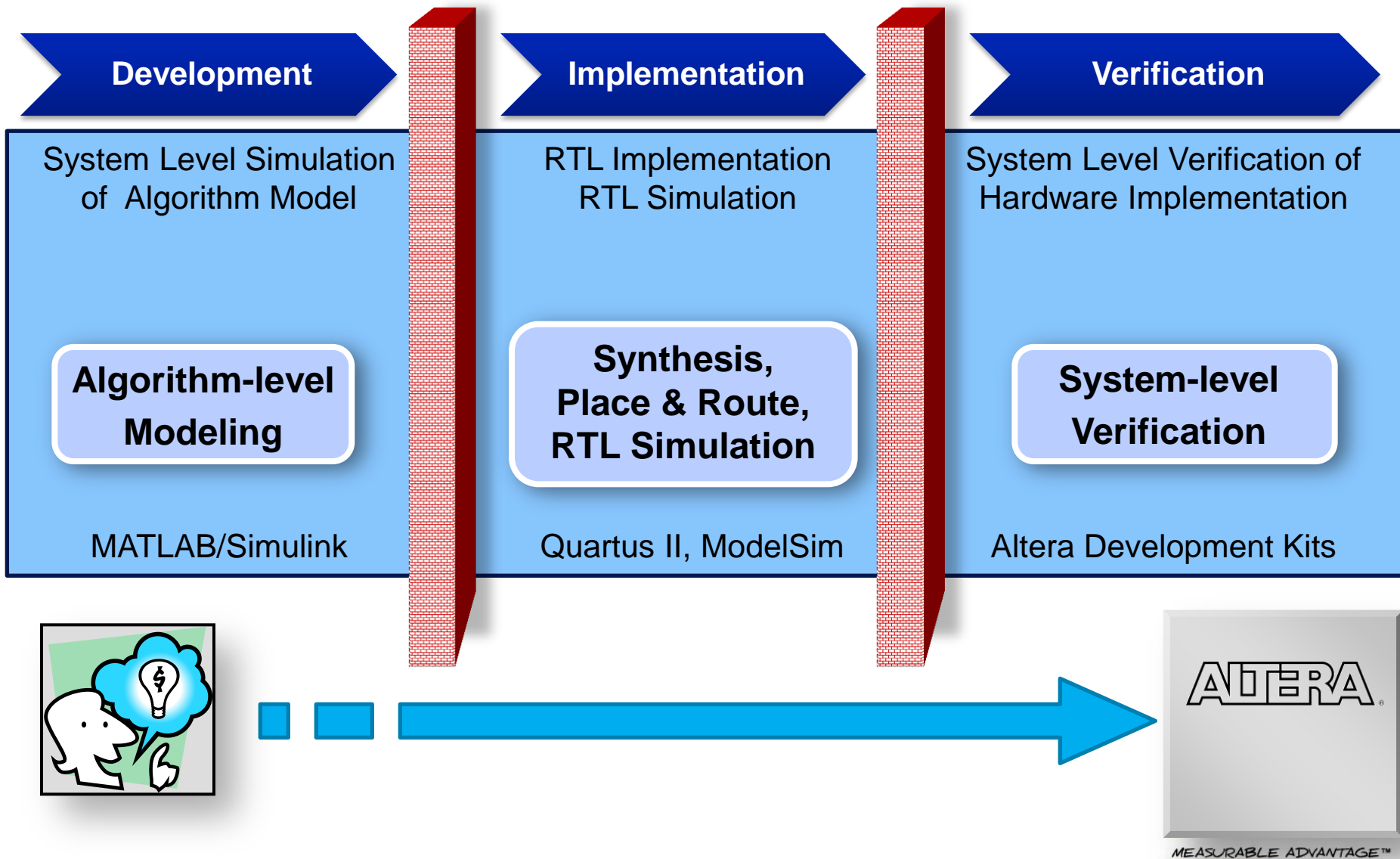




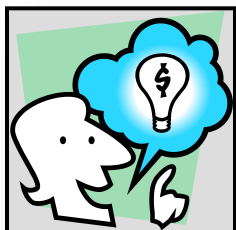
