

SEmulation: Turbocharging the FPGA Development Process

Introduction

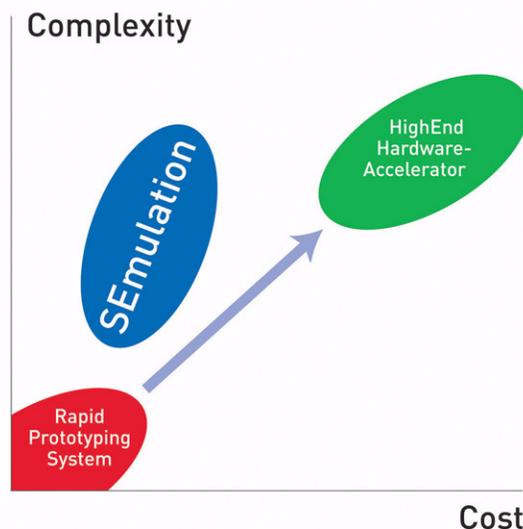
With the SEmulator®, Gleichmann Electronics Research introduces a new method of FPGA/ASIC design, which promise shorter development times and higher design security at a lower cost. With complex processor systems, complete interface structures and design, and up to 3 million ASIC gates per chip, FPGAs dominate the semiconductor market with their flexibility and reconfigurable architecture. Time pressures on the engineer with a short design cycle do not allow for failures. Every revision costs time and money, so the functions and responsiveness of the finished semiconductors have to be right the first time after silicon synthesis.

Today's FPGAs are developed in two steps:

1. HDL-functional blocks are simulated, individually at first, then as a whole.
2. The FPGA design is synthesized and tested in a rapid prototyping system.

SEmulation, simulator-controlled emulation, combines these two steps and allows the step-by-step transfer of the functional blocks from the simulator (software) into the FPGA (hardware), without leaving the simulation environment and thus shortening the development time. Currently, the simulation and emulation environments are separate and not always compatible, which results in wasted engineering time getting simulated code to run in the emulator environment. Using the SEmulator, mistakes and insecurities are eliminated as simulation takes place in the target hardware. The SEmulator also allows the engineer to introduce any external component into the simulation (also known as “hardware in the loop”). For the developer, this means functional first silicon, with a reduction of development costs, as well as an increased responsiveness to market requirements. [Figure 1](#) shows the cost and complexity benefits of SEmulation.

Figure 1. Commercial Position of SEmulation



SEmulator—Bridging the Gap Between Simulation and Emulation

The SEmulator, named for a made-up word that combines the words simulator and emulator, provides bridging functionality between the domain of digital hardware simulation and the world of FPGA prototyping. Design blocks can easily be moved between these two domains.

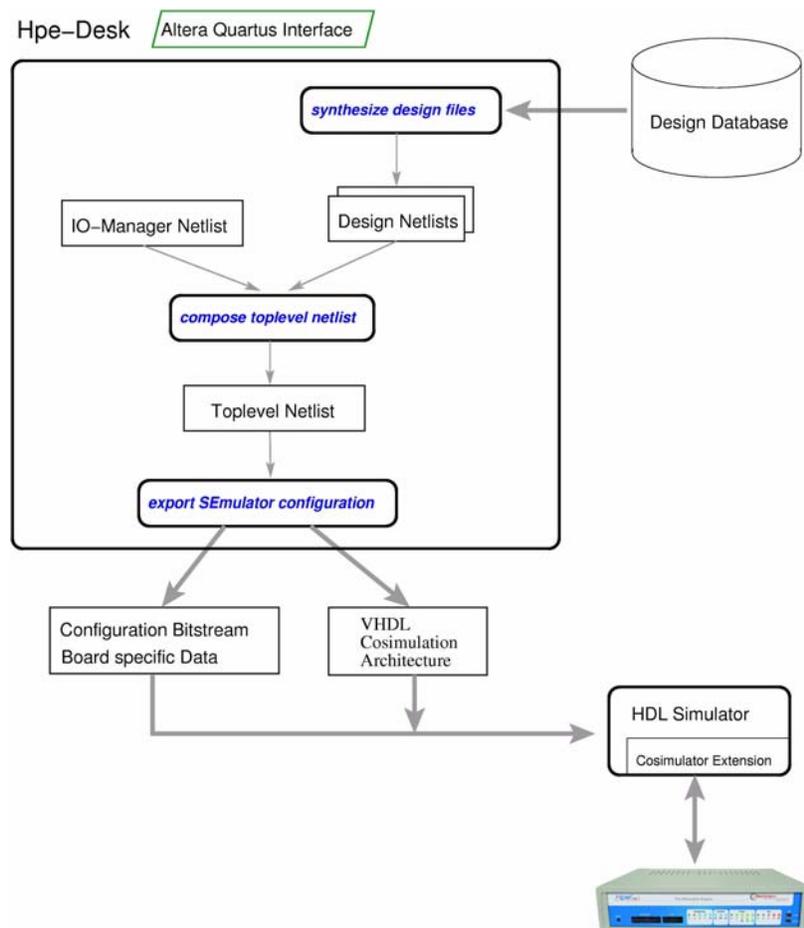
Nowadays digital hardware is described using hardware description languages (HDL) like VHDL or Verilog. These descriptions are simulated together with a problem-specific test bench (also written in an HDL). The initial functional verification of this design is done via simulation. This approach is a must for the complex designs that are developed today.

