System Verilog C/C++ Interface

Swapnajit Mittra
Chair, System Verilog C/C++
Interface Committee
SGI

System Verilog C/C++ Interface

- Objectives
  - To provide specialized APIs for specialized needs.
  - Three main areas of focus
    - DirectC API
    - Assertion API
    - Coverage Metrics API
  - Working with other groups within System Verilog team to reach a common goal.

System Verilog C/C++ Interface

- DirectC API
  - Not a replacement for Verilog PLI.
  - Users need a simple to use, high-performance interface to incorporate programming language (C/C++) models into their designs.
  - User supplies the C/C++ source code or object code/library to the simulator/compiler.
  - User declares external functions in System Verilog.

System Verilog C/C++ Interface

- DirectC Example

```
module test;
  initial
     hello_directc();
endmodule

extern void hello_directc();

void hello_directc() {
   printf("Hello All!\n");
}
```

System Verilog C/C++ Interface

- Assertion API
  - Tools and test benches require access to assertion information
    - Reactive test benches
    - Debug/analysis tools
  - Static assertion information
    - Assertion iterators
    - Type information
    - Source information
  - Dynamic assertion information
    - Callbacks
    - Assertion controls

System Verilog C/C++ Interface

- Coverage Metrics API
  - Realtime coverage information needed
    - By reactive testbenches
    - To control random stimulus generators
  - Provides a common means to access coverage information regardless of tool or coverage metric.
  - Independent of the simulator implementation.

System Verilog C/C++ Interface

- Draft version of LRM

http://www.eda.org/sv-cc/

- Availability of the final LRM
  - Target is System Verilog 4.0
  - Planned release date is around DAC 2003 timeframe.