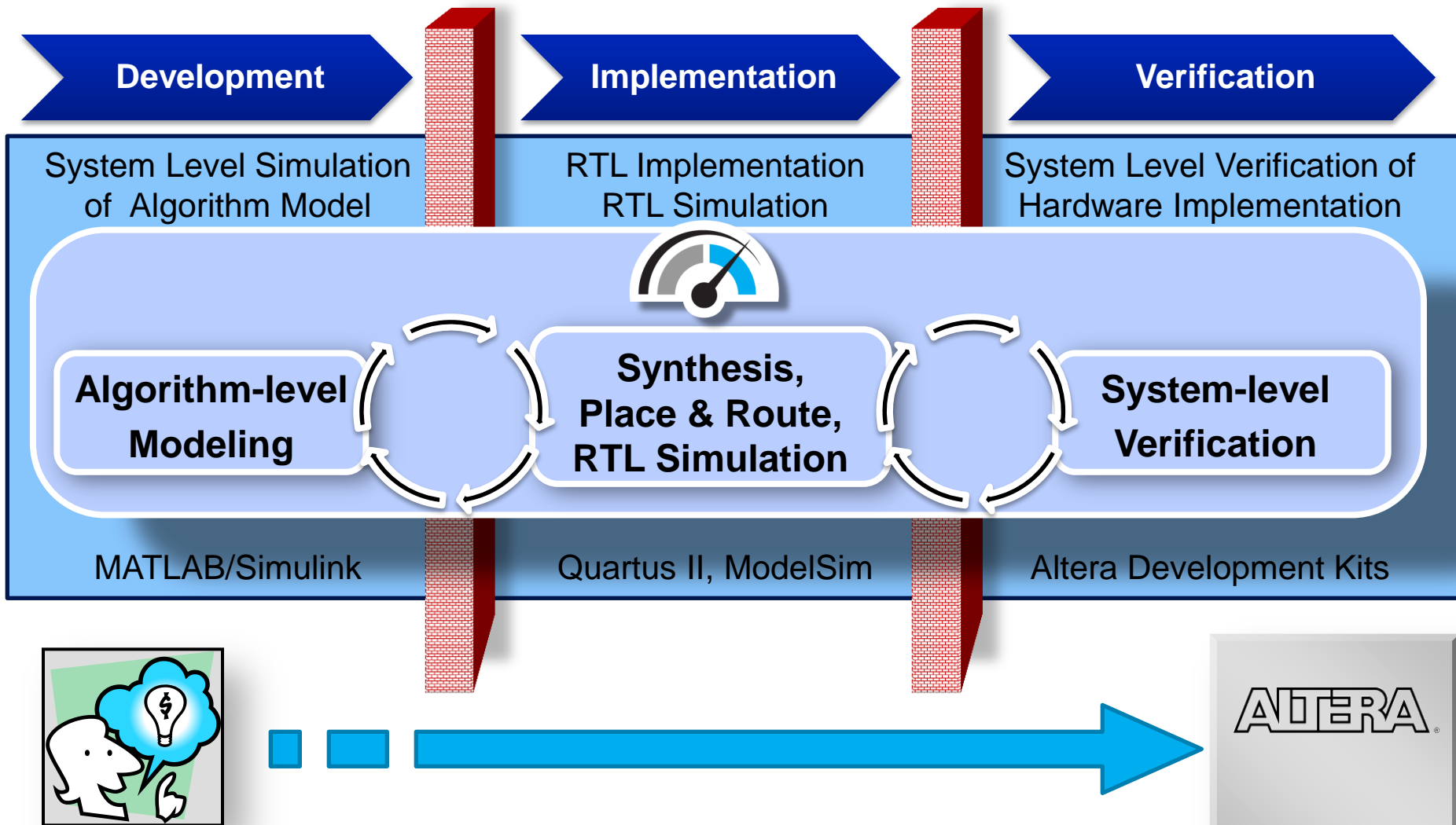
# The New Implementation methodology of FPGA