However, the real goal for the hardware designer is not the simulation; it is running the design on real hardware. The step from simulation to the real hardware prototype is a huge one and should not be underestimated. When the simulation results are satisfactory, the designer takes the whole design and downloads the synthesized netlist into his FPGA prototyping board. Then the designer needs a logic analyzer and plenty of time to locate and fix any bugs in his design.

The SEmulator approach enables the designer to use the prototyping FPGA board at an early state of the design flow. The SEmulator allows design blocks to be moved into the FPGA and to co-simulate them with the actual developed design blocks in the HDL simulator.

Figure 2 shows the SEmulator principle. The HDL simulator is used for functional verification of the design files; whenever a design file is stable, it can be transferred to the FPGA prototyping system. The transferred block will be co-simulated with the remaining blocks in the simulator. The designer can easily switch between a simulation with the HDL design description and the co-simulation with the real hardware located on the FPGA prototyping system. This approach allows the use of a real hardware platform in a very early design stage without much effort.

Figure 2. SEmulator Design Flow

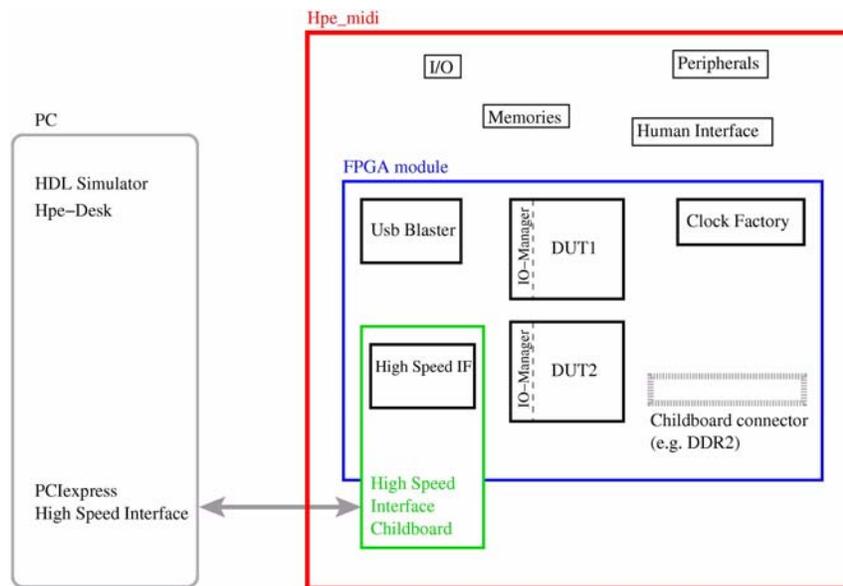


When all design blocks are handled by the HDL simulator, it is called simulation. When the whole design is transferred to the FPGA prototyping board, it can be executed at full clock speed and is called emulation. Mixing these two modes (having some blocks synthesized on the FPGA and others as simulation models in the simulator) is called SEmulation.

Figure 3 shows the most important parts of the SEmulator hardware platform. The motherboard (Hpe_midi, marked in red) includes the most common useful peripherals like memories, I/O interfaces, and the human interface. This is where the FPGA module boards (device under test (DUT) boards) are inserted. Figure 3 shows a 2-FPGA DUT board (in blue).

- The clock factory allows the selection of a required number of clock sources for the DUT clock inputs.
- The Altera® USB-Blaster™ download cable can be used to configure the DUT devices.
- The child-board connector is used to extend the system with application specific hardware blocks such as DDR2.

Figure 3. SEmulation Hardware Structure



The high-speed interface is achieved by a special child board and is connected to the board via one of the child-board connectors. The high-speed interface (green) communicates with the PC via a PCI Express (PCIe) interface board. It allows the PC software (HDL simulator + Hpe_desk) to communicate with an I/O manager placed in one or more of the DUT FPGAs. This system makes the design blocks in the DUT FPGAs observable and controllable, while the high-speed connection has a positive impact on the simulation performance.

Use Cases for SEmulation

The flexible SEmulator approach provides a wide range of different applications to accelerate the development process and to enhance the design quality. The following points illustrate some possibilities.

A Versatile FPGA Prototyping Environment

The hardware prototyping and emulation (Hpe) family provides a modular FPGA board concept that is well suited for complex high-speed designs.

The Hpe-midi is an expandable modular FPGA prototyping platform. The base board has a broad range of the most popular interfaces, including USB 2.0, Ethernet 10/100/1000, RS232, LIN, and CAN 2B. The board also provides additional memory such as Flash, SRAM, EEPROM and an SD-Card connection. A basic human interface consists of

a 12-key matrix keyboard, several LEDs, and an LCD display connector. For advanced user interface tasks, a VGA interface, a PS2 input connector, and an AC97 sound chip can be used.

