

FPGA-Based Hardware Accelerators for 10/40 GigE TCP/IP and Other Protocols

Dr. Endric Schubert, Univ. Ulm / Missing Link Electronics

▶ Ulrich Langenbach, Fraunhofer Heinrich-Hertz-Institute

We are

a Silicon Valley based technology company with offices in Germany. We are partner of leading electronic device and solution providers and have been enabling key innovators in the automotive, industrial, test & measurement markets to build better Embedded Systems, faster.

Our Mission is

To develop and market technology solutions for Embedded Systems
Realization via pre-validated IP and expert application support, and to combine off-the-shelf FPGA devices with Open-Source Software for dependable, configurable Embedded System platforms

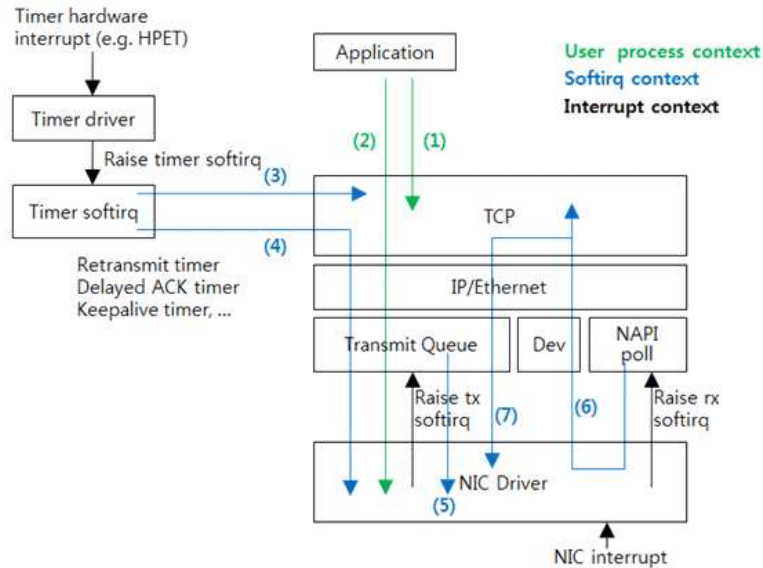
Our Expertise is

I/O connectivity and acceleration of data communication protocols, additionally opening up FPGA technology for analog applications, and the integration and optimization of Open Source Linux and Android software stacks on modern extensible processing architectures.

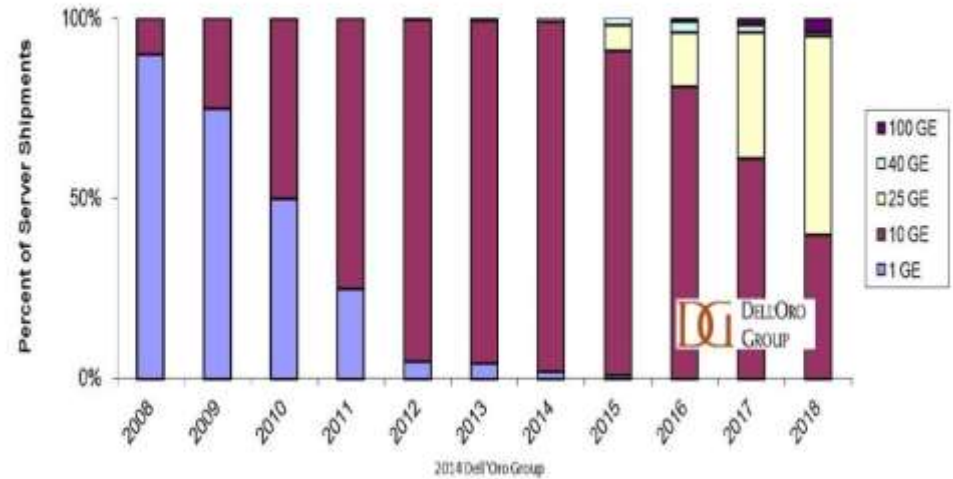
Network Processing at 10 GigE has a Huge Compute Burden ...

Transporting 1 bit per second needs 1 Hz

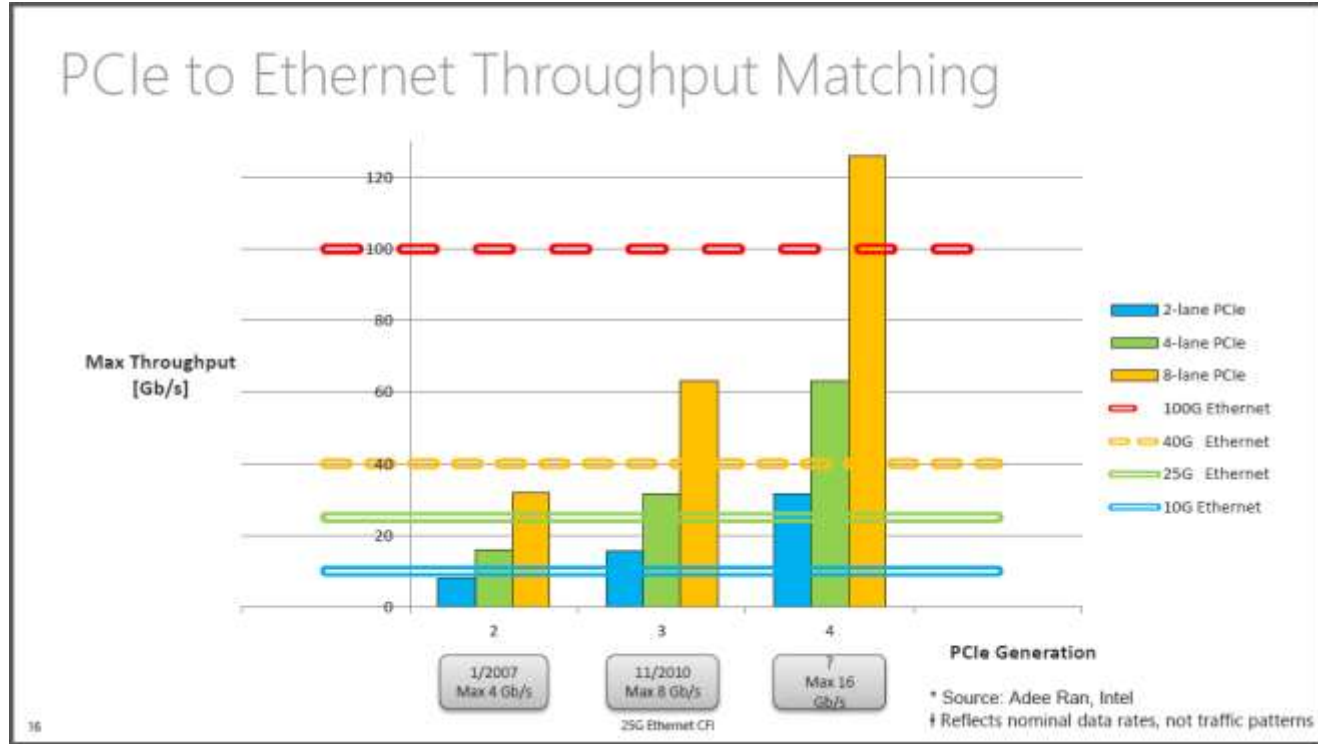
- 1 GigE → 1 CPU at 1 GHz
- 10 GigE → 4 CPUs at 2.5 GHz



Cloud Adoption of 25 GE in Stand-Alone Servers



... and soon we will see 25 GigE

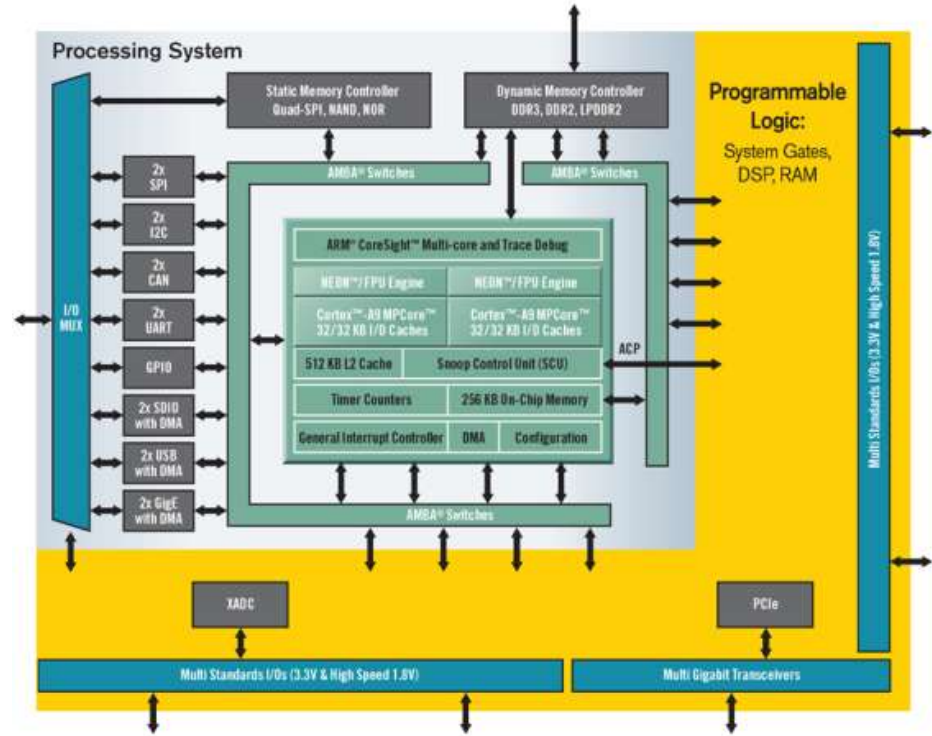


Design Choices for Network Processing in SoC FPGAs

SoC FPGA as (yet) another computer

	Intel i7-4770	Xilinx Zynq 7045
Compute	~100 GFLOPS	5 GFLOPS (PS) 778 GFLOPS (PL)
TDP	84 W	<20 W (typ)

SOC FPGA has 4x more compute
With 1/4 the power dissipation!



[<http://www.xilinx.com/products/technology/dsp.html>]

Network Stack in RTL from Fraunhofer Heinrich-Hertz-Institute

- Brings full TCP/UDP/IP connectivity to FPGAs even when there is no CPU available. Accelerate CPUs by offloading TCP/UDP/IP processing into programmable logic.

Hardware Accelerated Internet Protocol **High Speed Hardware Architectures**

Timeline: 2004 → 2008/09 → 2010 → 2012

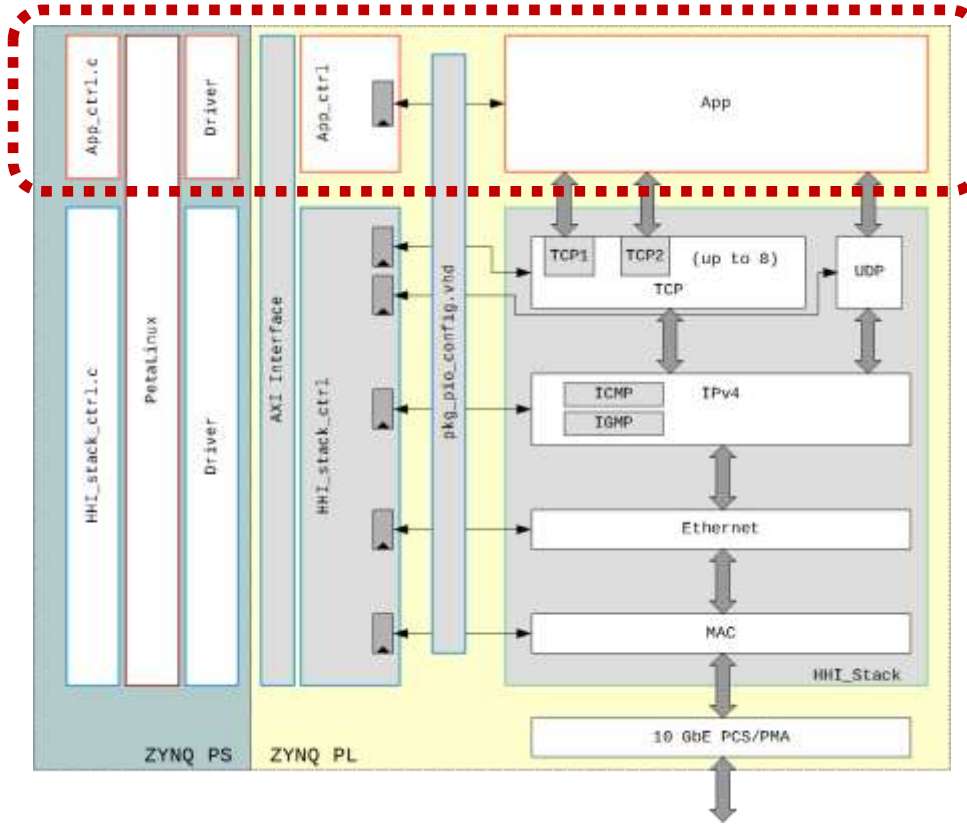
Year	Key Achievements
2004	<ul style="list-style-type: none">Mask-less lithography systemsPublished in 2006XILINX VIRTEX-II
2008/09	<ul style="list-style-type: none">1GbE TCP/IP stackDemonstrated at 2009 IFAUncompressed full HD video transfer
2010	<ul style="list-style-type: none">10GbE TCP/IP stackUncompressed full HD video transferMask-less electron beam lithography
2012	<ul style="list-style-type: none">10GbE TCP/IP stackPCIe IP coreUncompressed full HD video transferHigh Frequency TradingHigh Performance ComputingMask-less electron beam lithography

