

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

151

R. Piloty M. Barbacci
D. Borrione D. Dietmeyer
F. Hill P. Skelly

CONLAN Report



Springer-Verlag
Berlin Heidelberg New York Tokyo 1983

Editorial Board

D.R. Barstow W. Brauer P. Brinch Hansen D. Gries D.C. Luckham
C. Moler A. Pnueli G. Seegmüller J. Stoer N. Wirth

Authors

Robert Piloty

Institut für Datentechnik, Fachgebiet Rechnerorganisation
TH Darmstadt, Merckstr. 25, 6100 Darmstadt, FRG

Dominique Borriane

IMAG

B.P. 53X, 38041 Grenoble, France

Mario R. Barbacci

Research Computer Scientist

Dept. of Computer Science, Carnegie Mellon University
Pittsburgh, PA 15213, USA

D.L. Dietmeyer

University of Wisconsin Madison

Department of Electrical and Computer Engineering

1425 Johnson Drive, Madison, Wisconsin 53706, USA

Frederick J. Hill

Department of Electrical Engineering, University of Arizona
Tucson, Arizona 85721, USA

Patrick Skelly

3202 West Daily Street

Phoenix, Arizona 85023, USA

CR Subject Classifications (1982): B.5.2, B.6.3, D.3.1, D.3.2, D.3.3,
F.3.1, F.3.2, F.4.3

ISBN 3-540-12275-3 Springer-Verlag Berlin Heidelberg New York Tokyo

ISBN 0-387-12275-3 Springer-Verlag New York Heidelberg Berlin Tokyo

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically those of translation, reprinting, re-use of illustrations, broadcasting, reproduction by photocopying machine or similar means, and storage in data banks. Under § 54 of the German Copyright Law where copies are made for other than private use, a fee is payable to "Verwertungsgesellschaft Wort", Munich.

© by Springer-Verlag Berlin Heidelberg 1983

Printed in Germany

Printing and binding: Beltz Offsetdruck, Hemsbach/Bergstr.
2145/3140-543210

Preface

This report on CONLAN is directed primarily at toolmakers or developers of software for computer aided design of digital systems, which are familiar with specification and implementation techniques for hardware description and programming languages, rather than at the tool user, i.e. hardware designer, himself.

Every effort has been made to eliminate mistakes and ambiguities from the text. The chairman is indebted to Mssrs. K. Hahn and B. Kreling for their careful review of the manuscript. Many improvements are due to them.

Nevertheless the authors are aware that, due to time pressure and to the difficulties of coordinating the work of an international working group spread over two continents, the present version may still lack conciseness in some parts and sufficient illustrating examples in others.

In spite of possible deficiencies in these areas, it was felt that the report should be published as it stands to get exposure to public reaction as early as possible. This is expected to lead to improvements to be incorporated in a final version and in the CONLAN support software which is presently under development.

During this review period it is planned to publish complementing material on informal explanations of CONLAN features, on examples of language definition and application, and on the support software.

R. Piloty
(Chairman, CONLAN Working Group)

Table of Contents

Preface

I Introduction and General Philosophy	1
1. An Introduction to CONLAN	1
1.1. Historical Introduction	1
1.2. Motivation and Objectives	1
1.3. The CONLAN Approach	3
2. CONLAN Concepts	3
2.1. Segments, statements	4
2.2. Concurrency	4
2.3. Structure Description versus Behaviour Description	4
2.4. Language and Description Definition Segments	5
2.5. Type and Operation Definition Segments	7
2.6. Identifiers, Parameters and Interface Carriers	8
2.7. Statements	9
3. A Preview of Primitive Set CONLAN (pscl)	11
3.1. Types, Classes and Operations	11
3.2. pscl Model of Computation	12
4. A Preview of Base Conlan (bcl)	13
4.1. Basic Type Classes	13
4.2. Value Types	14
4.3. Typed Tuples	14
4.4. Arrays and Records	14
4.5. Time and Signals	15
4.6. Carriers	16
4.7. The bcl Model of Computation	17
5. Directions	18
6. References	18
II CONLAN Concepts and Constructs	21
7. Basic Symbols	21
7.1. Characters and Symbols	21
7.2. Keywords, Identifiers, and Comments	23
8. CONLAN entities	24
8.1. Objects	24
8.2. Operations	25
8.3. Types and Classes	26

