

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

195

Heinrich J. Stüttgen

A Hierarchical Associative
Processing System



Springer-Verlag
Berlin Heidelberg New York Tokyo

Editorial Board

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

Author

Heinrich J. Stüttgen
Universität Dortmund, Fachbereich Informatik
Postfach 500 500, D-4600 Dortmund 50

CR Subject Classifications (1985): B.3.2, C.1.2, D.4.2, D.4.3, H.2.6

ISBN 3-540-15652-6 Springer-Verlag Berlin Heidelberg New York Tokyo
ISBN 0-387-15652-6 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 1985
Printed in Germany

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

Acknowledgement

I would like to thank Professor Dr. Lutz Richter for his guidance in carrying out this research. Thanks are also due to Dr. Alfons Steinhoff and Dipl.-Inform. Michael Goedicke as well as numerous other people for their valuable comments and suggestions on various aspects of this project. Further I am indebted to Elke Schickentanz, Wolfgang Deiters Frank Paszota, and Burkhard Peuschel for their assistance in preparing the manuscript. Last not least, I thank my wife Evelyn for the improvements of my writing style and for her continuous encouragement during the course of this work.

Abstract

The concept of associative or content addressable memories (CAMs) has been around for over 25 years. They are generally considered an attractive but at the same time very expensive type of memory. Therefore any CAM organization has been tailored to and restricted to a specific application. In this book we describe the design of a two level associative memory hierarchy which is aimed at a broad spectrum of applications. The design is based on state of the art technology and configurable to a wide range of cost/performance requirements.

After a short review of the logical aspects of memories we survey and classify existing designs and implementations of the associative concept, with regard to programming languages and hardware organizations.

The actual design is top-down structured and consists of four levels.

The first level is the programming language PASCAL/A, which is an extension of standard PASCAL by a few powerful constructs providing associative data structures and processing. The second level is an intermediate language; suitable for direct interpretation on an associative processor. Third we describe the architecture of a two-level hardware associative memory. The first level is a processing ensemble built of single chip processors with on-chip memory. The second level is a cellular logic organization based on modified movable head disks. Fourth we suggest a memory hierarchy management scheme that employs staging and which is data structure and program semantic oriented. Finally we point out some open questions and make some suggestions for future research in the area.

Contents:

1. Introduction	1
1.1 Motivation	1
1.2 Survey of Project	3
1.3 Relations to Past Research and Accomplishments	5
2. Classification of Storage Access Philosophies	9
2.1 Motivation	9
2.2 Location Addressable Memory (LAM)	9
2.3 Content Addressable Memory (CAM)	10
2.4 Age Addressable Memory (AAM)	13
2.5 Other Access Mechanisms	14
2.6 Miscellaneous Aspects	16
3. Survey of Relevant Previous Work	17
3.1 Programming Languages Supporting Content Addressing	17
3.1.1 Languages for Array Processors	17
3.1.1.1 PFOR	18
3.1.1.2 ACTUS	21
3.1.2 Triple Languages	23
3.1.3 Relational Languages	25
3.1.3.1 The Relational Model of Data	25
3.1.3.2 PASCAL/R	27
3.2 Associative Memories and Processors	29
3.2.1 Terminology and Classification	29
3.2.2 Exact Match CAMs (F1)	34
3.2.3 General Comparison CAMs (F2)	38
3.2.4 Restricted Content Addressable Processors (RCAPs,F3)	40
3.2.5 Content Addressable Processors (CAPs,F4)	46
3.2.6 Comparison of Different CAM Architectures	56
3.3 Multilevel Associative Memories	60
3.3.1 The STARAN Data Management System (ST-DMS)	60
3.3.2 The Relational Associative Processor (RAP)	61
Virtual Memory System	
3.3.3 The OHIO Database Computer (DBC)	63
3.3.4 Shaw's Hierarchical Associative	65
Architecture for Relational Databases	
3.3.5 Gertz's Hierarchical Associative Memory	67
for Parallel Computation	

4. PASCAL/A	70
4.1 Design Goals	70
4.2 Features	71
4.2.1 Data Definition	71
4.2.2 Interfacing Associative Data Structures	72
4.2.3 Associative Processing	73
4.2.4 Control of Active Subsets	74
4.2.5 Procedures and Functions	77
4.2.6 Predefined Functions	78
4.3 Programming Techniques and Example	80
 5. Primitives for the Implementation of PASCAL/A	 83
5.1 Description of Primitives	83
5.1.1 Primitives for Data Definitions	84
5.1.1.1 Create	84
5.1.1.2 Bind	84
5.1.1.3 Start-Domain	85
5.1.1.4 Destroy	85
5.1.2 Interfacing Primitives	86
5.1.2.1 Allocate	86
5.1.2.2 Put	86
5.1.2.3 Get	86
5.1.2.4 Delete	86
5.1.3 Arithmetical Primitives	87
5.1.3.1 Addition	88
5.1.3.2 Subtraction	88
5.1.3.3 Multiplication	89
5.1.3.4 Division	89
5.1.3.5 Modulo	89
5.1.3.6 Absolute	89
5.1.3.7 Equal	89
5.1.3.8 Not Equal	90
5.1.3.9 Greater	90
5.1.3.10 Less	90
5.1.3.11 Greater or Equal	90
5.1.3.12 Less or Equal	90