Dylan Wang

# DSP Builder System Level Design Flow



# DSP Builder System Level Design Flow



MEASURABLE ADVANTAGE™

# Altera Blockset Libraries

## ■ DSP Builder Advanced Blockset

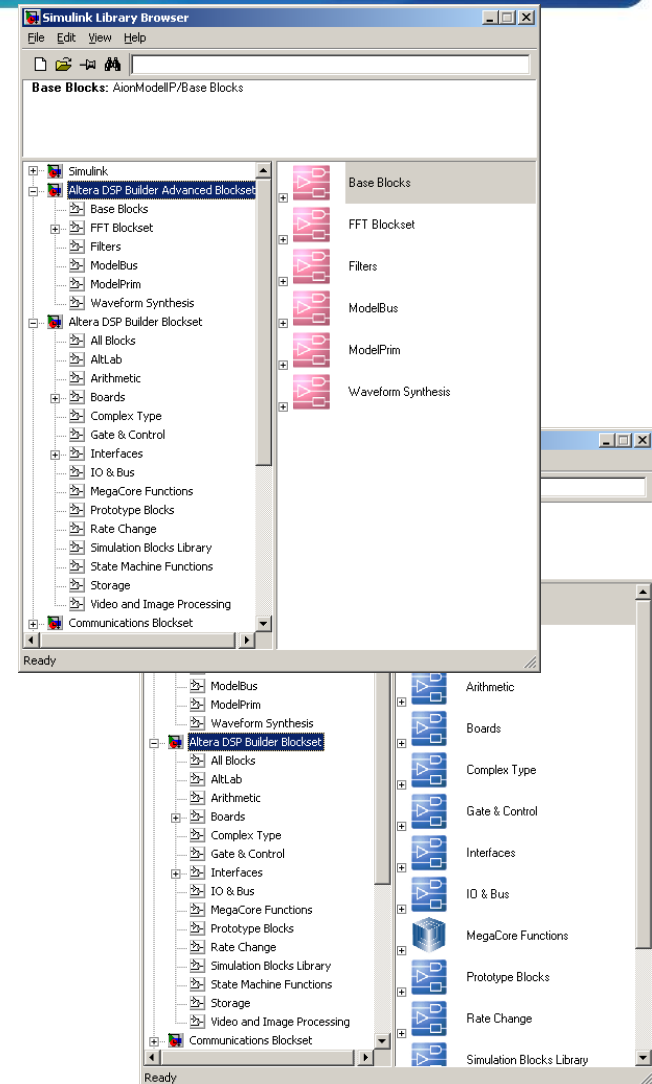
### – is High Level Synthesis Design

- Constraint driven design
- Abstracted, generic build blocks
- Single data path logic system clock
- Automatic pipelining and register balancing
- High data rate support
- Floating point support
- Tool creates the optimized h/w implementation

