SystemVerilog Tools in Python

Michał Czyż
PG ETI UE CHIP

The Other Dude
His Company / University

September 11, 2018

Contents

Abstract

Documentation for SystemVerilog Tools project available at:

`https:www.github.com/emcezet/sv-tools`

1 Introduction

The goal of project is to develop parsing engine for a subsystem of SystemVerilog (SV) Hardware Description Language (HDL) and provide following utility:

- module generation from user-defined templates,
- module instantiation from user file,
- top level file generation from user files,
- simple testbench file generation.

Both named port and interface style are accepted.

2 Acceptable constructs in SystemVerilog.

2.1 Formal syntax.

Formal syntax descriptions use Backus-Naur Form (BNF). This notation was also used in Annex A of [1]. Revised and limit list of supported constructs is provided in this section. Please note, that a specific coding style is expected from users. Syntax notation clarification.

⟨*symbol*⟩ ::= __expression__

1. Every rule consists of symbols. Right hand-side of rule is called an expression.
2. Curly brackets { and } denote one or more occurence of symbol.
3. Square brackets [ ] denote optional symbol.
4. Keywords are denoted without any special marking.
5. All other symbols are wrapped in < >.
6. Grammar specific delimiters are wrapped in ' '.
7.

⟨*source_text*⟩ **::=** [⟨*timeunits_declaration*⟩] {⟨*description*⟩}

⟨*description*⟩ **::=** ⟨*module_declaration*⟩
| ⟨*interface_declaration*⟩

⟨*module_declaration*⟩ **::=** ⟨*module_nonansi_header*⟩ [⟨*timeunits_declaration*⟩] { ⟨*module_item*⟩ } endmodule [: ⟨*module_identifier*⟩]
| ⟨*module_ansi_header*⟩ [⟨*timeunits_declaration*⟩] { ⟨*non_port_module_item*⟩ } endmodule [: module_identifier]
| extern ⟨*interface_nonansi_header*⟩
| extern ⟨*interface_ansi_header*⟩

⟨*module_nonansi_header*⟩ **::=** {⟨*attribute_instance*⟩} ⟨*module_keyword*⟩ [⟨*lifetime*⟩] ⟨*module_identifier*⟩ {⟨*package_import_declaration*⟩} [⟨*parameter_port_list*⟩] ⟨*list_of_ports*⟩ ';'

⟨*module_ansi_header*⟩ **::=** {⟨*attribute_instance*⟩} ⟨*module_keyword*⟩ [⟨*lifetime*⟩] ⟨*module_identifier*⟩ {⟨*package_import_declaration*⟩} [⟨*parameter_port_list*⟩] [⟨*list_of_port_declarations*⟩] ;

⟨*interface_declaration*⟩ **::=** ⟨*interface_nonansi_header*⟩ [⟨*timeunits_declaration*⟩] {⟨*interface_item*⟩} ⟨*endinterface*⟩ [':' ⟨*interface_identifier*⟩]
| ⟨*interface_ansi_header*⟩ [⟨*timeunits_declaration*⟩] {⟨*non_port_interface_item*⟩} ⟨*endinterface*⟩ [':' ⟨*interface_identifier*⟩]
| {⟨*attribute_instance*⟩} interface ⟨*interface_identifier*⟩ (.*) ; [⟨*timeunits_declaration*⟩] {⟨*interface_item*⟩} endinterface [':' ⟨*interface_identifier*⟩]
| extern ⟨*interface_nonansi_header*⟩
| extern ⟨*interface_ansi_header*⟩

⟨*interface_nonansi_header*⟩ **::=** {⟨*attribute_instance*⟩} interface [⟨*lifetime*⟩] ⟨*interface_identifier*⟩ {⟨*package_import_declaration*⟩} [⟨*parameter_port_list*⟩] ⟨*list_of_ports*⟩ ';'

⟨*interface_ansi_header*⟩ **::=** {⟨*attribute_instance*⟩} interface [⟨*lifetime*⟩] ⟨*interface_identifier*⟩ {⟨*package_import_declaration*⟩} [⟨*parameter_port_list*⟩] [⟨*list_of_port_declarations*⟩] ';'

3 Usage.

All scripts support -h and –help parameters for usage help.

References

[1] 8299595,IEEE Std 1800-2017 (Revision of IEEE Std 1800-2012),IEEE Standard for SystemVerilog–Unified Hardware Design, Specification, and Verification Language, 2018,Feb

[2] placeholder