Starting with this base board, a FPGA module can be selected with one, two or even four Altera Stratix® II FPGAs (Stratix III FPGA modules will be provided once silicon is available). Missing interfaces or hardware devices are easily added via the high-speed child-board connectors. One child-board connector can hold a DDR2 memory board for huge memory requirements.

Use Real Hardware Early in the Design Flow

The benefit to using real hardware early in the design flow is that the designer can move his design blocks step by step to the FPGA board. Using smaller steps makes debugging easier. Using the simulation test benches is also a big time saver for the designer, as there is no need for different test environments for the simulation and for the FPGA prototyping board.

 Moving a design block to the FPGA prototyping board is not a one-way road. When a bug is found in a synthesized description, it can easily be switched to the simulation model of that block. After the bug is fixed, the block can then be transferred back to the hardware.

Check Different Versions of a Design Block

The DUT FPGA can hold different versions of a design block in parallel, and the designer can easily switch between different versions for debugging purposes.

Co-Simulation With Real Hardware (Hardware in the Loop)

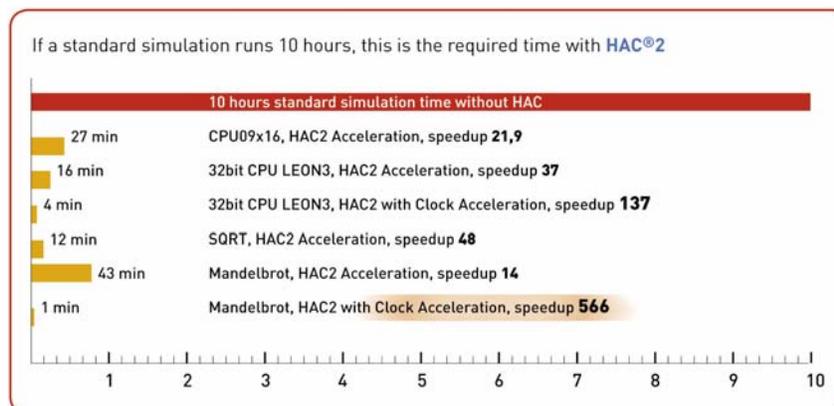
The SEmulator system allows co-simulation of real hardware blocks (e.g., Ethernet, display controllers) with an existing HDL simulation. This approach allows for the integration of real hardware earlier in the design cycle. Another benefit of using hardware in the loop is to co-simulate with hardware blocks, where no simulation model is available. It is possible to co-simulate a newly developed peripheral block with an existing CPU (without a simulation model) by putting the CPU on a child board and co-simulating it with the newly developed component.

Simulation Acceleration

Simulating huge designs with an HDL simulator is a tedious task. When the designer transfers parts of the design to the FPGA prototyping board and co-simulates them with the HDL simulation, the simulation run times can be decreased. Small designs will not benefit, because the communication overhead is higher than the gained simulation performance, but real world designs will speed up quite impressively.

Figure 4 shows the enormous impact and achievable reductions on the overall simulation time. The HAC2 technology is integrated into the SEmulator system, so similar results are achieved with the new system.

Figure 4. HAC2 Benchmarks



In the LEON3 example above, running the HDL simulation takes 10 hours. By placing this design into the FPGA and running it with clock accelerated simulation, the HDL test bench is still in the HDL simulator but this time it runs 37 times faster and only takes 16 minutes to complete. The clock acceleration technology allows the design to be run at full speed (up to 100 MHz) for selected time periods, and can be exploited for the LEON3 design to reduce the simulation time to 4 minutes, an acceleration of 137 times.

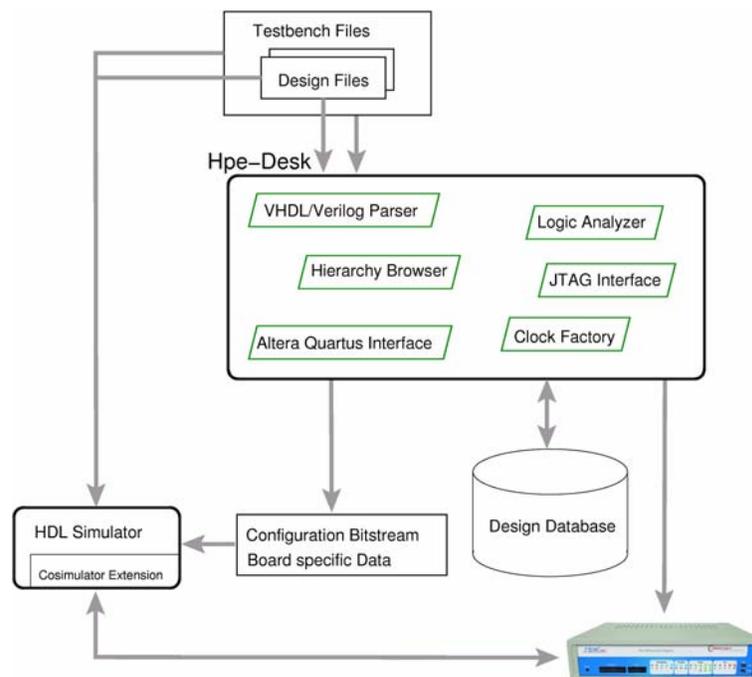
Faster simulations allow:

- Faster design iterations
- And/or the simulation of more test cases

The Hpe_desk Software

The Hpe_desk provides an intuitive and powerful scriptable graphical interface for the SEmulator functionality. Figure 5 shows the main software blocks (Hpe_desk and the HDL simulator) and the way the design files are processed.