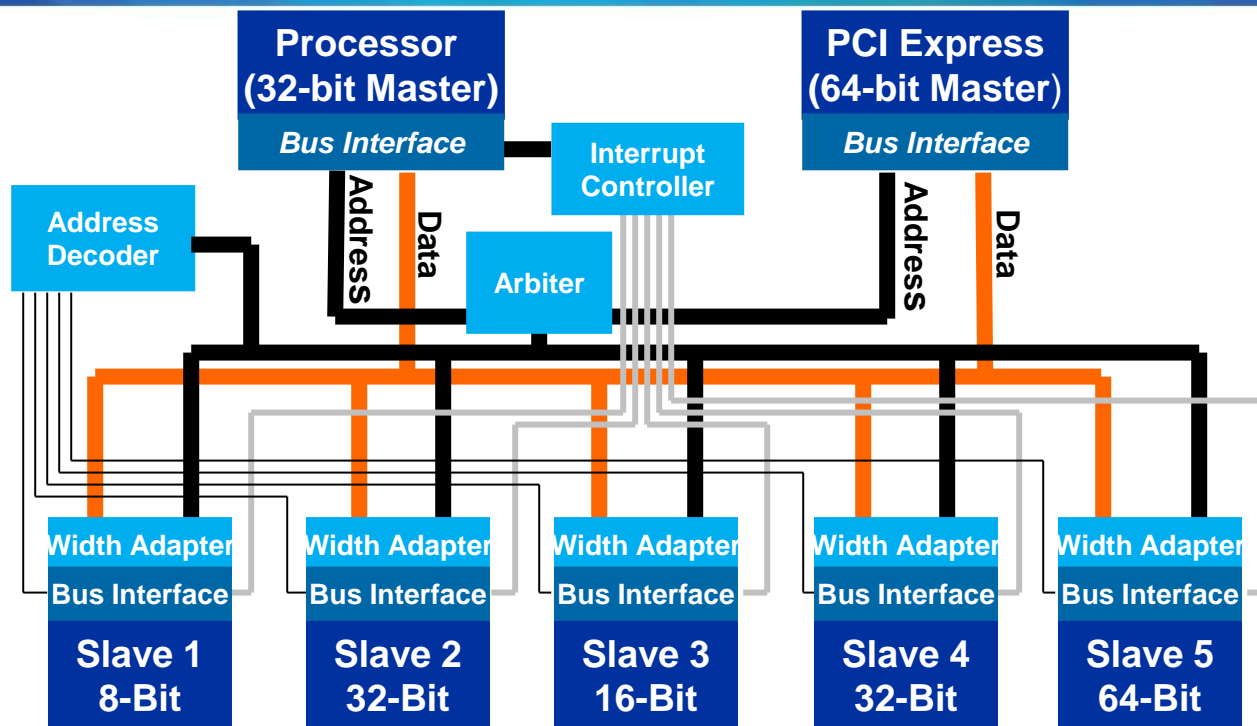
## ■ DSP Builder Standard Blockset

### – is WYSIWYG design

- Structural design
- Hardware-like building blocks
- Multiple clock domain design
- User HDL and DSP IP import
- Hardware Co-simulation
- Enables fine-grain control of h/w implementation

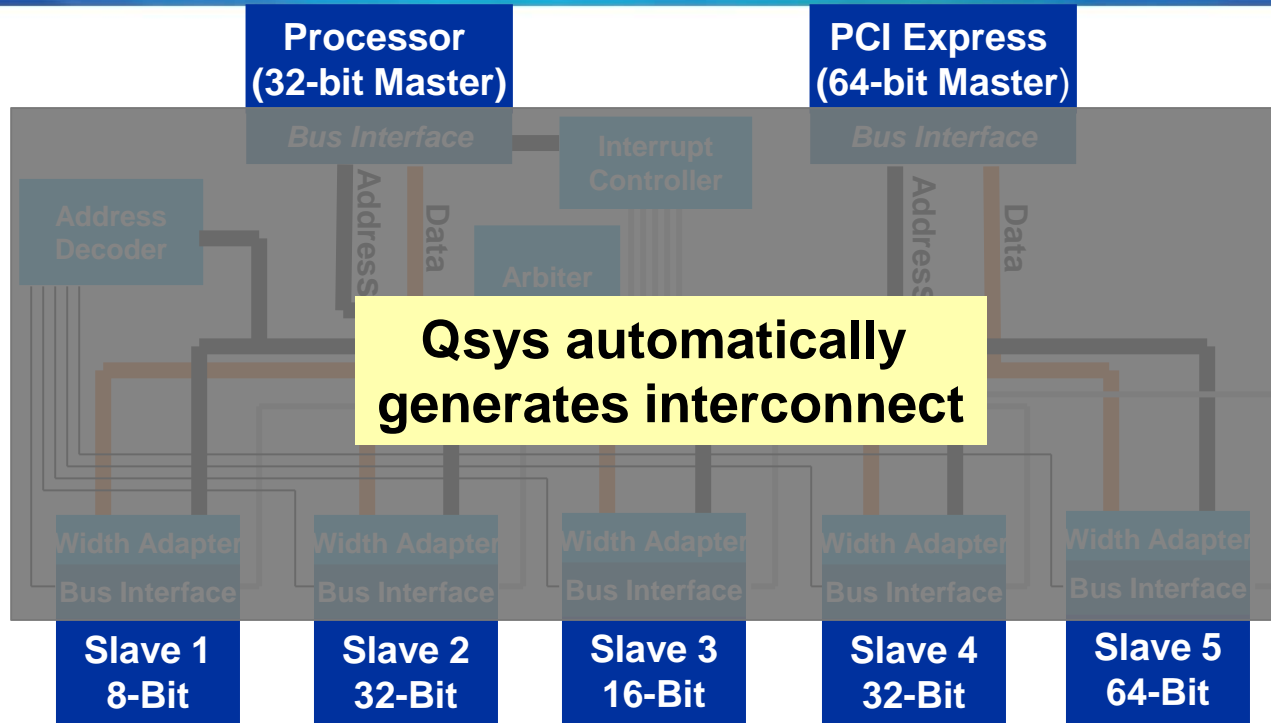


# Traditional System Design



- Components in system use different interfaces to communicate (some standard, some non-standard)
- Typical system requires significant engineering work to design custom interface logic
- Integrating design blocks and intellectual property (IP) is tedious and error-prone

# Automatic Interconnect Generation



- Avoids error-prone integration
- Saves development time with automatic logic & HDL generation
- Enables you to focus on value-add blocks

***Qsys improves productivity by automatically generating the system interconnect logic***

# Introducing Qsys

Qsys - sm\_transfer\_system.qsys (C:\altera\_trn\Working\Introduction\_to\_Qsys\3C25kits\Lab1\sm\_transfer\_system.qsys)

File Edit System View Tools Help

Component Library System Contents Address Map Clock Settings Project Settings Instance Parameters System Inspector HDL Example Generation

Component Library

- Project
  - New Component...
  - System
  - Training
- Library
  - Altera PCIe Reconfig Driver
  - Bridges
  - Clock and Reset
  - Configuration & Programming
  - DSP
  - Embedded Processors
  - Interface Protocols
  - Memories and Memory Controllers
  - Merlin Components
  - Microcontroller Peripherals
  - Peripherals
  - PLL
  - Qsys Interconnect
  - SLS
  - Verification
  - Window Bridge

System Contents

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> <li>clk</li> <li>clk_in</li> <li>clk_in_reset</li> <li>clk</li> <li>clk_reset</li> </ul>	<ul style="list-style-type: none"> <li>Clock Source</li> <li>Clock Input</li> <li>Reset Input</li> <li>Clock Output</li> <li>Reset Output</li> </ul>	<ul style="list-style-type: none"> <li>clk_clk_in</li> <li>clk_clk_in_reset</li> <li>Double-click to export</li> <li>Double-click to export</li> </ul>	clk			
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> <li>p1l</li> <li>inclk_interface</li> <li>inclk_interface_reset</li> <li>p1l_slave</li> <li>c0</li> <li>c1</li> <li>locked_conduit</li> <li>phasedone_conduit</li> </ul>	<ul style="list-style-type: none"> <li>Avalon ALTPLL</li> <li>Clock Input</li> <li>Reset Input</li> <li>Avalon Memory Mapped Slave</li> <li>Clock Output</li> <li>Clock Output</li> <li>Conduit</li> <li>Conduit</li> </ul>	<ul style="list-style-type: none"> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> </ul>	clk	0x0000_0010	0x0000_001f	
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> <li>start_pushbutton</li> <li>clk</li> <li>reset</li> <li>s1</li> <li>external_connection</li> </ul>	<ul style="list-style-type: none"> <li>PIO (Parallel I/O)</li> <li>Clock Input</li> <li>Reset Input</li> <li>Avalon Memory Mapped Slave</li> <li>Conduit Endpoint</li> </ul>	<ul style="list-style-type: none"> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> </ul>	sys_clk	0x0000_0060	0x0000_006f	
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> <li>av_sm_master</li> <li>clock</li> <li>reset</li> <li>avalon_master</li> </ul>	<ul style="list-style-type: none"> <li>Avalon State Machine Master</li> <li>Clock Input</li> <li>Reset Input</li> <li>Avalon Memory Mapped Master</li> </ul>	<ul style="list-style-type: none"> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> </ul>	sys_clk			
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> <li>led_out</li> <li>clock</li> <li>reset</li> <li>st_sink</li> <li>conduit_end</li> </ul>	<ul style="list-style-type: none"> <li>LED Flasher</li> <li>Clock Input</li> <li>Reset Input</li> <li>Avalon Streaming Sink</li> <li>Conduit</li> </ul>	<ul style="list-style-type: none"> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> </ul>	sys_clk			
<input checked="" type="checkbox"/>		<ul style="list-style-type: none"> <li>led_timing_adapter</li> <li>clk</li> <li>reset</li> <li>in</li> <li>out</li> </ul>	<ul style="list-style-type: none"> <li>Avalon-ST Timing Adapter</li> <li>Clock Input</li> <li>Reset Input</li> <li>Avalon Streaming Sink</li> <li>Avalon Streaming Source</li> </ul>	<ul style="list-style-type: none"> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> <li>Double-click to export</li> </ul>	sys_clk			

Messages

Description	Path
7 Warnings	
4 Info Messages	
0 Errors, 7 Warnings	

# OpenCL (Open Computing Language) Overview

## ■ Software programming model:

- C/C++ API for host program
- OpenCL C for acceleration device

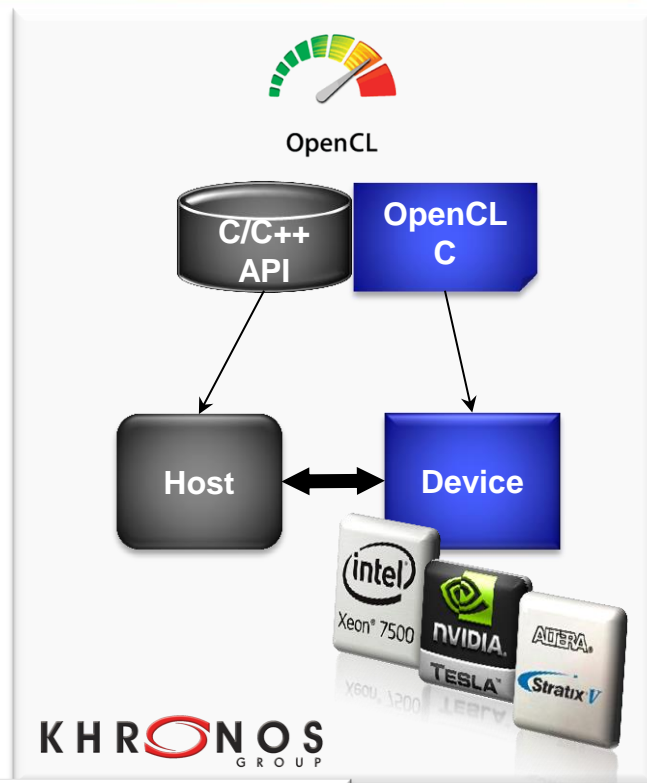
## ■ Provides increased performance with hardware acceleration

- CPU offload to appropriate accelerator
  - Local Memory
  - Explicit Parallelism
    - Task (SMT)
    - Data (SPMD)

Low Level Programming Language!

## ■ Open, royalty-free, standard

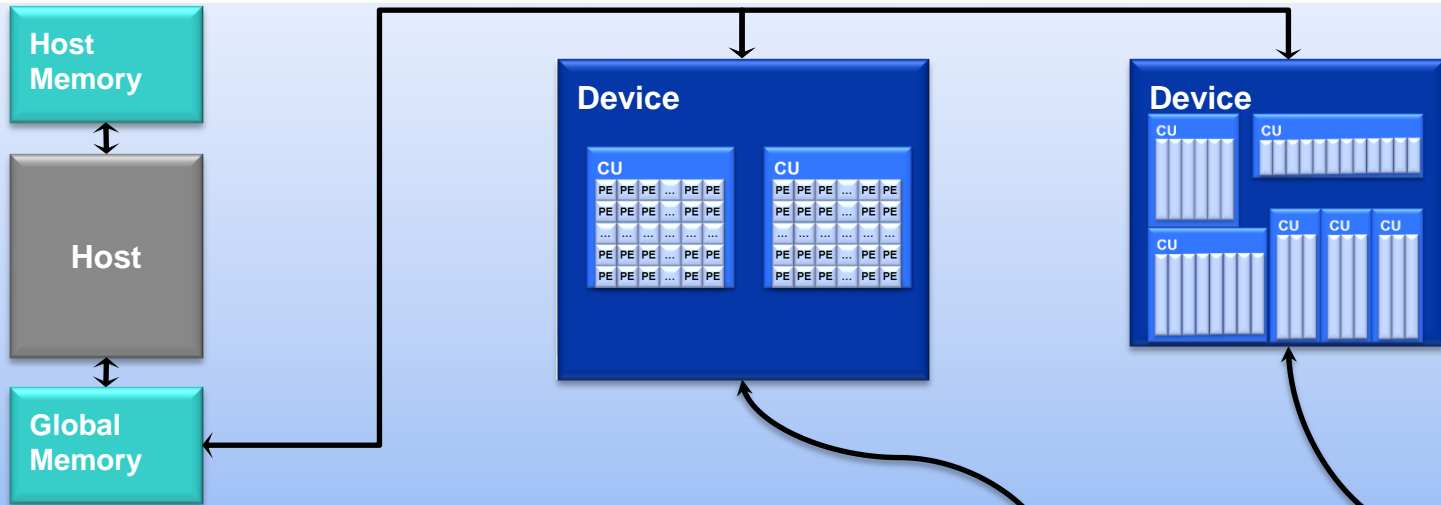
- Managed by Khronos Group
- Altera active member
- Conformance requirements
  - V1.0 is current reference
  - V2.0 is current release
- <http://www.khronos.org>



