PROPOSAL - Mantis #680 - 5/8/2005 by Cliff Cummings

Observations:

- We do not have complete agreement on the discussed configurations proposal
- There are already at least two implementations that recognize the keywords: config, endconfig, cell
- The two implementations have minor use-model differences that will have no consequence to 99%+ of all Verilog designs
- Most engineers will separate library files from config files from design files, and the divergent use-models are actually very similar when engineers make the separations
- config, endconfig and cell are already passed keywords in the Verilog-2001 Standard
- We do not want to set a precedent of reviewing keywords after they have become part of an IEEE Standard and after they have been implemented by one or more vendors
- The current proposed changes are too significant to be added to a re-circulation ballot of the standard and do not promote consensus.
- The current proposed changes will change at least one yes-vote to a no-vote on both the 1364 and 1800 re-circulation ballots. The no-votes will be related to changes in the standard and will have to be addressed in a third re-circulation ballot.
- There is already at least one book and multiple copies of training materials that show the use of the config-endconfig implementation of configurations. All of these published materials become obsolete if the keywords change.
- There are other config-issues that should be addressed in a more-encompassing proposal, including the issues of wildcard directories and file-naming to wildcard matching restrictions (both hinted at by Steve Sharp).

Proposal options (I vote yes to either proposal under the conditions noted below).

(1) Do nothing (not even BNF or typo fixes) and respond to the negative ballot with the template form: "Although the proposed changes have merit, they are beyond the scope (etc., etc.) ..."

This option is the safest because no new negative votes can be registered against the 1364 and 1800 standards based on any configuration issues.

(2) Make the minor BNF changes and typo fixes shown below and still respond with the above template form.

I am willing to support this option only if a straw-poll show that there will be no new configuration section objections raised by the changes. If Cadence or any other entity indicates that they would register a new negative vote due to the BNF and typo changes, then I will vote no to even these minor changes.

Formal syntax definition

WAS: The formal syntax of Verilog HDL is described using Backus-Naur Form (BNF).

PROPOSED: The formal syntax of Verilog HDL is described using Backus-Naur Form (BNF). The syntax of Verilog HDL source is derived from the starting symbol source_text. The syntax of a library map file is derived from the starting symbol library_text.

A.1 Source text

A.1.1 Library source text

library_text ::= { library_description~~s~~ }
library_description~~s~~ ::=
  library_declaration
| include_statement
~~| config_declaration~~
library_declaration ::=
  library library_identifier file_path_spec [{ **,** file_path_spec }]
  [**-incdir** file_path_spec [{ **,** file_path_spec }] **;**
include_statement ::= **include** file_path_spec **;**

(note - moved A.1.3 to A.1.2 and renamed the section)
A.1.2 Verilog source text

source_text ::= { description }
description ::=
  module_declaration
| udp_declaration
| config_declaration
module_declaration ::=
  { attribute_instance } module_keyword module_identifier [module_parameter_port_list]
  [list_of_ports] **;** { module_item }
  endmodule
| { attribute_instance } module_keyword module_identifier [module_parameter_port_list]
  [list_of_port_declarations] **;** { non_port_module_item }
  endmodule
module_keyword ::= **module** | **macromodule**

(Renumber A.1.4 & A.1.5)
A.1.3 Module parameters and ports ...

A.1.4 Module items ...