8.3.1. Object Types	26
8.3.2. Generic Types - Classes	27
8.3.3. Strong Typing - Type Checking	28
8.4. Descriptions and Instances of Descriptions	29
8.5. Languages	30
9. Block Structure	31
9.1. Segments and Statements	31
9.2. Reference Languages	32
9.3. Naming Rules	32
9.3.1. General Rule	33
9.3.2. Scope of Names	33
9.3.3. Rules for Referencing Entities	34
9.3.4. Overloading of Operation Names	34
9.3.5. Implicit Compound Names	34
10. Segments	34
10.1. FUNCTION and ACTIVITY Segments	34
10.1.1. Operation Templates	34
10.1.2. Parameter Lists for Operation Definitions	36
10.1.3. Parameter Lists for Operation Invocation	37
10.1.4. Nesting Rules for FUNCTION and ACTIVITY Segments	38
10.1.5. Importing Rules for FUNCTION and ACTIVITY Segments	38
10.1.6. Exporting Rules for FUNCTION and ACTIVITY Segments	38
10.1.7. STATIC Operations	38
10.1.8. INTERPRETER@ Operations	39
10.1.9. EXTERNAL Operations	39
10.2. DESCRIPTION Segments	39
10.2.1. Description Template	39
10.2.2. Parameter Lists for Description Definitions	40
10.2.3. Nesting Rules for DESCRIPTION Segments	40
10.2.4. Importing Rules for DESCRIPTION Segments	40
10.2.5. Exporting Rules for DESCRIPTION Segments	41
10.2.6. EXTERNAL Descriptions and Operations	41
10.3. TYPE Segments	41
10.3.1. Type Template	41
10.3.2. Nesting Rules for TYPE Segments	42
10.3.3. Importing Rules for TYPE Segments	42
10.3.4. Exporting Rules for TYPE segments	42
10.4. SUBTYPE Segments	43
10.5. Type Relations and Type Conversions	43

10.6. CLASS Segments	45
10.6.1. CLASS Template	45
10.7. CONLAN Segments	46
10.7.1. CONLAN Template	46
10.7.2. Nesting Rules for CONLAN Segments	46
10.7.3. Importing Rules for CONLAN Segments	46
10.7.4. Exporting Rules for CONLAN Segments	47
11. Statements	47
11.1. Operation Invocation Statements	47
11.1.1. Predicate Expressions	47
11.1.2. FORALL@ Predicate	47
11.1.3. FORSOME@ Predicate	48
11.1.4. FORONE@ Predicate	48
11.1.5. Set Enumeration	48
11.1.6. Element Selection	48
11.1.7. Subset Construction	48
11.1.8. Direct Invocation	48
11.1.9. Conditional Invocation	49
11.1.10. Repeated Invocation	49
11.1.11. Result Expression	49
11.2. REFLAN Statements	50
11.3. DECLARE and USE Statements	50
11.3.1. DECLARE Statement	50
11.3.2. USE statement	51
11.4. FORMAT@ Statements	54
11.5. CARRY Statements	55
11.6. ASSERT Statements	57
11.7. IMPORT Statements	57
III Primitive Set CONLAN	59
12. Introduction	59
13. Psci Object Types and Operations	59
13.1. int	59
13.2. bool	61
13.3. string	61
13.4. cell@	61
13.5. tuple@	62
13.6. univ@	62
13.7. Operator Precedence	63

14. Pscl Classes	63
14.1. any@	64
14.2. carr_type	64
15. Pscl Model of Computation	64
15.1. Basic Concepts of the Evaluation Model	64
15.1.1. Context	65
15.1.2. Principles of Messages	65
15.1.3. Objects	66
15.1.4. Principles of Evaluation	66
15.1.4.1. Stage 1: Compile Time Actions	66
15.1.4.2. Stage 2: Activity Parameter Evaluation	67
15.1.4.3. Stage 3: Activity Execution	68
15.2. The Worker Model of Evaluation	68
15.2.1. Messages	69
15.2.2. Behavior of Cells, and their Specific Messages	71
15.2.2.1. Access Rights	71
15.2.2.2. Nature of a Cell	72
15.2.2.3. Exchange of Messages with a Cell Worker	72
15.2.3. Behavior of Operation Workers	74
15.2.3.1. Hiring Local cell and Operation Workers	74
15.2.3.2. Characterization of Operation Worker	75
15.2.3.3. Parameter Evaluation	75
15.2.3.4. Evaluation of an Operation BODY	76
15.2.4. Behavior of DESCRIPTION Workers	78
15.2.4.1. Initialization	78
15.2.4.2. EVAL-EVALUATED Phase	79
15.2.4.3. START-DONE Phase	79
15.3. The Semantics of Pscl in Terms of the Worker Model	79
15.3.1. The Universe of Primitive Objects	79
15.3.2. Primitive Functions on Types other than cell@.	80
15.3.3. Primitive Operations on Type Cell@	80
15.3.3.1. Function empty@	80
15.3.3.2. Function get@	81
15.3.3.3. Operation put@	81
15.3.4. Conditional Expressions	82
15.3.4.1. IF Conditional Expression	82
15.3.4.2. CASE Conditional Expression	83
15.3.5. Predicate, Element and Subset Constructors	84
15.3.5.1. FORALL@ Predicate	84