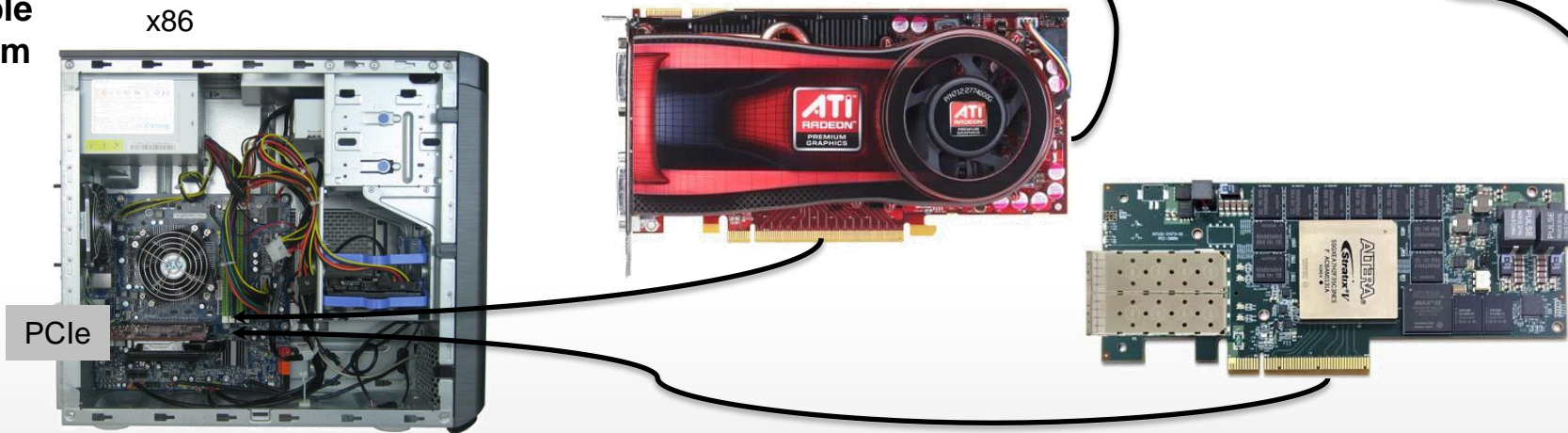


# Heterogeneous Platform Model

OpenCL Platform Model



Example Platform

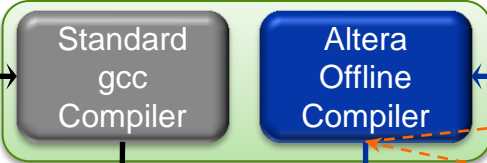


# Use Model: Abstracting the FPGA away

```
main() {  
  read_data( ... );  
  manipulate( ... );  
  clEnqueueWriteBuffer( ... );  
  clEnqueueNDRange(..., sum,...);  
  clEnqueueReadBuffer( ... );  
  display_result( ... );  
}
```

OpenCL  
Host Program + Kernels

```
__kernel void sum  
(__global float *a,  
 __global float *b,  
 __global float *y)  
{  
  int gid = get_global_id(0);  
  y[gid] = a[gid] + b[gid];  
}
```



EXE

AOCX

Verilog

Quartus II



# What happens in future

- **Frame work + core processing**
- **Auto HDL code generating**
- **Design based on prototype tools and diagram**
- **Have the chip design without deeply understanding of the device**



**Thank You**

© 2014 Altera Corporation—Confidential

ALTERA, ARRIA, CYCLONE, ENPIRION, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at [www.altera.com/legal](http://www.altera.com/legal).

**ALTERA**<sup>®</sup>  
*MEASURABLE ADVANTAGE*<sup>™</sup>