Lecture Notes in Computer Science 891
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Advisory Board: W. Brauer D. Gries J. Stoer

Claus Lewerentz Thomas Lindner (Eds.)

Formal Development of Reactive Systems

Case Study Production Cell

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Series Editors

Gerhard Goos
Universität Karlsruhe
Vincenz-Priessnitz-Straße 3, D-76128 Karlsruhe, Germany

Juris Hartmanis
Department of Computer Science, Cornell University
4130 Upson Hall, Ithaka, NY 14853, USA

Jan van Leeuwen
Department of Computer Science, Utrecht University
Padualaan 14, 3584 CH Utrecht, The Netherlands

Volume Editors

Claus Lewerentz
Lehrstuhl für Software-Technik, Brandenburgische Technische Universität Cottbus
Karl-Marx-Str. 17, D-03044 Cottbus, Germany

Thomas Lindner
Forschungszentrum Informatik, Bereich Programmstrukturen
Haid-und-Neu-Str. 10-14, D-76131 Karlsruhe, Germany

CR Subject Classification (1991): D.2, C.3, D.1, D.3.1, J.6, J.7

ISBN 3-540-58867-1 Springer-Verlag Berlin Heidelberg New York

CIP data applied for

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1995
Printed in Germany

Typesetting: Camera-ready by author
SPIN: 10479269 45/3140-543210 - Printed on acid-free paper

Preface

Mathematically precise methods play an increasingly important role in software development. To a growing extent, industry is utilizing this technology in areas where failure of software would result in injury to people or significant loss of money.

In one of the areas of computer science where a wide variety of methods confuse the practitioner, a comparative study can foster industrial acceptance by clarifying strengths and weaknesses, and elucidating the main ideas on a sample basis. This is the motivation and the goal of this book.

The book is based upon work done within the German KorSo project. We gratefully acknowledge the support given by the German Federal Ministry of Research and Technology (BMFT) with funding this project (grant number 01 IS 203). The editors would like to thank everyone who contributed by specifying, verifying, or implementing control programs. We thank the authors for their instructive contributions.

Special thanks are due to our colleague Emil Sekerinski, who brought this case study to our attention. We thank our colleagues from the Microcomputer Research Group at the Forschungszentrum Informatik, who provided us with a first version of a toy model of the production cell.

We hope that this work will stimulate many discussions and encourage everybody to try out their favourite method on the case study "production cell". We will keep on maintaining an ftp database for task description, simulation, and recording contributed work, be it published papers, reports, or specifications and implementations. Thus, all accumulated contributions may help in putting formal methods into industrial use by showing their benefits, explaining their use, and providing a roadmap for their selection and application.

Karlsruhe, September 1994

Claus Lewerentz
Thomas Lindner

Table of Contents

I. Introduction
Claus Lewerentz, Thomas Lindner 1

1.1 Goal and Context of the Case Study 1
1.2 Problem 2
1.3 Required Properties 3
1.4 Validation Environment: Toy Model and Graphic Simulation 4

II. Task Description
Thomas Lindner 7

2.1 Description of the Production Cell 7
2.2 Actuators and Sensors 13
2.3 Requirements 15

III. Comparative Survey
Claus Lewerentz, Thomas Lindner 21

3.1 Criteria 22
3.2 Contributions 24
3.3 Conclusions 49

IV. CSL
Klaus Nökel, Klaus Winkelmann 55

4.1 Goals of the CSL Contribution 55
4.2 A First Look at CSL 56
4.3 Modelling the Production Cell 59
4.4 Verification Using Model Checking 64
4.5 Evaluation 71
4.6 Conclusion 73

V. ESTEREL
Reinhard Budde 75

5.1 A Quick Introduction to Esterel 75
5.2 The Production Cell in Esterel 83
5.3 Structure and Properties of the Production Cell 87
5.4 Remark About Collisions and Systems Engineering 90
5.5 From Esterel Programs to Executable Code 92
5.6 Experiences and Conclusion 94

VI. LUSTRE
Leszek Holenderski 101

6.1 A Quick Introduction to LUSTRE 101
6.2 Production Cell in LUSTRE 104
6.3 Verification 109
6.4 Evaluation 111

VII. SIGNAL
Tochéou Pascalin Amagbegnon, Paul Le Guernic, Hervé Marchand, Éric Rutten 113

7.1 A Brief Introduction to Signal 113
7.2 Specification of the Production Cell in Signal 117
7.3 Specifying and Verifying the Properties 123
7.4 Evaluation 128

VIII. Statecharts
Werner Damm, Hardi Hungar, Peter Kelb, Rainer Schlör 131

8.1 Introduction 131
8.2 Statecharts 133
8.3 Requirement Specifications Using Symbolic Timing Diagrams 138
8.4 Verification by Symbolic Model Checking 141
8.5 Conclusion 147

IX. TLT
Jorge Cuellar, Martin Huber 151

9.1 Introduction 151
9.2 An Abstract Centralized Controller 152
9.3 The Environment of the Controller: Dealing with Sensors and Actuators 158
9.4 Distributing the Solution 161
9.5 Refining the Program: Safe Movements 164
9.6 Running the Simulation 167
9.7 Conclusion 167

X. SDL
Stefan Heinkel, Thomas Lindner 171

10.1 Aim 171
10.2 SDL 172
10.3 Production Cell in SDL 175
10.4 Evaluation 180
10.5 Conclusion 182

XI. Focus
Max Fuchs, Jan Philipps 185

11.1 Introduction 185
11.2 Method 186
11.3 Application of the Method 190
11.4 Conclusion 195

XII. SPECTRUM
Dimitris Dranidis, Stefan Gastinger 199

12.1 Introduction 199
12.2 A CSP Modelling 201
12.3 A Stream Based Specification 212
12.4 Evaluation 225
12.5 Conclusion and Remarks 227

XIII. KIV
Gerhard Schellhorn, Axel Burandt 229

13.1 Introduction 229
13.2 The Intended Model 230
13.3 Algebraic Specification of State Oriented Systems 231
13.4 The Control Program and its Correctness 238
13.5 Verification 242
13.6 Conclusions and Further Work 244

XIV. Tatzelwurm
Stefan Klingenbeck, Thomas Käufl 247

14.1 Introduction 247
14.2 The Program Verifier *Tatzelwurm* 248
14.3 Formal Description of the Production Cell 251
14.4 The Implementation 254
14.5 The Safety Requirements 257
14.6 Results 258
14.7 Conclusions 258

XV. HTTDs and HOL
Rachel Cardell-Oliver 261

15.1 Introduction 261
15.2 Method 262
15.3 Specification 264
15.4 Verification 268
15.5 Results 272
15.6 Conclusions 273

XVI. RAISE

François Erasmy, Emil Sekerinski 277

16.1 Goal of the RAISE Contribution 277
16.2 Introduction to RAISE 278
16.3 Development with RAISE 281
16.4 Evaluation of the Refinement Approach with RAISE 289
16.5 Conclusion 292

XVII. Deductive Synthesis

Jochen Burghardt 295

17.1 Aim 295
17.2 Deductive Program Synthesis 296
17.3 Modelling the Production Cell 297
17.4 Synthesis 301
17.5 Evaluation 305
17.6 Conclusion 308

XVIII. Symbolic Timing Diagrams

Franz Korf, Rainer Schlör 311

18.1 Introduction 311
18.2 Symbolic Timing Diagrams 315
18.3 Synthesis Path and Design Methodology 319
18.4 Using ICOS^2 for the Production Cell Case Study 320
18.5 Synthesis Process and Synthesis Techniques 323
18.6 Discussion of the Complexity 326
18.7 Experimental Results of the Production Cell 328
18.8 Conclusion 329

XIX. LCM and MCM

Roel Wieringa 333

19.1 Introduction 333
19.2 Method for Conceptual Modeling (MCM) 334
19.3 Language for Conceptual Modeling (LCM 3.0) 341
19.4 Verification of Safety and Liveness 348
19.5 Discussion of the Specification 350
19.6 Conclusions 353

XX. Modula-3
Andreas Rüping, Emil Sekerinski 357

20.1 Introduction 357
20.2 Modelling 359
20.3 Verification 363
20.4 Implementation 367
20.5 Evaluation and Conclusions 369

XXI. TROLL *light*
Rudolf Herzig, Nikolaos Vlachantonis 373

21.1 Goals and Concepts of TROLL *light* 373
21.2 The Specification 376
21.3 Evaluation 379

A. Simulation
Artur Brauer, Thomas Lindner 383

A.1 Problem 383
A.2 Simulation 384
A.3 Installing the Simulation 385
A.4 Starting the Simulation 386
A.5 Definition of the Protocol 386
A.6 Operating the Cell 392