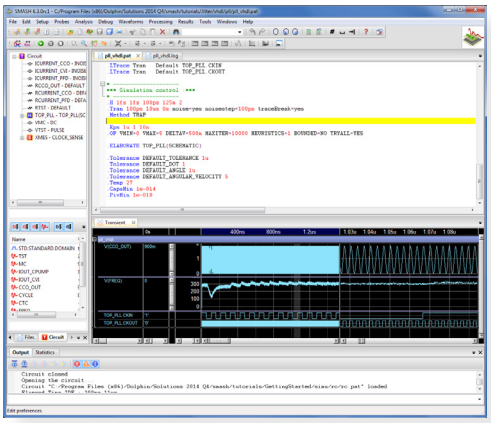
SMASH 6.3

SMASH NEW GENERATION

Most of our users are aware that Dolphin Integration has been preparing major changes in SMASH to increase our users' productivity with:

- A new IDE sporting a new waveform viewer
- An upgraded implementation of the already best-in-class VHDL and VHDL-AMS language compliance and simulation engine

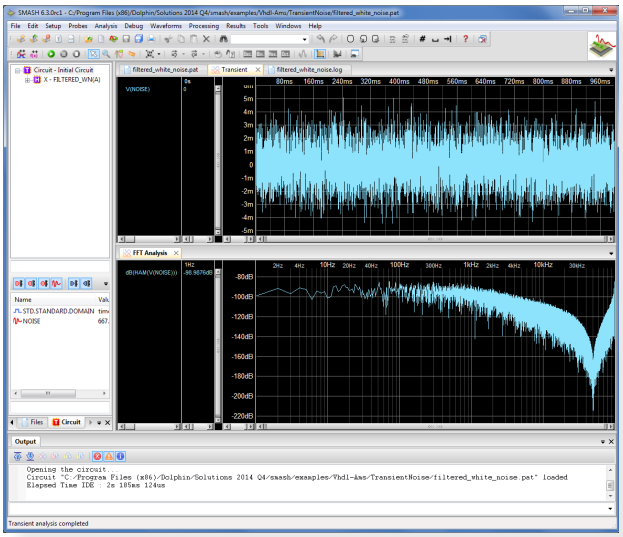
SMASH 6.3 cumulates these enhancements to significantly improve user experience.



Overview of SMASH mixed signal simulator

KEY FEATURES OF SMASH 6.3

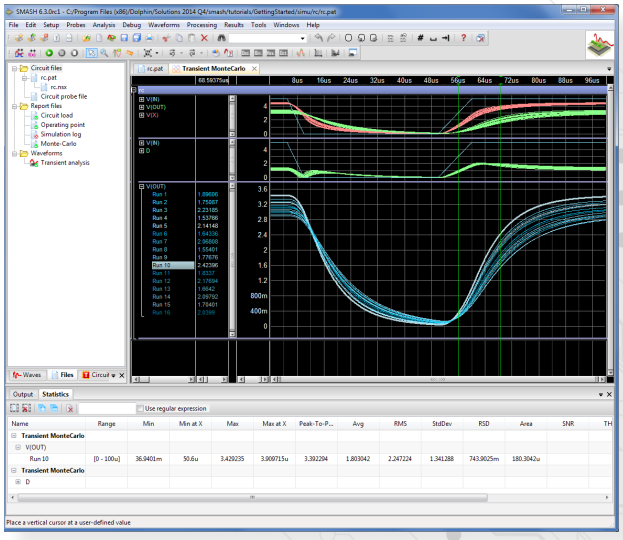
- New IDE with enhanced user interface and waveform viewer
- Upgraded VHDL and VHDL-AMS implementation
  - Substantial extension of language compliance
  - Improved loading and simulation time on some designs
  - Finer debug when using breakpoints and step-by-step execution
- Integrated standard UTSOI compact model for predictive analysis of the FDSOI process
- Extended support of transient noise analysis to all noise sources
- Automatic detection of analysis runs in batch mode
- Enhanced HSPICE compatibility
  - Support of Safe Operating Area (SOA) checks specified in foundry model parameter sets



Transient noise analysis on HDL-AMS description

HIGHLIGHT ON THE MAJOR ENHANCEMENTS OF THE USER INTERFACE AND THE WAVEFORM VIEWER

- Smooth navigation between simulator control file, netlist / design files, and waveforms... using tabs
- Smarter management of panels
- Capability to resize, move, add or delete columns
- Creation of a hierarchy of waveform groups
- Capability to expand busses and multiple runs of Monte Carlo and Sweep analyses
- Statistics pane to display statistics of analog signals at a glance
- Use of several vertical cursors to perform measurements



Overview of SMASH IDE and waveform viewer

DOLPHIN INTEGRATION
The Enabler of mixed-signal Systems-on-Chip

SMASH 6.3

VHDL AND VHDL-AMS NEW TECHNOLOGY

Our renowned state-of-art VHDL-AMS simulation engine has undergone a significant upgrade to provide:

- Enhanced language compliance,
- Improved loading and simulation time,
- Increased behavioral modeling and simulation capabilities with Transient Noise Analysis (TNA).

It paves the way to further speed improvements as well as to the support of VHDL 2008 and SystemVerilog.

```
1
2   library ieee;
3   use ieee.electrical_systems.all;
4   use ieee.math_real.all;
5
6   entity e is
7       generic (f: real := 1.0e3);
8   end entity;
9
10  architecture a of e is
11      quantity q1, q2: voltage;
12  begin
13
14      procedural is
15          variable var: real;
16      begin
17          var := sin(math_2_pi * f * now);
18          q1  := var;
19          q2  := cos(3.0 * var);
20      end procedural;
21
22  end architecture;
23
```

Example of procedural statement

Your feedback matters

SMASH is renowned among the VHDL-AMS community as the simulator with the most advanced language compliance. SMASH 6.3 pushes the edge further and extend its VHDL-AMS support.

Please provide feedback on your user experience with your test cases to solutions@dolphin-integration.com.

- New features
  - Expressions as input port actuals in component instantiations
  - Subprogram declarations and definitions in more contexts (entities, architectures, processes)
  - Type declarations in more contexts (entities, processes, generate blocks)
  - Full support of use clauses
  - References to definitions in other packages with the full name to the definition, with no use clause
  - Reference to VHDL signals in Verilog with hierarchical names
  - VHDL-AMS
    - Simultaneous procedural statements
    - Break statements on a single elements of composite quantities
    - Attributes on composite, indexed or selected quantities
    - Support of small signal noise analysis

- Enhancements
  - Large improvement of the support of aliases
  - Extended support of record types for ports and generics
  - Local Truncation Error (LTE) algorithm applied to VHDL-AMS simulations
  - Improved coverage reports
  - More detailed debugger breakpoints and step-by-step execution
  - Support of large arrays (memories), record signals and complex expressions which previously slowed down or blocked circuit loading
  - Exit and next statements with loop labels to apply to the designated loop
  - Improved mixity between VHDL-AMS and Verilog-A or Spice descriptions
  - Up to 2 times faster when running transient analysis

```
4   library ieee;
5   use ieee.std_logic_1164.all;
6   use work.defs.word_size;
7   architecture arch of tb is
8   begin
9     process is
10      type tstep is (ST0, ST1, ST2, STF);
11      variable step     : tstep;
12      variable errcount : integer;
13
14      procedure init is
15      begin
16        work.logger.init;
17        work.logger.log("Beginning tests");
18        step := ST0;
19        errcount := 0;
20      end procedure;
21
22      subtype  word is std_logic_vector(word_size-1 downto 0);
23      variable data : word;
```

Illustration of extended support of VHDL