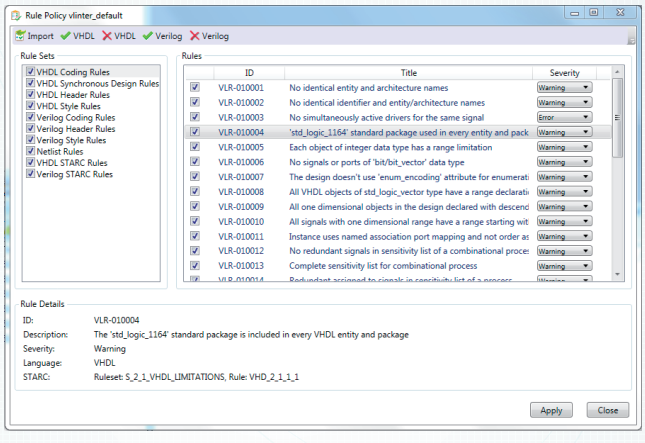
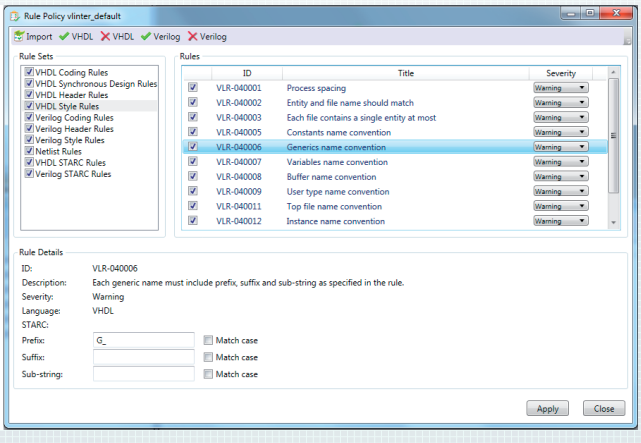
Rule Policy settings

The user can activate / deactivate the rules out of the rule list, creating a user-defined rule policy. The rule policy is stored in a special rule policy file, which can be then exported / imported to / from other design projects. The rule policy file name is noted in all vLinter reports. vLinter rules having matching STARC rules, include a special reference to STARC rule set and rule number.



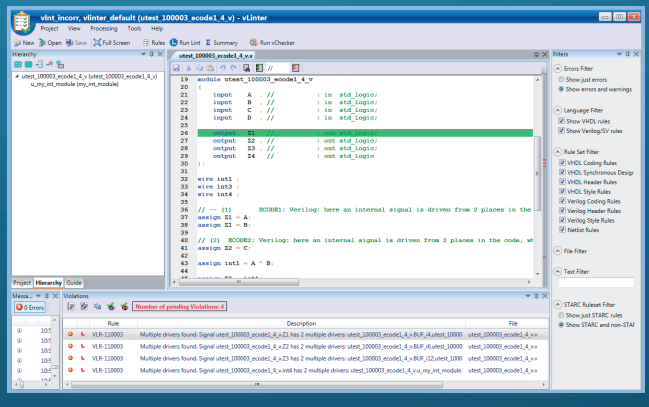
Rule Customization

Some vLinter rules can be customized, e.g. name conventions prefix/suffix/sub-string can be customized. In addition, rule severity can be also customized: the user can pick up from warning and error options. The customized rules become a part of the user defined rule policy.



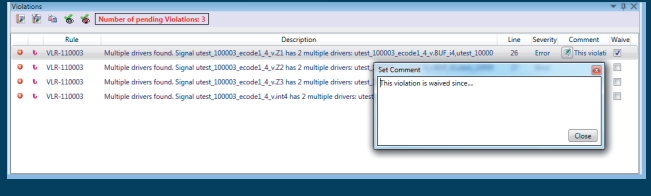
Violation Analysis

vLinter reports violations in a special window. The violations can be filtered and grouped in different ways inside the window. For each violation, vLinter provides a cross-probe to RTL code, allowing efficient debugging.



Violation Management

The user can decide to disregard some reported violations. This decision can be managed through Waive and Comment vLinter interface. The user marks the violation as waived and writes a comment describing the reason for this decision. The waived violations can be filtered out from the report and appear as a special section in a final HTML report of the project.



Violation Report

vLinter support both textual and HTML reports generation. The HTML report allows an efficient review of the analysis results and can be generated in both full and a filtered / reduced (focus on certain issues) form.

www.vsyncc.com
info@vsyncc.com
+972-54-4248169

VSYNC circuits

vLinter

Early design analysis and verification
Rule-based design management

vLinter is a static analysis-based verification tool employed in early design stages for hunting design bugs related to a bad coding practices. Some examples of such bugs are: unsynthesizable code, unintentional latches, undriven signals, race conditions, out of range indexing, incomplete case statements, mismatch between simulation and synthesis, etc.. vLinter contains multiple rule sets that are employed to construct a customized rule policy for a project or a design group. A RTL design is analyzed according to the rule policy, and all the breached rules are reported as violations for an user review. vLinter allows an efficient violation report management through different violation report views. At the last stage, vLinter supports generation of comprehensive analysis reports, required for further project design reviews, qualifications and certifications.

vLinter can catch bugs, requiring no specific test vectors and it is very effective when employed pre-simulation, reducing the number of simulation cycles needed to cover a functionality of a RTL module. The recommended design flow suggests employing vLinter to compile a fresh code, before any functional verification.

vLinter supports both ASIC and FPGA design flows, allows easy and fast setup by directly loading in project files of leading ASIC and FPGA vendor synthesis tools. vLinter is compatible with vSync Vincent CDC platform, allowing an immediate switch to CDC (Clock Domain Crossing) analysis after reviewing linter violations.

vLinter supports all VHDL and Verilog/System Verilog standards, including mixed-language design. In addition, the tool handles efficiently partial design, having internal black boxes. vLinter supports both Windows and Linux platforms.

Main Features

- **Multiple pre-defined rule sets**
  - Including STARC, RMM, etc.
- **An efficient rule policy management**
  - Per project
  - Per design group
  - Rule customization by user
  - Rule import and export
  - Rule severity settings
- **An automatic setup using main FPGA/ASIC vendor project files**
- **Support of all VHDL and Verilog/SV standards**
- **Support of Windows and Linux platforms**
- **Support of ASIC and FPGA flows**
- **Efficient violations management**
  - "Waive and Comment" ability at violation and a rule levels
  - Grouping messages by type
  - Different filtering abilities
- **Black-box support**
- **Unified data base with vSync Vincent CDC platform**
- **HTML report**
  - Rule policy employed
  - Black-box list
  - Violations list
  - Waivers and comments list
- **Qualification data-base**
  - Ready for DO-254 tool qualification
- **Examples of reported violations**
  - Non-synthesizable code
  - Missing signals in sensitivity list
  - Combinational loops
  - Unintentional latches
  - Array range mismatches
  - Case statement style issues
  - Contending entities/modules
  - Clock issues
  - Reset issues
  - Signal / bus contention
  - Unused signals / Dead code
  - Missing / partial header issues
  - Missing hooks for DO-254
  - Divide by zero
  - Violations of name convention
  - Black-boxes

www.vsyncc.com
info@vsyncc.com
+972-54-4248169