Figure 5. Hpe_desk and FPGA Board



The Hpe_desk software consists of the following main components:

- VHDL/Verilog parser
- Hierarchy browser
- Altera Quartus® development software interface
- Clock factory
- Logic analyzer
- JTAG board test interface

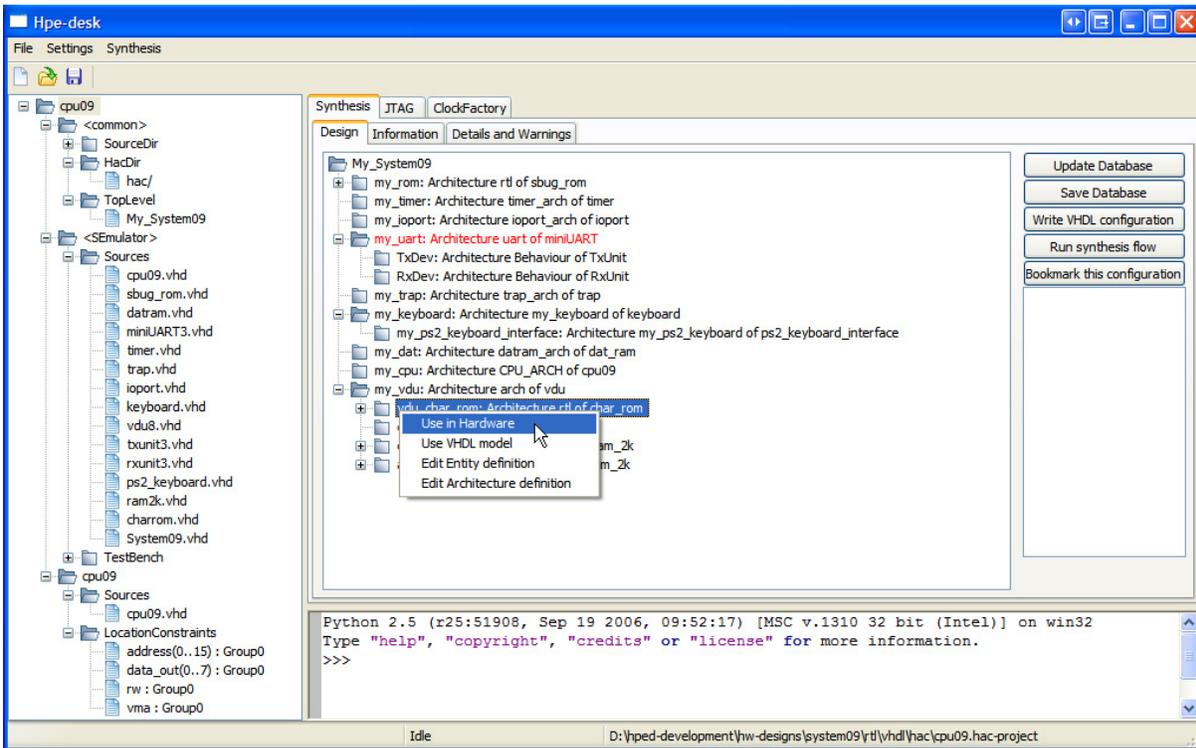
VHDL/Verilog Parser

The VHDL/Verilog parser takes the design descriptions and extracts the design hierarchy and other relevant data for SEmulation. The result is stored in the design database.

Hierarchy Browser

The hierarchy browser allows the designer to select design blocks that should be executed on the FPGA. The required information about the designs is stored in the design database. A screenshot is shown in [Figure 6](#).