5.1.4 Primitives for Associative Control	91
5.1.4.1 Select	91
5.1.4.2 Push	91
5.1.4.3 Pop	91
5.1.4.4 Exclude	92
5.1.4.5 Select-First	92
5.1.4.6 Int-Max	92
5.1.4.7 Int-Min	93
5.1.5 Primitives for Element Size Control	93
5.1.5.1 Extend	93
5.1.5.2 Shrink	93
5.1.6 Primitives for Aggregate Functions	94
5.1.6.1 Match	94
5.1.6.2 Count	94
5.1.6.3 Empty	94
5.1.6.4 Size	94
5.2 Implementation of PASCAL/A Constructs	95
5.2.1 Translation of Data Definitions	95
5.2.2 Translation of Domain Switches	95
5.2.3 Translation of Interfacing Constructs	96
5.2.4 Translation of Integer Expressions	96
5.2.5 Translation of Control Structures	97
5.2.6 Translation of Parallel Procedures and Functions	100
5.2.7 Translation of Predefined Functions	102
 6. Design of the Memory Architecture	 103
6.1 Design of the First CAM-Level	104
6.1.1 Choice of Technology	104
6.1.2 Architecture of the Primary CAM	107
6.1.3 Primary CAM Instruction Set	112
6.2 Secondary CAM-Design	121
6.2.1 Choice of Technology	121
6.2.2 Survey of the Secondary CAM	122
6.2.3 Architecture of the Secondary Control Unit (SCU)	130
6.2.4 Architecture of the Interface Modules (IMs)	131
6.2.5 Architecture of the Track Information Processors (TIPs)	139

7. Implementation of Primitives	141
7.1 Implementation of Primitives on the First CAM Level	142
7.1.1 Storage Layout and Directory Structure	142
7.1.2 Data Representation	147
7.1.3 Data Definition Primitives	150
7.1.4 Interfacing Primitives	160
7.1.5 Arithmetic and Logical Primitives	167
7.1.6 Associative Control	171
7.1.7 Element Size Control Primitives	176
7.1.8 Aggregate Functions	177
7.2 Implementation of Primitives on Second CAM Level	179
7.2.1 Storage Layout and Directory Structure	179
7.2.2 Basic TIP Operations	183
7.2.3 Data Definition Primitives	203
7.2.4 Interfacing Primitives	211
7.2.5 Arithmetic and Logical Primitives	219
7.2.6 Associative Control Primitives	220
7.2.7 Size Control Primitives	225
7.2.8 Aggregate Functions	228
8. Memory Management	230
8.1 Survey of the Memory Manager	230
8.2 Level Management	232
8.3 Hierarchy Management	237
8.3.1 The Staging Concept	237
8.3.2 Hierarchy Management Functions	239
8.4 Space Management	249
8.4.1 First Level Space Management	249
8.4.2 Second Level Space Management	252
9. Summary and Suggestions for Future Research	255
A. Appendix	257
A.1 List of Names and Abbreviations	257
A.2 List of Constants	258
A.3 Instruction Sets	258
A.4 Directories	264
B. Bibliography	266

List of Figures

2- 1	Catalog CAM	12
2- 2	Memory Classification Tree	15
3- 1	Intel 3104 Logic Diagram	35
3- 2	Distributed Logic Memory	36
3- 3	REM - S100 Storage Organization	38
3- 4	SURE System Architecture	39
3- 5	Cellular Logic Device Cell Structure	41
3- 6	CASSM System Architecture	42
3- 7	RAP System Architecture	43
3- 8	ECAM System Architecture	44
3- 9	RARES Storage Layout	45
3-10	ALAP Cell Architecture	47
3-11	PEPE System Architecture	48
3-12	STARAN System Architecture	50
3-13	STARAN Array Module	51
3-14	STARAN Processing Element	51
3-15	STARAN Multi-Dimensional Access Memory	52
3-16	EXOR Skew Network	53
3-17	VASTOR System Architecture	54
3-18	Associative Movable Head Disk	55
3-19	Organization vs. Functionality Diagram	57
3-20	Processing Time vs. Logic per Bit Diagram (I)	59
3-21	Processing Time vs. Logic per Bit Diagram (II)	59
3-22	STARAN Data Management System	60
3-23	RAP Virtual Memory Organization	62
3-24	DBC Mass Memory Organization	63
3-25	DBC System Architecture	64

XII

6- 1	First Level CAM Structure	107
6- 2	First Level CAM Processing Element	113
6- 3	Two Level CAM System Architecture	122
6- 4	Level Interconnection Scheme	125
6- 5	Interface Module Memory Allocation Scheme	127
6- 6	Secondary CAM Match Indicator	140
7- 1	First Level CAM Layout	142
7- 2	First Level Block Organization	143
7- 3	First Level Element Data Format	148
7- 4	First Level Activity Stack Scheme	171
7- 5	TIP Instruction Format	184
8- 1	Memory Management Function Groups	231
8- 2	Hierarchy Management Functions	240

List of Tables

6- 1	First Level Instruction Modes	116
6- 2	Flag Functions	117
6- 3	Communication Functions	118
6- 4	First Level I/O Processing Scheme	119
6- 5	I/O - Time Requirements	119
6- 6	Arithmetic and Logical Functions	120
6- 7	Interface Module Registers	132
7- 1	First Level Directory	146
7- 2	Tag Bit Combinations	149
7- 3	Second Level Directories	182
7- 4	Local TIP Operations	186
7- 5	Vertical Tip Operations	187
7- 6	Administrative TIP Operations	191