

A high-level alternative to writing RTL, Bluespec is reinventing hardware design for ASICs and FPGAs with the only ESL synthesis solution that addresses algorithms, control logic and complex datapaths. Bluespec's ESL synthesis solution is fully synthesizable without compromise in area, speed or latency. The Bluespec toolset allows ASIC and FPGA designers to appreciably reduce design time, bugs, verification resources and re-spins that contribute to product delays and escalating costs. Product highlights:

- Significantly raises the level of abstraction for faster time to completed designs
- Seamless integration with current tools and methodologies, including support for multiple clock domains and verification assertions
- HW-centric design language based on industry standard SystemVerilog and Verilog, familiar to engineers
- Maintains the designer's intent – Bluespec makes the designer faster and more accurate by delivering a better design. Bluespec does not try to think for the designer by inventing architecture.
- Synthesizes control logic for correct-by-compiler construction, while retaining 100% of the designer's structure and intent
- Generates no compromise Verilog HW & cycle-accurate models

Tremendous attention has been placed on the mounting cost and schedule issues of verification and timing closure. This focus has overlooked the root cause problem – spiraling design complexities. Verilog and VHDL have changed little in over fifteen years. Designers are stuck trying to construct multi-million gate designs at too low a level, with outdated tools. Low level design composition is contributing to the downstream verification problem by causing more bugs and making the timing closure problem more acute by limiting designer options.

Bluespec attacks the source of the problem, making the ASIC or FPGA designer's job faster and more accurate. By accelerating and improving the quality of designs, Bluespec greatly improves the verification, both with a faster schedule and fewer resources. And by designing in an environment where the micro-architecture can be safely changed, Bluespec designers can confidently effect timing closure instead of trying to close timing sub-optimally, when making Verilog or VHDL changes would be avoided.

Benefits to the Engineering Team

Bluespec materially impacts designs:

- Accelerate time to verified design by 50%
- Reduce both bugs and verification costs by 50%
- Retain flexibility to make architectural changes late in the design cycle
- Enable rapid timing closure
- Experience unparalleled reuse, including faster derivatives
- Leverage a unified environment for transaction level modeling through to hardware generation
- Deliver safe, low-impact ECOs that do not disrupt designs

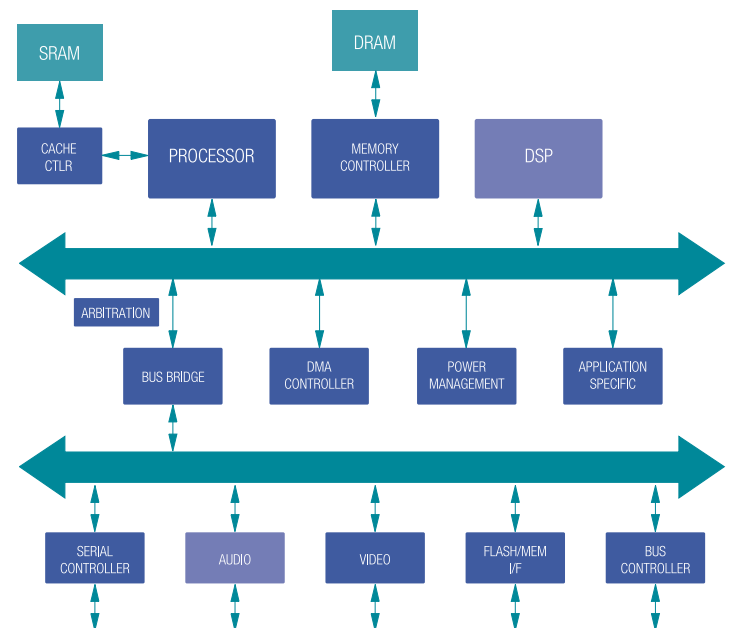
Raising the Level of Abstraction

There have been multiple attempts at attacking design complexity, the root cause behind many ASIC and FPGA development issues. While all have raised the level of abstraction in some way, they have been unable to deliver both modeling abstraction and hardware generation rivaling RTL design.

While a seductive concept, the reality has inevitably been inferior hardware, in area and speed. As attractive the notion that computers can determine optimal architectural and implementation approaches from an abstract algorithm, the problems are far too complex and the solution spaces too expansive.

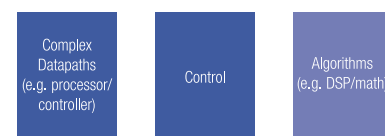
The second approach has been with C/C++/SystemC. As C provides a higher layer of abstraction and is already used as a modeling language by some, it has been tempting to exploit it for hardware implementation as well. Fundamentally, this requires technology that can translate sequential software semantics into parallel, state machine-based hardware. But, there is a problem – this capability has proven to have very limited success, in several decades of research in parallelizing compilers. Despite tremendous investments, the semantic chasm between sequential software models and parallel hardware models has proven to be too wide.

Bluespec™ ESL Synthesis



DESIGNS COMPLETED WITH BLUESPEC

RISC processor	SRAM Ctlr	Queuing Engines	I2C	IDCT
MIPS	Bus Converters	Sorting Queue	Pixel Processor	Motion Compensator
IA-64	AMBA DMA Ctlr	Arbiter	Waveform Generator	MPEG
Power processor	Network Processors	IP Lookup	Pong	DES
L2 Cache Ctlr	UTMI (PHY)	Debug Controller		FIR Filter



Bluespec is the only ESL synthesis solution that also addresses control and complex datapaths.