Fraunhofer Research Hertz Institute
contact-HS@hhi.fraunhofer.de
2013-05-20 14

Hardware Accelerated Internet Protocol **High Speed Hardware Architectures**

Fraunhofer Research Hertz Institute
contact-HS@hhi.fraunhofer.de
2013-05-20 23

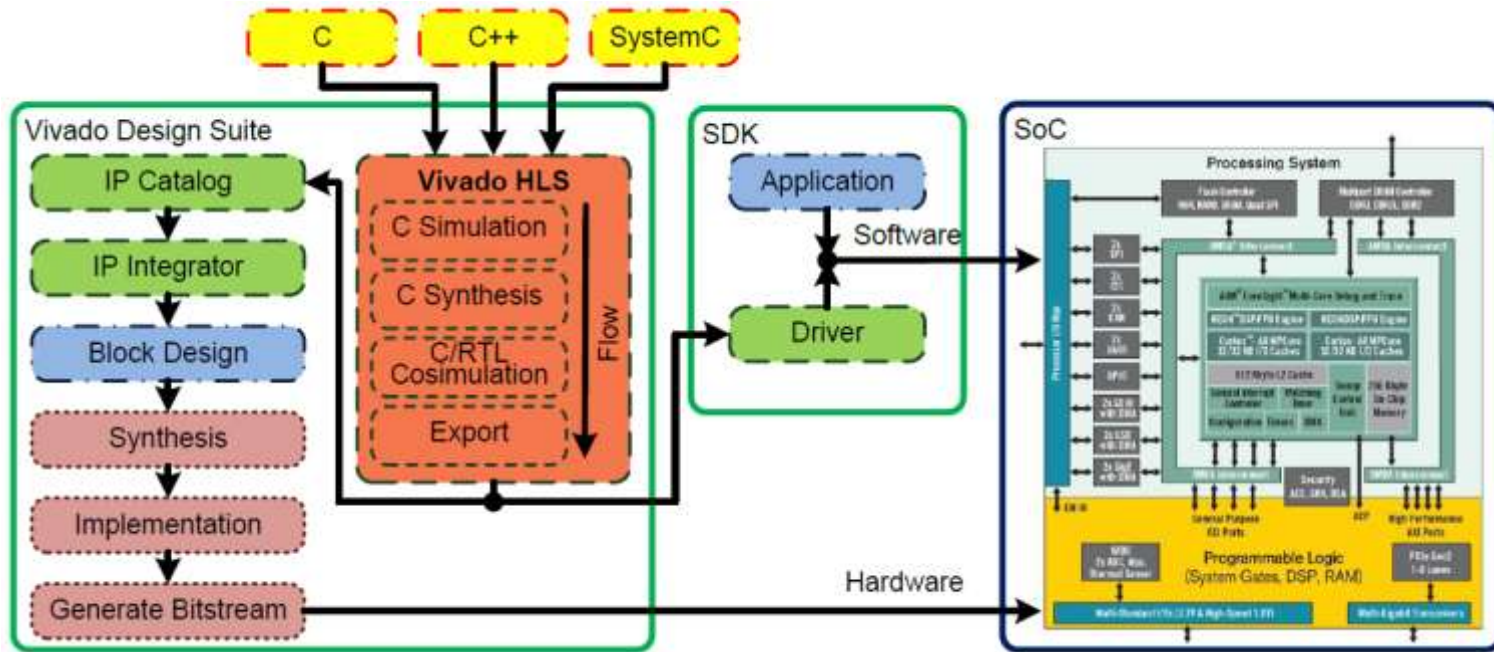
Network Protocol Acceleration Platform Architecture



Network protocol processing at application layer (ISO Layer 7) can more efficiently be implemented via a programming approach (in C or C++) than by digital circuit design (in VHDL or Verilog).

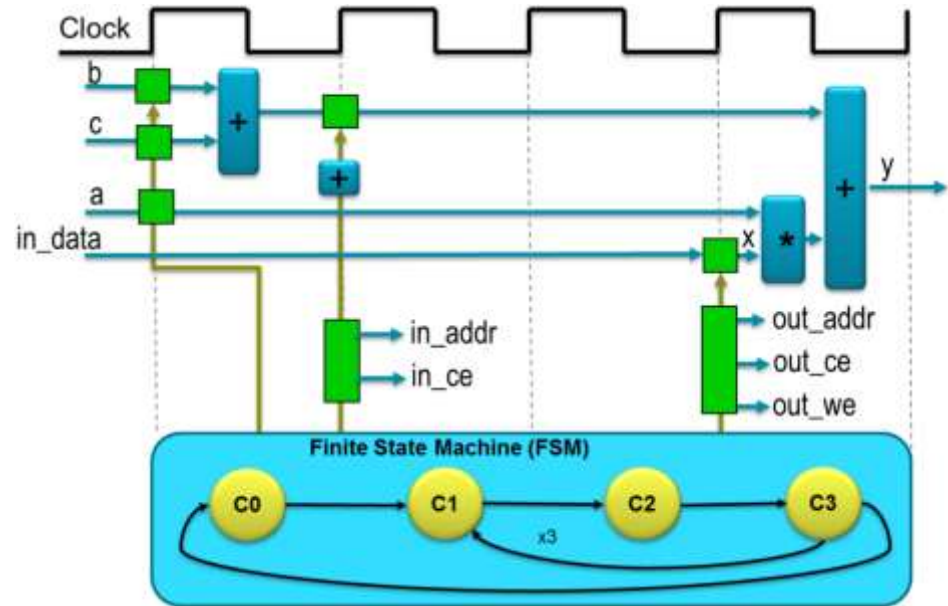
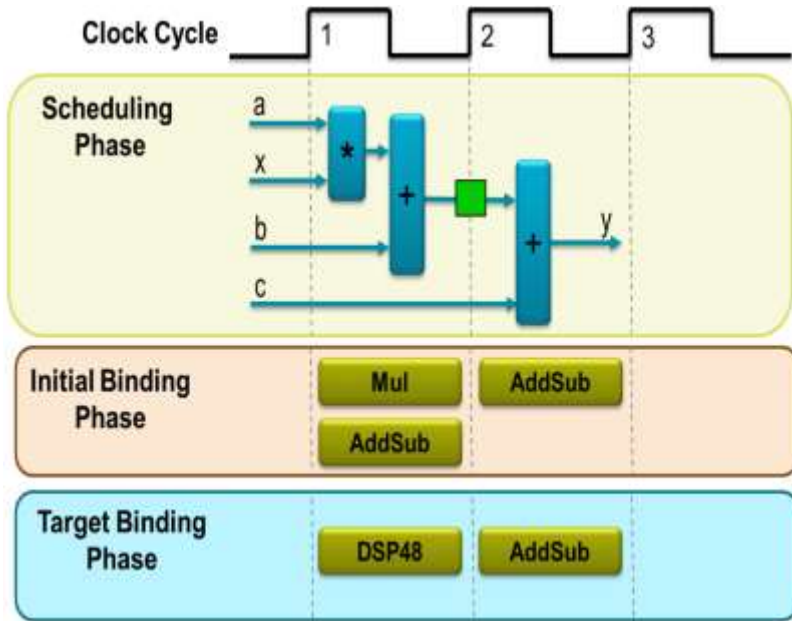
High-Level Synthesis Design Flow for SoC FPGA

- Input C/C++/SystemC into High-Level Synthesis to generate VHDL/Verilog code



Working Principles of High-Level Synthesis

- Design automation runs scheduling and resource allocation to generate RTL code comprising data path plus state machines for control.