Figure 6. Hierarchy Browser



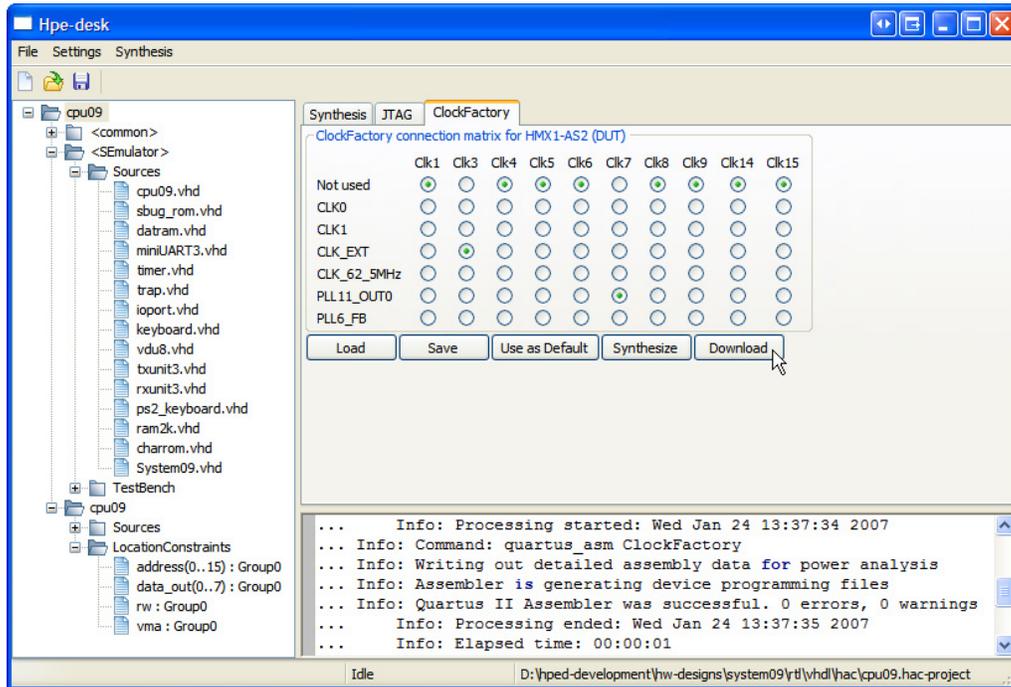
Altera Quartus Interface

The Altera Quartus incremental synthesis flow can be used to shift selected design blocks to the FPGA board quickly. Then, the generated configuration bitstreams can be downloaded to the FPGA prototyping board.

Clock Factory

The clock factory is an easy-to-use interface to select the required clocks for the Emulation mode, as shown in [Figure 7](#).

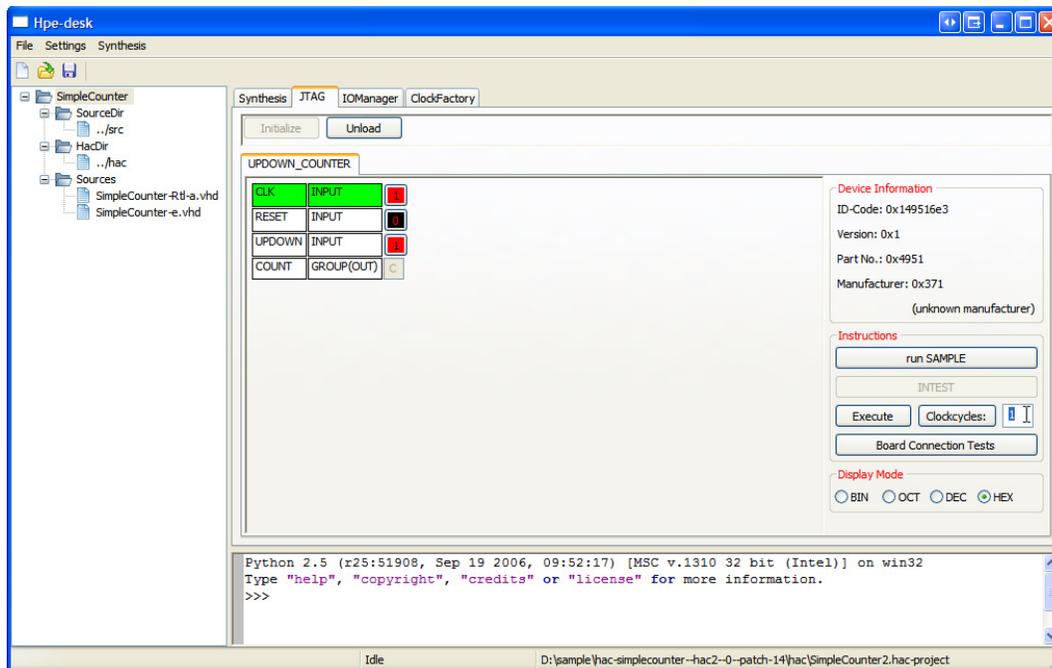
Figure 7. Clock Factory



JTAG Board Test Interface

With access to the JTAG-chain on the boards, the designer can use this standard protocol for debugging purposes. Figure 8 shows the JTAG GUI for a small test design. Adding a powerful connection tester allows it to be used for board tests.