15.3.5.2. FORSOME@ Predicate	84
15.3.5.3. FORONE@ Predicate	84
15.3.5.4. THE@ Constructor	85
15.3.5.5. ALL Constructor	85
15.3.6. Primitive Conditional Activity Invocations	85
15.3.6.1. IF Conditional	86
15.3.6.2. CASE Conditional	87
15.3.7. Translation of a Segment Definition into the Model	88
15.3.7.1. Translation of a CONLAN Segment	88
15.3.7.2. Translation of a CLASS Segment	88
15.3.7.3. Translation of a TYPE Segment	88
15.3.7.4. Processing ATT Parameters of a FUNCTION, ACTIVITY or DESCRIPTION Segment	89
15.3.7.5. Translation of an ACTIVITY or FUNCTION Segment	89
15.3.7.6. Translation of a Description Segment	94
16. An Example of Type Derivation in PscI	99
IV Base CONLAN	101
17. Introduction	101
17.1. Bcl Object Types and Operations	101
17.1.1. Basic Categories	101
17.1.2. Scalar Data Types	102
17.1.2.1. CLASS val_type	102
17.1.2.2. CLASS sig_type	102
17.1.2.3. CLASS carr_type	103
17.1.3. Constructor Types for Arrays and Records	103
17.1.4. Toolmaker Types	104
17.1.5. User Types	104
17.2. CONLAN Model of Time	104
17.3. System Interfaces	106
17.3.1. Activity error@	106
17.3.2. Real Time Counter t@	107
17.3.3. Parameter matching functions	107
17.3.4. Signal Modification Activities in Carriers	107
17.4. CONLAN Model of Signal Computation	108
18. Formal Construction of Bcl	110
18.1. Reference Language	110
18.2. Subsets and Ranges of Integers	111
18.3. Additional Boolean Operations	111

18.4. Class of Scalar Value Types	112
18.5. Typed Tuples	113
18.6. Range of Integers	115
18.7. Array Dimensions	117
18.8. Indexers	119
18.9. Arrays	121
18.10. Arrays with Fixed Dimensions	130
18.11. Construction of Records	130
18.12. Computation Step Signals	132
18.13. Real Time Signals	133
18.14. Class of Signal Types	138
18.15. Carriers	138
18.16. Class of Carrier Types	142
19. Bcl Model of Computation	143
19.1. Modelling the time behavior of a description written in CONLAN	143
19.1.1. Stabilization of a Description	143
19.1.2. The bcl Step	143
19.2. Additional Mechanisms at the Base CONLAN Level	145
19.2.1. Global Time and Step Counters	145
19.2.2. Local Step Counters	145
19.2.3. Additions to an Operation Task	145
19.3. Evaluation of Operations at the bcl Level	145
19.3.1. Functions	145
19.3.2. Activities	148
19.4. Evaluation of Instances of DESCRIPTION in bcl	148
V Pscl and Bcl Syntax	150
20. Meta Notation	150
21. Syntax	151
21.1. Language Root	151
21.2. Basic Symbols	151
21.3. Constant Denotation	153
21.4. Segments	156
21.4.1. Function	156
21.4.2. Activity	158
21.4.3. Description	159
21.4.4. Type, Subtype and Class	161
21.4.5. Conlan	163
21.5. Statements	163
21.5.1. Expression	163

21.5.2. Operation invocation	166
21.5.3. Declare and Use	168
21.5.4. Syntax Modification	169
Index	172

List of Tables

Table 7-1: ISO 646-IRV Characters and CONLAN Semantics	22
Table 7-2: CONLAN Compound Symbols	23
Table 7-3: Publication Symbols	23
Table 7-4: CONLAN Family Keywords	23
Table 13-1: PscI Types and Operations	60
Table 13-2: Integer Denotation	61
Table 13-3: Operator Precedence	63
Table 14-1: Class Operations	63
Table 17-1: Bcl parameter matching table	107

List of Figures

Figure 2-1: CONLAN text structure	6
Figure 3-1: Type and class structure of pscI	13
Figure 10-1: Function Segment Template	35
Figure 10-2: Activity Segment Template	36
Figure 10-3: Description Segment Template	40
Figure 10-4: Type Segment Template	42
Figure 10-5: Subtype Segment Template	43
Figure 10-6: Class Segment Template	45
Figure 10-7: Language Definition Segment Template	46
Figure 11-1: Module Intercommunication	53
Figure 17-1: bcl derivation Tree	105
Figure 17-2: Examples of 3 time unit CONLAN boolean signal	109