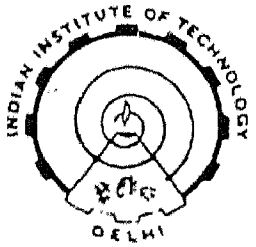
AUGMENTED BINARY TREE ARCHITECTURES AND THEIR FAULT TOLERANCE

BY

RAVI MITTAL
Department of Electrical Engineering

Thesis submitted in fulfilment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

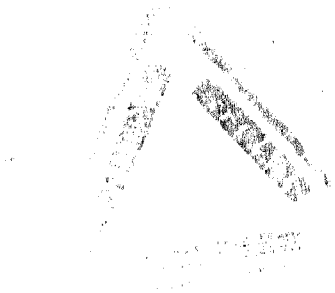


to the

INDIAN INSTITUTE OF TECHNOLOGY, DELHI
HAUZ KHAS, NEW DELHI 110 016
August 1990

To

My Father and Mother



Acknowledgements

It is my privilege to profusely thank Prof. B. N. Jain for his invaluable guidance, encouragement, and help all through the course of my doctoral program. He has been a source of inspiration. My association with him has benefitted me academically and otherwise. My intellectual debt to him, can never be repaid.

I am grateful to Dr. R. K. Patney who guided and encouraged this work to completion. He has been extremely supportive of the path I have taken. I sincerely thank him for careful review of the thesis and useful suggestions.

Prof. D. P. Agrawal (North Carolina Univ), Prof. A. D. Singh (Univ of Massachusetts), and Prof. A. K. Agarawala (Univ. of Maryland) have offered their valuable comments at various stages of this work. It is a delight to express my thanks to them all. I wish to express my thanks to Pratul, Cheriyan, Saxena, Dr. Samar Sing, for their useful discussion and comments.

Prof. N. K. Nanda (Univ of Roorkee) introduced me to the fascinating field of Fault tolerant Computing during my M. Tech. program. I am indebted to him.

I sincerely thank my friends Anand, Sandeep, Suresh, Rajiv, Rao, Praveen, and Ramana, who always considered me a member of the ERNET-lab.

I am grateful to Prof. R. Nagaraja, Head of the Department of Computer Science & Engineering, I. I. T. Madras, for his encouragement and help during my stay at I. I. T. Madras.

All the time during my Ph.D., my sister Maneesha has been very encouraging and supportive. I wish to thank her, my brother-in-law Nisheet, Ammajee, and dear Golu for their love and affection.

Nitin Bhaiyya and Veena Bhabhi have helped me whenever I needed. I express my gratitude to them.

It was great to have friends like Vijay Babbar, Narayan Vikas, Ashwani, Nishant, and Mirza, who have made this period joyous and cheerful.

My family members have been a great source of love, inspiration, and strength. I thank my brother Rakesh, Pawan jee, Neeta Didi, Ajai, and Nidhi for their affection and support all along. I thank my wife for her loving care and patience.

My parents have always encouraged and supported my endeavors and have tried to make me an honest and hardworking man. Without their blessings this work would not have been possible.

Ravi Mittal

INDIAN INSTITUTE OF TECHNOLOGY, DELHI
NEW DELHI 110 016, INDIA

CERTIFICATE

This is to certify that the thesis entitled *"Augmented Binary Tree Architectures and their Fault Tolerance"* being submitted by Ravi Mittal to the Department of Electrical Engineering, Indian Institute of Technology, New Delhi, for the award of the degree of Doctor of Philosophy, is a record of bona fide research work carried out by him under our supervision and guidance and in our opinion, it has reached the standard fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted to any other university or institute for the award of any degree or diploma.

B. N. Jain
Professor
Department of Computer
Science & Engineering

R. K. Patney
Asst. Professor
Department of
Electrical Engineering

ABSTRACT

A full binary tree is a widely used interconnection network architecture for parallel processing. This thesis presents a study of the fault tolerance of some proposed augmented binary tree architectures. The full binary tree is augmented in a variety of ways with a view to improve its fault tolerance and other properties.