Figure 8. JTAG Tool Window



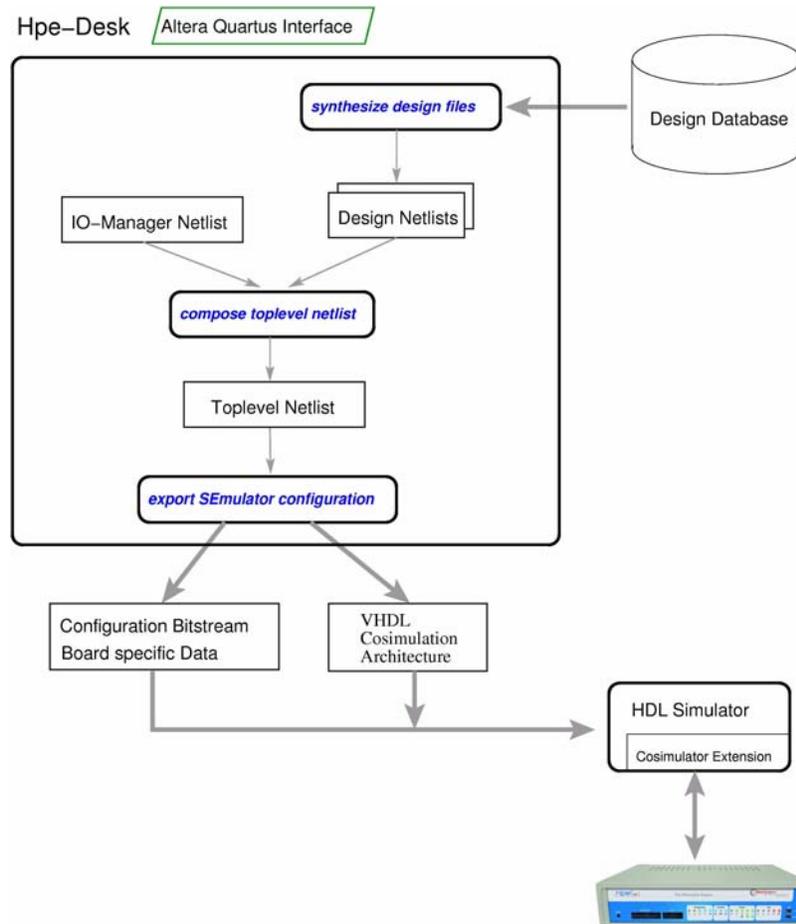
Logic Analyzer

A high-speed logic analyzer (available in 2008) is a useful debugging aid that will help to locate bugs without the need to use an external logic analyzer. The logic analyzer communicates via the PCIe X4 interface with the PC.

SEmulator Synthesis Flow Details

Once the designer selects the blocks to be placed on the FPGA prototyping board, the Altera Quartus interface synthesizes the blocks. The resulting design netlist and a predefined I/O manager netlist are combined to create a top-level netlist. The I/O manager is needed to observe and control the design, as well as provide a communication interface to the PC software. The Quartus interface exports the board specific bitstreams and the needed board-specific data for the co-simulator extension. This synthesis flow is shown in Figure 9.

Figure 9. SEmulator Synthesis Flow

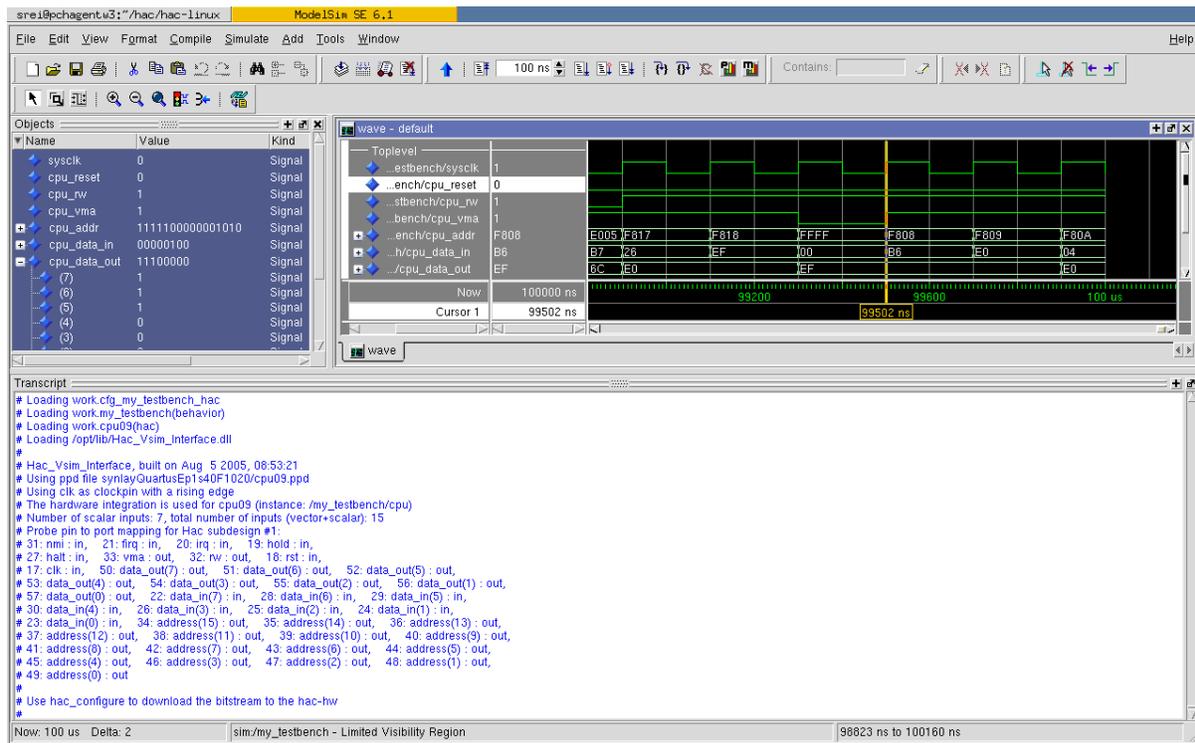


Normally, an entire design hierarchy is placed on an FPGA, as it is not possible to exclude one or more sub-blocks. The SEmulator design flow allows the exclusion of sub-blocks from the synthesis and keeps them as simulation models. This feature allows for very fast co-simulation of a design block, since the large hardware block is located on the FPGA and only small design blocks are simulated by the HDL simulator.

