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

SMASH 6.2

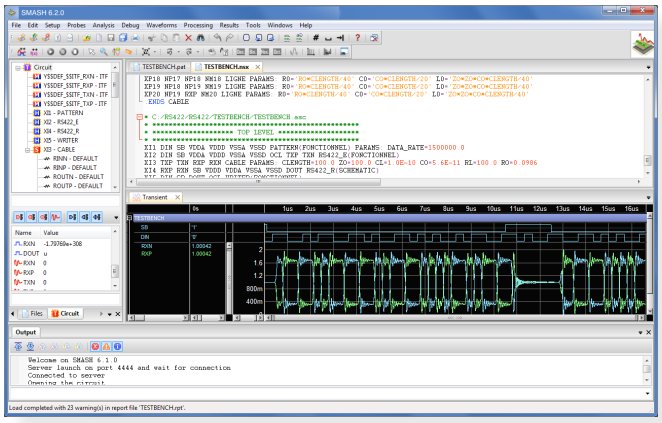
EDA Solutions

Back to the Future

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
- A new implementation of the VHDL and VHDL-AMS standards

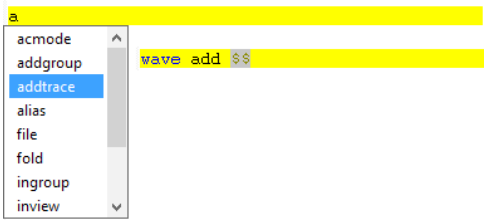
SMASH 6.2 is the opportunity to test these improvements in the field in your own environment and share your feedback and needs for the future.



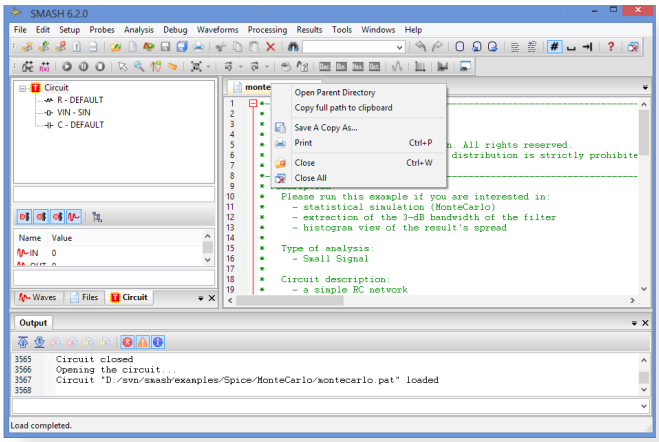
Overview of SMASH mixed-signal simulator

Key Features of SMASH 6.2

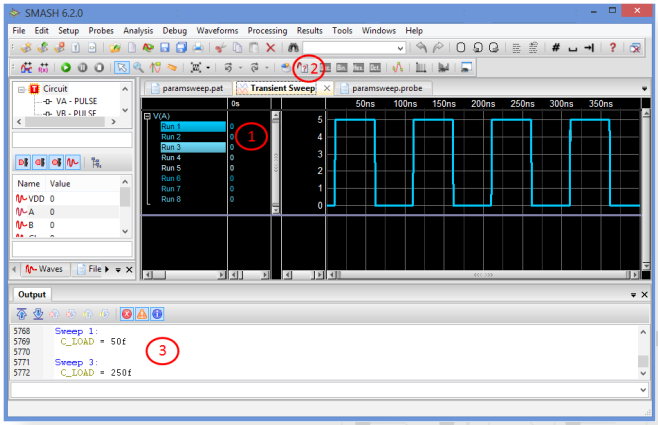
- New IDE with enhanced user interface and waveform viewer
  - Up to 10 times faster display for multiple runs
  - Contextual menu on tabs for a faster access to source files in the workstation directory hierarchy
  - Dialog to export waves to text format
  - Multiple selection of Monte Carlo and Sweep runs to get information
  - Scriptable capabilities to add traces with .probe file
    - Independent from circuit loading
    - Tcl-like
    - Capability to create and save group of signals
    - Auto-completion
- Reduced memory consumption (-30%) and loading time (-20%) for SPICE circuits using nanometric processes
- Increased accuracy for CCS power characterization
- Dedicated API for creating parasitic capacitors on the nets of the circuit
  - Anticipate the impact of parasitic capacitors on performances prior to layout availability
- Technology Preview of new VHDL and VHDL-AMS implementation
  - Substantial extension of language compliance
  - Improved loading and simulation time on specific designs



Auto-complemention feature to add traces



New contextual menu on tab in new IDE



Multiple selection of several runs with new IDE

SMASH and ICLys™

ICLys™ from Infiniscale can now process simulation results from SMASH for process variability analysis and reliability qualifications.

ICLys™ coupled with SMASH enables to improve not only the quality but also the speed of the yield verification.

Impressive simulation time speed-up for local variability analysis

VHDL AND VHDL-AMS TECHNOLOGY PREVIEW

Going further than our renowned top-notch VHDL-AMS simulation engine, the future VHDL and VHDL-AMS engine of SMASH is delivered as a Technology Preview with:

- Enhanced language compliance,
- Improved loading and simulation time.

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

Your feedback matters

The VHDL and VHDL-AMS Technology Preview is delivered in the standard release. It can be activated easily in the application Preferences.

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

```
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

- New features
  - VHDL-AMS
    - Simultaneous procedural statements
    - Break statements on a single elements of composite quantities
    - Attributes on composite, indexed or selected quantities
  - 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

- 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
- Corrections
  - Sensitivity on indexed-ids, selected-ids and slices
  - 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
  - Value of the VHDL-AMS 'ABOVE attribute at operating-point
  - Support of composite terminals and composite quantities

```
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