SLOVENSKI STANDARD

SIST-TP CLC/R217-012:2004

september 2004

Specifikacijski jeziki elektronskega sistema - Jedrni informacijski model VDHL 93

Electronic system specification languages - VHDL 93 information core model

ICS 35.060; 35.240.50

Referenčna številka
SIST-TP CLC/R217-012:2004(en)

CENELEC

R217-012

REPORT

November 1996

English version

Electronic system specification languages
VHDL 93 information core model

This CENELEC Report has been prepared by the Technical Committee CENELEC TC 217, Electronic Design Automation (EDA). It was approved by TC 217 on 1996-01-30 and endorsed by the CENELEC Technical Board on 1996-07-02.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland and United Kingdom.

CENELEC

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

Central Secretariat: rue de Stassart 35, B - 1050 Brussels

Ref. No. R217-012:1996 E

Foreword

This report, which was developed by Cristian A. Giumale[1)], was submitted to CENELEC TC 217, Electronic Design Automation.

It was approved by TC 217 on 1996-01-30 and approved for publication by the CENELEC Technical Board on 1996-07-02.

Background

The problems discussed in the present document resulted from work supported by the UK Defence Research Agency to produce comprehensive information models of VHDL'87 and VHDL'93. As the research progressed it became clear that a single model of VHDL is not a satisfactory solution. Instead, a hierarchy of models, each of which describing the relevant aspects relating to a specific application purpose and from a specific perspective of VHDL, is a more satisfactory alternative. However, the model at the apex of this hierarchy describes the essential objects and the semantics of VHDL at the description, analysis, elaboration and simulation levels of design. It can be seen as an abstraction of all the other models in the hierarchy. Such a model of VHDL'93, called here **core** model, and which is part of the present document, is a result of the above mentioned project.

Acknowledgements

Support for the work described has been received from the UK Defence Research Agency and the Commission of the European Communities through the ESIP (ESPRIT 8370) project. We wish to thank in particular Hilary Kahn (Manchester University UK), Serafin Olcoz and Juana Lopez (TGI Spain) for their useful comments and suggestions concerning VHDL modelling, and VHDL itself.

1) University of Manchester, Oxford Road, Manchester M13 9PL UK

Contents

Chapter 1

VHDL modelling focus

The core model of VHDL'93 is based on the core model of VHDL'87 [GC94]. Apart from minor modifications, the main extensions of the model focus the following new features of VHDL'93 (as discussed in [BJM93]).

```
New Simulation Mechanisms
  Postponed Processes
  Shared Variables

New Structuring Mechanisms
  Direct Instantiation
  Incremental Binding

New Interfacing Mechanisms
  Impure Functions

Slight Enhancements
  Inertial Signal Assignments
```


The modelling of the aspects above spans almost the whole model, from the design description level to the elaboration and simulation level. All aspects, except the direct instantiation, are conceptual extensions. Although the direct instantiation mechanism can be considered rather as a programming commodity, it was felt that it deserves explicit specification for outlining its special role in the description of a design.

1.1 Postponed Processes

A postponed process is a process which executes in the last delta cycle of a sequence of simulation delta cycles. The VHDL model is modified and extended as explained below.

1.1.1 Source level processes

The entity process classifies processes according to their description context (passive - for processes contained in an entity declaration, non_passive for other processes) and to their type (postponed, non_postponed).

```
ENTITY process
```