Running the Co-Simulation

Figure 10 shows a screenshot from a running Mentor Graphics® ModelSim® simulation connected to the FPGA prototyping board.

Figure 10. vsim-hac Snapshot



After the design block is synthesized and the simulator loads a test bench that instantiates this design block, the bitstream is automatically transferred to the DUT FPGAs. At this point the simulation can be started. All top-level signals of the synthesized block and some selected internal signals can be displayed in the standard waveform window.

Hardware for SEmulation

To combine simulation and emulation, the functionality of a hardware accelerator is necessary. In 1999, Gleichmann Research developed a hardware accelerator and co-simulator, based on a previous product, the HAC2. The HAC2 was an efficient and easy-to-handle product, but there was a major challenge with the data exchange, including protocol, speed, and the method of data acquisition. The HAC2 is a PCI card, expandable via a 10-Gbit proprietary interface. The handicap of such a system is the additional cost for hardware and the inherent difficulties of adapting such a system to an application.

As the hardware accelerator was developed, Gleichmann developed and produced FPGA rapid prototyping boards in parallel, using the advancements learnt from the HAC2. These Hpe® boards are already well accepted in the market, based on the strength of the numerous on-board interfaces and the ability to adapt to high-speed applications. Hundreds of these systems are already used in universities and labs worldwide.

The goal in 2006 was to combine these two systems. Customer discussions and requests revealed the following major targets:

- No additional hardware cost
- No different configurations between simulation and emulation
- Fast switch between simulation and emulation, with no additional synthesis of netlist
- Easy handling
- Additional debug tools on the same user interface

- Standard interfacing to PC
- Competitive pricing

The Hpe_midi can solve all of these requests.

Hpe_midi—The Most Flexible Rapid Prototyping System

The Hpe_midi (Figure 11) is the next step up from the Hpe_compact, an FPGA prototyping system, which has been available since 2002. The components consist of a motherboard, many user interfaces, and a removable FPGA module. Available today are 1- and 2-FPGA modules with the 4- and 8-FPGA modules currently under development.

Figure 11. Hpe_midi—Front View (left) and Open/Rear View (right)



All Hpe_midis are delivered in an enclosure. This protects the board against mechanical and environmental damage. But whenever necessary, the system can be used safely with the cover removed. This follows Gleichmann's philosophy of decreasing the number of hidden bugs, allowing the designer to concentrate on the real job of development.

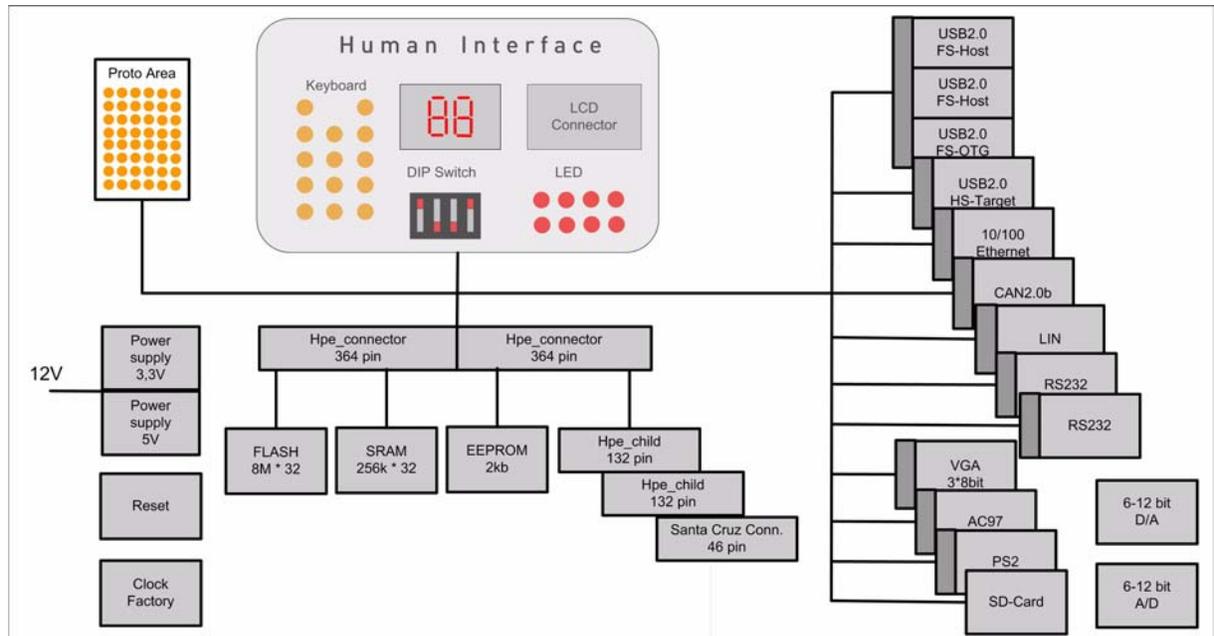
Functional Description

The Hpe_midi base board (shown in Figure 12) contains the following features:

