

Proposal to Exclude Verilog and VHDL Declarations from the IEEE1647 Standard

{eran,vitaly}@verisity.com
1647-fitf@ieee1647.org

1 Background

Chapter 25 in the draft describes constructs to support the interface between Verilog and VHDL simulators and the *e* language implementation. This interface is based on the tick-access notation, and contains several declarations, used to support tick-access notation and integration with simulators.

These declarations are not general. They usually relate to a specific construct in either Verilog or VHDL (for example, the **vhdl code** declaration). Some of them exist to get over Verilog and VHDL limitations (for example, the **verilog variable** declaration). Sometimes they can be replaced by user-defined macros, as part of a vendor-specific integration.

The *e* language supports another mechanism for integration with third-party simulators, the *e* Ports (described in Chapter 6). *e* ports achieve a similar functionality by using port attributes. This way is more consistent in a linguistic sense. It is also easily extensible by third party integrators.

2 The Proposal

The aim of the IEEE1647 standard is to describe the *e* language itself. Keeping the Verilog and VHDL declarations in the standard subset of *e*, prevents it from being neutral and hence suitable for any HDL. Based on this, we propose the following:

- Any Verilog or VHDL declarations will be removed from this Chapter. In other words, sections 25.1 and 25.2 would be removed. A full listing of the declarations is at the end of the document.
- Chapter 25 will describe:
 - **force** (currently 25.3.1)
 - **release** (currently 25.3.2)
 - **'HDL-pathname'** — tick-access notation (currently 25.4.1)

- **simulator_command** (currently 25.5.1)
- **stop_run** (currently 25.5.2)
- The '**HDL-pathname**' construct — the tick access itself — will be extended to describe both access to HDL objects and HDL subprogram calls.
- To enable better interoperability:
 - Vendors can add macros.
 - Vendors can define additional unit and port attributes.

Table 1 Proposed Declarations for Exclusion from IEEE1647

Declaration	Section	Page
verilog code	25.1.1	803
verilog function	25.1.2	805
verilog import	25.1.3	807
verilog task	25.1.4	809
verilog time	25.1.5	811
verilog variable reg wire	25.1.6	812
verilog variable memory	25.1.7	818
vhdl code	25.2.1	821
vhdl driver	25.2.2	823
vhdl function	25.2.3	827
vhdl procedure	25.2.4	830
vhdl time	25.2.5	837

All sections and pages refer to the 0.2 Draft

Note Section 25.4.2 "**specman deferred**" should be excluded from the standard as well (as mentioned in Issue 101).