing with ancillary display control issues. In addition, there is no need for an external display controller device that would consume precious PCB space and unnecessarily extend the project's BOM.



Example Design

The slide above shows an example design for a HDMI display controller with integrated 2D accelerator.

For the ease of use, the internal video and register bank signals have been grouped into custom interfaces. This allows the user to connect such interfaces with just one (bus) wire, just as it is possible for AXI4 interfaces. The same has been done for the external HDMI interface.


ENCLUSTRA
FPGA SOLUTIONS
Agenda
Enclustra Company Profile
Vivado IP Integrator from an IP User's Perspective
Successful IP Core Development for Vivado IP Integrator
Worked Example: The FPGA Manager IP Solution
Worked Example: The Universal Drive Controller IP Core
Worked Example: The Display Controller IP Core
Conclusions
- 37 -

Conclusions

- *Use as little generics/parameters as possible. Go for a blockset instead.*
- *Stick to standard interfaces (AXI4, AXI4-Lite, AXI4-Stream).*
- *Provide timing constraints for your IP.*
- *Add a product guide document to your IP package.*
- *Provide a unit/regression test for your IP. Target the Vivado Simulator.*
- *Provide a reference design for your IP. Target evaluation boards.*
- *Use a version control system. Use Vivado TCL scripting capabilities.*
- *Know the Vivado tools, especially the IP Packager and the IP Integrator.*

- 38 -

ENCLUSTRA FPGA SOLUTIONS

Questions?

Martin Heimlicher
President & Founder
martin.heimlicher@enclustra.com
+41 43 343 39 40

Quarterly newsletter:
subscribe@enclustra.com

Upcoming Events:

- SPS IPC Drives
  November 25-27, 2014
  Messezentrum Nürnberg
  Germany

  sps ipc drives

- Embedded World
  February 24-26, 2015
  Messezentrum Nürnberg
  Germany

  embedded world 2015
  Exhibition&Conference
  ... it's a smarter world

- 39 -

Image References

- *Slide 4 FPGA image and Xilinx Alliance Program logo courtesy of xilinx.com*
- *Slide 4 location pin image (location pin) courtesy of C at clker.com*
- *Slide 5 image (maze) and slide 28 image (3D bar chart) courtesy of ddpavumba at FreeDigitalPhotos.net*
- *Slide 6 image (jigsaw puzzle), slide 16 image (clock) and slide 18 image (right sign) courtesy of digitalart at FreeDigitalPhotos.net*
- *Slide 8 image (hand picking block) courtesy of idea go at FreeDigitalPhotos.net*
- *Slide 9 image (drawing), slide 14 image (exploding blocks) and slide 24 image (ladders) courtesy of Stuart Miles at FreeDigitalPhotos.net*
- *Slides 10 and 15 images (Lego bricks) courtesy of Boaz Yunior Wibowo at dreamstime.com*
- *Slide 11 image (suitcases) courtesy of phasinphoto at FreeDigitalPhotos.net*
- *Slide 12 image (tennis) courtesy of Phaitoon at FreeDigitalPhotos.net*
- *Slide 17 image (books) courtesy of Boykung at FreeDigitalPhotos.net*
- *Slide 19 image (stick figure) courtesy of ikea.ch*
- *Slide 20 image (folders) courtesy of jscreationzs at FreeDigitalPhotos.net*
- *Slide 21 image (box) courtesy of foto76 at FreeDigitalPhotos.net*
- *Slide 26 image (blocks) courtesy of Master isolated images at FreeDigitalPhotos.net*
- *Slide 38 image (lightbulb) courtesy of KEKO64 at FreeDigitalPhotos.net*
- *Slide 39 sps ipc drives logo courtesy of www.mesago.de*
- *Slide 39 embedded world logo courtesy of www.embedded-world.de*
- *All other images courtesy of Enclustra GmbH*

Enclustra GmbH
Technoparkstr. 1 – CH-8005 Zürich – Switzerland
Phone +41 43 343 39 43 – www.enclustra.com