

IEEE Standard VHDL Language Reference Manual

Section Clause¹ 0

Overview of this standard

This ~~section clause~~² describes the purpose and organization of this standard, the IEEE Standard VHDL Language Reference Manual.

0.1 Intent and scope of this document

The intent of this standard is to define VHDL accurately. Its primary audiences are the implementor of tools supporting the language and the advanced user of the language. Other users are encouraged to use commercially available books, tutorials, and classes to learn the language in some detail prior to reading this manual. These resources generally focus on how to use the language, rather than how a VHDL-compliant tool is required to behave.

At the time of its publication, this document was the authoritative definition of VHDL. From time to time, it may become necessary to correct and/or clarify portions of this standard. Such corrections and clarifications may be published in separate documents. Such documents modify this standard at the time of their publication and remain in effect until superseded by subsequent documents or until the standard is officially revised.

0.2 Structure and terminology of this document

This manual is organized into ~~sections clauses~~³, each of which focuses on some particular area of the language. ~~Every fifth line of each section clause~~⁴, ~~not including section clause~~⁵ ~~headings, footers, and the section clause~~⁶ ~~title, is numbered in the left margin.~~⁷ Within each ~~section clause~~⁸, individual constructs or concepts are discussed in each ~~clause subclause~~⁹.

Each ~~clause subclause~~¹⁰ describing a specific construct begins with an introductory paragraph. Next, the syntax of the construct is described using one or more grammatical “productions.”

-
1. To conform to IEEE rules.
 2. To conform to IEEE rules.
 3. To conform to IEEE rules.
 4. To conform to IEEE rules.
 5. To conform to IEEE rules.
 6. To conform to IEEE rules.
 7. The editor has been unable to generate either paragraph or line numbers, so this sentence is, for the time being, removed. He hopes to be able to provide this feature prior to final publication.
 8. To conform to IEEE rules.
 9. To conform to IEEE rules.
 10. To conform to IEEE rules.

A set of paragraphs describing the meaning and restrictions of the construct in narrative form then follow. Unlike many other IEEE standards, which use the verb “shall” to indicate mandatory requirements of the standard and “may” to indicate optional features, the verb “is” is used uniformly throughout this document. In all cases, “is” is to be interpreted as having mandatory weight.

Additionally, the word “must” is used to indicate mandatory weight. This word is preferred over the more common “shall,” as “must” denotes a different meaning to different readers of this standard.

- a) To the developer of tools that process VHDL, “must” denotes a requirement that the standard imposes. The resulting implementation is required to enforce the requirement and to issue an error if the requirement is not met by some VHDL source text.
- b) To the VHDL model developer, “must” denotes that the characteristics of VHDL are natural consequences of the language definition. The model developer is required to adhere to the constraint implied by the characteristic.
- c) To the VHDL model user, “must” denotes that the characteristics of the models are natural consequences of the language definition. The model user can depend on the characteristics of the model implied by its VHDL source text.

Finally, each clause may end with examples, notes, and references to other pertinent clauses.

0.2.1 Syntactic description

The form of a VHDL description is described by means of context-free syntax, using a simple variant of backus naur form; in particular:

- a) Lowercased words in roman font, some containing embedded underlines, are used to denote syntactic categories, for example:

formal_port_list

Whenever the name of a syntactic category is used, apart from the syntax rules themselves, spaces take the place of underlines (thus, “formal port list” would appear in the narrative description when referring to the above syntactic category).

- b) Boldface words are used to denote reserved words, for example:

array

Reserved words must be used only in those places indicated by the syntax.

- c) A *production* consists of a *left-hand side*, the symbol “::=” (which is read as “can be replaced by”), and a *right-hand side*. The left-hand side of a production is always a syntactic category; the right-hand side is a replacement rule.

The meaning of a production is a textual-replacement rule: any occurrence of the left-hand side may be replaced by an instance of the right-hand side.

- d) A vertical bar separates alternative items on the right-hand side of a production unless it occurs immediately after an opening brace, in which case it stands for itself:

letter_or_digit ::= letter | digit

choices ::= choice { | choice }

In the first instance, an occurrence of “letter_or_digit” can be replaced by either “letter” or “digit.” In

the second case, “choices” can be replaced by a list of “choice,” separated by vertical bars (see item f for the meaning of braces).

- e) Square brackets enclose optional items on the right-hand side of a production; thus the two following productions are equivalent:

```
return_statement ::= return [ expression ] ;
```

```
return_statement ::= return ; | return expression ;
```

Note, however, that the initial and terminal square brackets in the right-hand side of the production for signatures (in 2.3.2) are part of the syntax of signatures and do not indicate that the entire right-hand side is optional.

- f) Braces enclose a repeated item or items on the right-hand side of a production. The items may appear zero or more times; the repetitions occur from left to right as with an equivalent left-recursive rule. Thus, the following two productions are equivalent:

```
term ::= factor { multiplying_operator factor }
```

```
term ::= factor | term multiplying_operator factor
```

- g) If the name of any syntactic category starts with an italicized part, it is equivalent to the category name without the italicized part. The italicized part is intended to convey some semantic information. For example, *type_name* and *subtype_name* are both syntactically equivalent to name alone.
- h) The term *simple_name* is used for any occurrence of an identifier that already denotes some declared entity.

0.2.2 Semantic description

The meaning and restrictions of a particular construct are described with a set of narrative rules immediately following the syntactic productions. In these rules, an italicized term indicates the definition of that term and identifiers appearing entirely in uppercase refer to definitions in package STANDARD (see 14.2).

The following terms are used in these semantic descriptions with the following meaning:

erroneous	The condition described represents an ill-formed description; however, implementations are not required to detect and report this condition. Conditions are deemed erroneous only when it is impossible in general to detect the condition during the processing of the language.
error	The condition described represents an ill-formed description; implementations are required to detect the condition and report an error to the user of the tool.
illegal	A synonym for “error.”
legal	The condition described represents a well-formed description.

0.2.3 Front matter, examples, notes, references, and annexes

Prior to this ~~section~~ clause¹¹ are several pieces of introductory material; following the final ~~section~~ clause¹² are some annexes and an index. The front matter, annexes, and index serve to orient and otherwise aid the user of this manual but are not part of the definition of VHDL.

Some ~~clauses~~ subclauses¹³ of this standard contain examples, notes, and cross-references to other ~~clauses~~ subclauses¹⁴ of the manual; these parts always appear at the end of a ~~clause~~ subclause¹⁵. Examples are meant to illustrate the possible forms of the construct described. Illegal examples are italicized. Notes are meant to emphasize consequences of the rules described in the ~~clause~~ subclause¹⁶ or elsewhere. In order to distinguish notes from the other narrative portions of this standard, notes are set as enumerated paragraphs in a font smaller than the rest of the text. Cross-references are meant to guide the user to other relevant ~~clauses~~ subclauses¹⁷ of the manual. Examples, notes, and cross-references are not part of the definition of the language.

-
11. To conform to IEEE rules.
 12. To conform to IEEE rules.
 13. To conform to IEEE rules.
 14. To conform to IEEE rules.
 15. To conform to IEEE rules.
 16. To conform to IEEE rules.
 17. To conform to IEEE rules.