Bluespec Approach

In contrast to previous approaches, Bluespec raises the level of abstraction without compromising the quality of hardware generated. Using familiar hardware semantics, Bluespec builds on RTL by leveraging the same hardware intuitions, providing the designer transparency, predictability and control over the architecture and micro-architecture, and integrating with the same Verilog and VHDL toolsets and flows already in place. On this strong, hardware centric foundation, Bluespec adds a powerful new way to express concurrency and inter-module communications, not available in any other Hardware Design Language (HDL) or modeling language:

- High-level description of behavior (Rules and Interface Methods) resulting in designs that are more succinct, more correct-by-construction, and easier to verify
- Very powerful interface semantics, which enhance correctness when an IP block is plugged into varying environments and automatically manage resource sharing
- Very high degree of parameterization, which greatly improves reuse
- Strong support for embedded SystemVerilog Assertions (SVA), multiple clock domains and gated clocks

Bluespec has been successfully used in designs as diverse as (partial list):

- Controllers: cache/memory/DMA/serial/peripheral
- Bus Converters/Arbiters
- Network Processors
- PHYs
- Queuing/Sorting Engines
- Processors, RISC or CISC
- Pixel Processors
- MPEG4

Designers have the option of writing their designs at different levels, from transaction-level to implementation-targeted hardware. When starting intentionally high, designers can perform, at their control, a series of successive refinement steps on the design. Different parts of the design can be at different levels and progressively they become closer to the hardware – all in the same environment and all with the same design. For the first time, Bluespec offers a unified design environment where designs can be architected, modeled, rapidly prototyped, and hardware generated that competes with hand-crafted designs.

- Verilog, for use with current Verilog synthesis and accelerated simulation tools (or)
- Cycle accurate execution models, for accelerated simulation

Additionally, Bluespec is completely interoperable with Verilog based designs. Bluespec can easily incorporate Verilog IP – alternatively, Bluespec can be used within Verilog implementations.

Bluespec sits directly in front of current design flows. The designer specifies the design using Bluespec SystemVerilog and generates Verilog output or cycle accurate execution models. Bluespec saves time in your total design approach.

Tool Capabilities

As the first effective ESL synthesis compiler and simulator, Bluespec's toolset provides a unique mix of capabilities. The Bluespec toolset includes both the Compiler and Simulator.

Bluespec Compiler (BSC)

Key features:

- Bluespec SystemVerilog with rules and methods
- Correct-by-compiler construction of control and datapath logic
- Generation of no compromise Verilog RTL
- Comprehensive static verification of designs to eliminate problems before simulation
- Code succinctness and static elaboration for high-level abstractness and code reuse
- Integrates into and with existing Verilog/VHDL/SystemC IP
- Rich library of design building blocks
- Integrated compiler algorithms and techniques:
 - Automated and user-defined scheduling of hardware
 - Scheduling visualization and feedback
 - Resource assignment, optimization
 - Standard optimizations, including common sub-expression elimination and logic

Bluespec Simulator (Bluesim)

Key features:

- Simulation of the high-level design
- 100% cycle accurate with Verilog RTL
- Full visibility to Bluespec interfaces, state elements and design assertion rules
- Generates standard VCD files

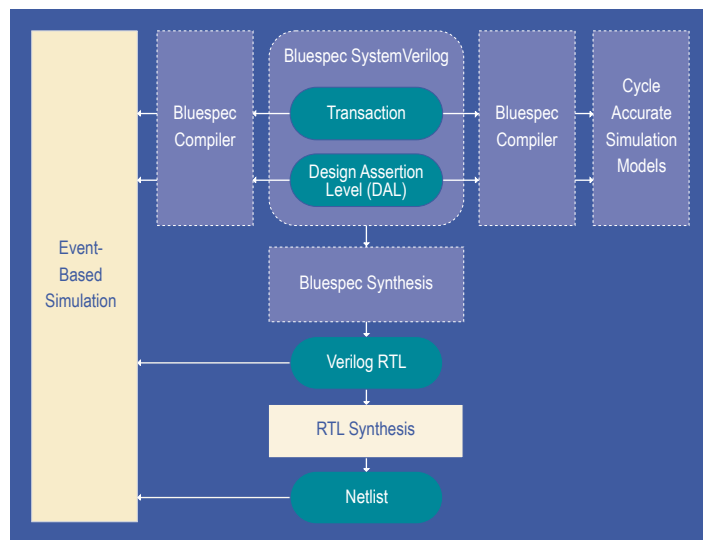
Both Bluesim and BSC were designed for and tested with Linux Red Hat 7.2 and 8.0.

Term Rewriting Systems

Bluespec's patented technology is based on over seven years of research at MIT. At the core of Bluespec's compiler technology is the introduction of the application of Term Rewriting Systems (TRS) to hardware synthesis. Term Rewriting Systems are a well-understood formalism from computer science. A TRS consists of "terms" which describe hardware states, and "rules" which describe behavior. A "rule" captures both a state change (an "action") and the conditions under which it can occur. As the rules in a Term Rewriting System have atomic semantics, analysis of hardware can be done even though it may be highly concurrent and complex. Research at MIT has shown how this can greatly assure formal correctness, for example in proving that an RTL description properly implements a high-level specification. Research has also shown how to compile a TRS into highly efficient RTL code.

Further information can be read on www.bluespec.com about additional technical solution areas such as verification time reduction and Term Rewriting Systems.

Bluespec's three day training course gives designers the tools they need to be successful and show productivity improvement on their first project. Please contact your sales representative to sign up.



Design Flow

As illustrated by the design flow diagram above, Bluespec is designed to fit into current, Verilog-based tools without interrupting your current flow. Bluespec produces either: