Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

404

Joachim Beer

Concepts, Design, and Performance Analysis of a Parallel Prolog Machine

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

Editorial Board

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

Author

Joachim Beer
International Computer Science Institute
1947 Center Street, Berkeley, CA 94707, USA

CR Subject Classification (1987): C.1.2, D.3, I.2.5

ISBN 3-540-52053-8 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-52053-8 Springer-Verlag New York Berlin Heidelberg

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 other ways, and storage in data banks. Duplication of this publication or parts thereof is only permitted under the provisions of the German Copyright Law of September 9, 1965, in its version of June 24, 1985, and a copyright fee must always be paid. Violations fall under the prosecution act of the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1989
Printed in Germany

Printing and binding: Druckhaus Beltz, Hemsbach/Bergstr.
2145/3140-543210 – Printed on acid-free paper

Preface

The research presented in this thesis was carried out in conjunction with a joint research project between the GMD - Research Center for Innovative Computer Systems and Computer Technology at the Technical University Berlin, the University of Kaiserslautern, NIXDORF Computer AG, and Epsilon GmbH.

Of all the people involved in this research project I owe the most to Professor Dr.-Ing. W.K. Giloi who trusted me enough to let me lead a research group at the GMD-FIRST even though I was a newcomer to his institute. He also showed enough faith in my ideas to let me go on when others had their doubts.

I also wish to acknowledge the support of all my colleagues during the time this work was done. I would especially like to express my gratitude to Mrs. G. Tysper and Mr. Abel who helped me tremendously with the administrative duties a project leader has to live with. I gratefully acknowledge many discussions with the scientists and engineers from the other research partners who always showed a keen interest in my work.

This project was funded by the German Ministry of Research and Technology (BMFT) and NIXDORF Computer AG to whom I express my gratitude for their financial support.

Last but not least I would like to thank my wife and my little son for their support and their understanding when daddy could not take them to the playground but had to work.

Berkeley, October 1989 Joachim Beer

Contents

Preface iii

1 Introduction
1.1 Outline of the Thesis 1
1.2 Historical Perspective and Foundations 2
1.3 Parallel Prolog Dialects 6

2 Compilation of Sequential Prolog
2.1 Sequential Prolog 10
2.2 General Principles of Prolog Compiler Construction 12
2.2.1 Data Typing 14
2.2.2 Runtime Memory Management 15
2.2.3 Compilation of the Unification Algorithm 18
2.2.4 Clause Indexing 19
2.2.5 Tail Recursion Optimization and Mode Declarations 20

3 Pipelined Execution of Sequential Prolog
3.1 Introduction 21
3.2 A Pipelined Architecture for Prolog 23
3.3 Execution Model 27
3.3.1 Processor-Processor Communication and Synchronization . 31
3.3.2 Implementation of Backtracking 37
3.3.3 AND-Parallelism 38
3.4 Summary 40

4 The Occur-Check Problem
4.1 Introduction 42
4.2 A Variable Classification Scheme 44
4.3 Variables as Structure Arguments 49
4.4 The Occur-Check Problem 53
4.5 Summary 56

5 The Abstract Parallel Prolog Machine
5.1 Data Representation 57
5.2 Registers and Status Flags 59
5.3 The Parallel Instruction Set 62
5.3.1 Control Instructions 67
5.3.2 Goal Instructions 70
5.3.3 Clause Head Instructions 74
5.3.4 Structure Instructions 78
5.3.5 Clause Control Instructions 83
5.3.6 Fork/Join Instructions 86

5.3.7 Miscellaneous Instructions and Micro-Operations 88
5.3.8 Built-in Predicates 91

6 Simulation
6.1 Introduction 92
6.2 Simulated Hardware Structures 92
6.2.1 Pipeline Buffer 95
6.2.2 Execution Unit 97
6.2.3 Instruction Fetch Unit 99
6.3 Simulation Results 101

7 Conclusion
7.1 Contributions 109
7.2 Future Work 109

A Benchmark Programs **110**

B Compilation Examples **115**

References **125**