- On-Board System Functions
 - Power supply 5V/11A and 3.3V/6A
 - Reset generation
 - Complex clock factory
 - Hpe_connector for FPGA module
 - 2 Hpe_child connectors for extensions
 - Altera Santa Cruz connector
 - PS/2 connector for mouse and keyboard
 - SD card slot
- On-Board Memory
 - FLASH 8M * 32
 - SRAM 256K * 32
 - 2-Kbyte EEPROM
- Human Interface
 - 3 * 4-button field
 - 2-digit 7-segment display
 - 8 LEDs
 - Reset button
 - Single-step key (debounced)
 - Connector for LCD display

- Others
 - Prototyping area
 - 2 quartz sockets (clock factory)
 - External clock input (clock factory)
 - D/A converter
 - A/D converter

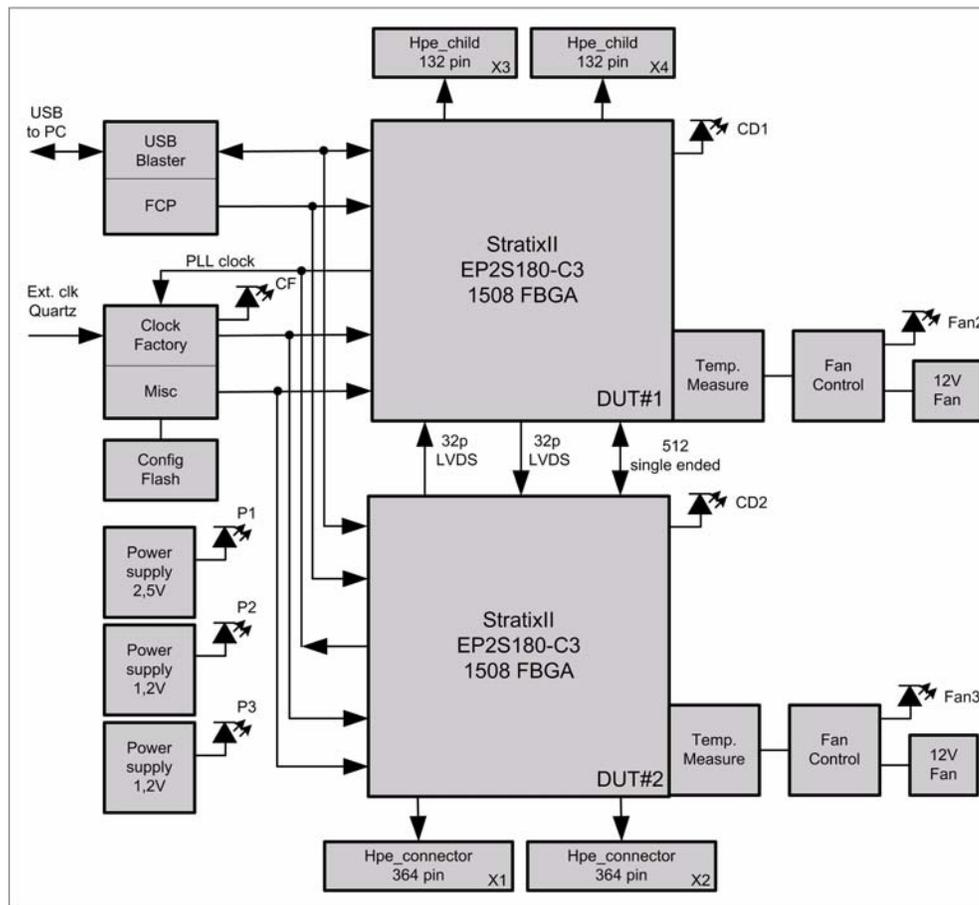
Figure 12. Hpe_midi Block Diagram



As an example, [Figure 13](#) shows a block diagram of the 2-FPGA module. Gleichmann will not accept limitations on maximum speed, so all systems provide the highest possible speed, giving customers all the functionality that the FPGA technology offers:

- All FPGAs are protectable with the Altera AES encryption.
- A programmable clock factory on the module provides more flexibility.
- A temperature-regulated fan for every FPGA guarantees highest security/reliability.
- Two Hpe_child boards on the module, plus two Hpe_child boards on the main board allow the adaptation of nearly every application.
- A high number of connections between the FPGAs guarantee easy synthesis.

Figure 13. Hpe_module 2X Block Diagram

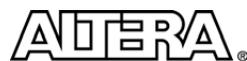


Conclusion

This paper has discussed how both simulation and design emulation can be combined into a single development environment to deliver shorter verification times and therefore shorter development times and reduced development costs. The SEmulator board is currently available at a very competitive price/performance ratio. The only additional component necessary is an Hpe_child board for PCIe external connectors, such as a PCIe X4 interface. This can connect up to four FPGAs to the PC by a card in the PCIe graphic slot (X16), and guarantees sufficient bandwidth for the required data throughput.

Acknowledgements

- Dieter Scheurer, Gleichmann Electronics Research
- Dr. Stefan Reichör, Gleichmann Electronics Research



101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
<http://www.altera.com>

Copyright © 2007 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.