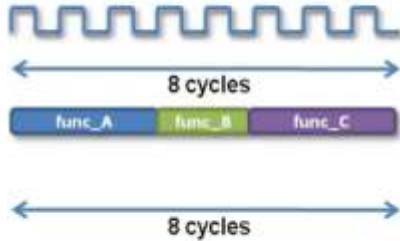
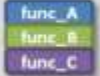


Benefits of High-Level Synthesis

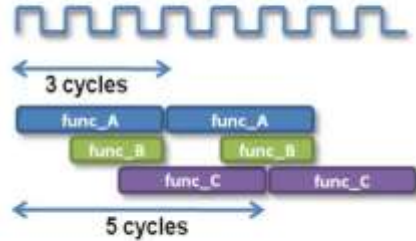
- Automatic performance optimization via parallelization at dataflow level

```

void top (a,b,c,d) {
    ...
    func_A(a,b,11);
    func_B(c,11,12);
    func_C(12,d);
    return d;
}
    
```



(A) Without Dataflow Pipelining



(B) With Dataflow Pipelining

- Automatic interface synthesis and code generation for variety of real-life HW/SW connectivity

Bus Interfaces

Argument Type	Variable			Pointer Variable			Array			Reference Variable		
	Pass-by-value			Pass-by-reference			Pass-by-reference			Pass-by-reference		
	I	IO	O	I	IO	O	I	IO	O	I	IO	O
Interface Type	I	IO	O	I	IO	O	I	IO	O	I	IO	O
ap_none	D			D						D		
ap_stable	D											
ap_ack												
ap_vid						D						D
ap_ovld						D						D
ap_hs												
ap_memory								D	D	D		
ap_fifo												
ap_bus												
ap_ctrl_none												
ap_ctrl_hs						D						
ap_ctrl_chain												

Supported Interface (Green) Unsupported Interface (Red)

Visualization and User Interaction in High-Level Synthesis Tool

Vivado HLS - ownErode (/home/local/work/florianh/test90_HLS_example/ownErode/ownErode) <@tasse>

File Edit Project Solution Window Help

Debug Synthesis Analysis

Module Hierarchy

	BRAM	DSP	FF	LUT	Latency	Interval	Pip
ownErode	9	0	1913	3294		undef	dat
init	0	0	50	50	0	0	nor
init_1	0	0	26	26	0	0	nor
AXIvideo2Mat_32_1080_1920_16_s	0	0	180	220		undef	nor
Erode_16_16_1080_1920_s	9	0	1526	2617		undef	nor
filter_opr_erode_kernel_16_16_unsigned_char	9	0	980	1859	63~208225	63 ~ 208225	nor
getStructuringElement_unsigned_char_int_in	0	0	469	756		undef	nor
Mat2AXIvideo_32_1080_1920_16_s	0	0	57	111	1~2076841	1 ~ 2076841	nor

Performance Profile

	BRAM	DSP	FF	LUT	Bits P0	Bits P1	Bits P2	Banks/Depth
ownErode	9	0	1913	3294				
I/O Ports(16)					152			
Instances(5)	9	0	1839	3024				
Memories(0)	0	0	0	0			0	
Expressions(3)	0	0	0	6	3	3	0	
Registers(14)			14	14				
FIFO(12)	0	60	264	120			18	
Multiplexers(0)	0	0	0	0				

Performance - ownErode

Current Module : ownErode

Operation/Control Step	C0	C1	C2	C3	C4	C5
cols_read(wire_read)						
rows_read(wire_read)						
init(function)						
init_1(function)						
AXIvideo2Mat_32_1080_1920_16_s						
Erode_16_16_1080_1920_s						
Mat2AXIvideo_32_1080_1920_16_s						

Performance Resource

Conclusion and References

- Significant productivity increase for protocol oriented or dataflow based design blocks.
- Easy to adopt: Known languages C/C++ combined with known tool chain.
- → Add this to your bag of tricks!
- UG998 - Introduction to FPGA Design Using High-Level Synthesis
- UG871 - Vivado Design Suite Tutorial: High-Level Synthesis
- XAPP1209 - Designing Protocol Processing Systems with Vivado High-Level Synthesis
- UG949 - UltraFast Design Methodology Guide for the Vivado Design Suite

Contact Information

Dr. Endric Schubert

endric.schubert@uni-ulm.de

endric@MLEcorp.com

Phone US: +1 (408) 320-6139

Phone DE: +49 (731) 141149-66