(Renumber A.1.2)
A.1.5 Configuration source text
*(note - no change to **config-enconfig** and **cell** keywords)*
config_declaration ::=
  config config_identifier **;**
  design_statement
  {config_rule_statement}
  endconfig

design_statement ::= **design** { [library_identifier**.**]cell_identifier } **;**
config_rule_statement ::=
PROPOSED: ***(add missing semicolons)***
default_clause liblist_clause **;**
| inst_clause liblist_clause **;**
| inst_clause use_clause **;**
| cell_clause liblist_clause **;**
| cell_clause use_clause **;**
default_clause ::= **default**
inst_clause ::= **instance** inst_name
inst_name ::= topmodule_identifier{**.**instance_identifier}
cell_clause ::= **cell** [library_identifier**.**]cell_identifier
liblist_clause ::= **liblist** [{library_identifier}]
use_clause ::= **use** [library_identifier**.**]cell_identifier[**:config**]

(No change)

Annex B
(normative)
List of keywords

(No change)
cell
config
endconfig

```
escaped_hierarchical_identifier* ::= (From Annex A - A.1.1)
library_text ::=
        { library_description~~s~~ }
library_description~~s~~ ::=
          library_declaration
        | include_statement
~~        | config_declaration~~
library_declaration ::=
        library library_identifier file_path_spec [ { , file_path_spec } ]
              [ -incdir file_path_spec [ { , file_path_spec } ] ;
file_path_spec ::=
        file_path
include_statement ::=
        include <file_path_spec> ;
```

❖ The period in escaped_hierarchical_identifier and escaped_hierarchical_branch shall be preceded by white_space, but shall not be followed by white_space.

Syntax 13-2 Syntax for declaring library in the library map file

```
config_declaration ::= (From Annex A -~~A.1.2~~ A.1.5)
        config config_identifier ;
        design_statement
        {config_rule_statement}
        endconfig
design_statement ::=
        design { [library_identifier.]cell_identifier } ;
config_rule_statement ::=
PROPOSED: (add missing semicolons)
          default_clause liblist_clause ;
        | inst_clause liblist_clause ;
        | inst_clause use_clause ;
        | cell_clause liblist_clause ;
        | cell_clause use_clause ;
```

Syntax 13-4 Syntax for configuration

```
inst_clause ::= (From Annex A - ~~A.1.2~~ A.1.5)
        instance inst_name
inst_name ::=
        topmodule_identifier{.instance_identifier}
```

Syntax 13-6 Syntax for instance clause

```
cell_clause ::= (From Annex A - ~~A.1.2~~ A.1.5)
        cell [ library_identifier.]cell_identifier
```

Syntax 13-7 Syntax for cell clause

```
liblist_clause ::= (From Annex A - ~~A.1.2~~ A.1.5)
        liblist [{library_identifier}]
```

Syntax 13-8 Syntax for liblist clause

```
use_clause ::= (From Annex A - ~~A.1.2~~ A.1.5)
        use [library_identifier.]cell_identifier[:config]
```

Syntax 13-9 Syntax for use clause

13.5.2 Using the default clause

```
    config cfg1;
WAS:  design rtlLib.top
PROPOSED: (add missing semicolon)
        design rtlLib.top;
        default liblist aLib rtlLib;
    endconfig

...

    config cfg2;
WAS:  design rtlLib.top
PROPOSED: (add missing semicolon)
        design rtlLib.top;
        default liblist gateLib aLib rtlLib;
    endconfig
```

13.5.3 Using the cell clause

```
    config cfg3;
WAS:  design rtlLib.top
PROPOSED: (add missing semicolon)
        design rtlLib.top;
        default liblist aLib rtlLib;
        cell foo use gateLib.foo;
    endconfig
```

13.5.4 Using the instance clause

```
    config cfg4
WAS:  design rtlLib.top
PROPOSED: (add missing semicolon)
        design rtlLib.top;
        default liblist gateLib rtlLib;
        instance top.a2 liblist aLib;
    endconfig
```

13.5.5 Using a hierarchical config

```
    config cfg5;
        design aLib.adder;
        default liblist gateLib aLib;
        instance adder.f1 liblist rtlLib;
    endconfig

...

    config cfg6;
        design rtlLib.top;
        default liblist aLib rtlLib;
WAS:    instance top.a2 use work.cfg5:config
PROPOSED: (add missing semicolon)
        instance top.a2 use work.cfg5:config
    endconfig
```