We propose a Link Augmented Binary (LAB)-tree as a fault tolerant and an efficient communication architecture. This tree is regular in topology, easily expandable and can be constructed modularly. The LAB-tree contains multiple node-disjoint paths for each pair of nodes. Hence, in the presence of one faulty node, the tree remains connected. The tree supports a simple algorithm for shortest path routing. This algorithm has been modified so as to route messages through a faulty tree. Compared to some other tree architectures, the LAB-tree is shown to have a superior traffic handling capability.

The LAB-tree architecture is further augmented with redundant nodes. Such a tree is called Node Augmented Binary (NAB)-tree architecture. In the presence of one faulty node at each level, this tree can be configured into a non-faulty full binary tree of the same height. Simple switch structures associated with nodes of the tree are proposed. The topology of the NAB-tree is regular. This tree can also be used for efficient routing of messages.

A systematic and generalized approach for designing a modular fault tolerant binary tree architecture, is proposed. The new architecture based on this approach, is called MOD-tree. A MOD-tree consists of modules connected in the topology of a tree. It can be configured to a full binary tree in the presence of multiple faulty nodes in its modules. Faulty nodes in a module can be replaced by spare

(redundant) non-faulty nodes in the same module and/or spare nodes from other modules. Using the global reconfiguration algorithm, spare nodes of a module can be used by its ancestor modules. The reliability of this tree is shown to be higher than that of other modular tree architectures. Using this approach, a wide choice in the size of the modules to construct a MOD-tree is available. A MOD-tree can be constructed using efficient VLSI layouts with the area proportional to the number of nodes in the tree.

In the presence of a large number of faults, an augmented binary tree can not be configured into a full binary tree, then one may identify and use one of the non-faulty (full) binary subtrees within it. We have given an algorithm for finding the largest non-faulty subtree available in a faulty tree. The minimum number of faulty nodes which may destroy all subtrees of a given height is also computed. We also study the availability of a non-faulty subtree in some known augmented tree architectures.

Contents

Certificate i
Acknowledgement ii
Abstract iii
Contents v
List of Abbreviations and Symbols viii
List of Figures x

CHAPTER 1. INTRODUCTION
1.1 Parallel processing Architectures 2
1.2 Fault Tolerance 3
1.3 Binary Tree Architectures 7
1.4 Fault Tolerance in Binary Tree Architectures 11
1.5 Thesis Outline 16

CHAPTER 2. LINK AUGMENTED BINARY TREE ARCHITECTURE
2.1 Introduction 18
2.2 The LAB-Tree Architecture 19
2.3 Fault Tolerance of the LAB-tree 27
2.4 Routing Algorithms and Traffic Characteristics 39
2.4.1 Routing Algorithm LABROUTE 1 40
2.4.2 Fault Tolerant Routing Algorithm LABROUTE 2 46
2.4.3 Traffic Characteristics 47
2.5 Other Properties of the LAB-tree 47
2.5.1 Configuring LAB-tree to full binary tree 47
2.5.2 Modular Implementation of the LAB-tree 51
2.6 Comparison with other architectures 55
2.7 Summary and Discussion 56

Appendix 2.1 The proof of the Lemmas 2.2, 2.3, and 2.6 58
Appendix 2.2 Fault tolerant routing algorithms 60

CHAPTER 3. NODE AUGMENTED BINARY TREE ARCHITECTURE
3.1 Introduction 63
3.2 The Node Augmented Binary Tree Architecture 64
3.3 Obtaining Full Binary Tree from NAB-tree 65
3.3.1 Definitions and some preliminaries 67
3.3.2 Reconfiguration Algorithms 75
Centralized Reconfiguration Algorithm 76
Distributed Reconfiguration Algorithm 77
3.4 Distributed Fault Diagnosis 80
3.5 Routing and Message Traffic Characteristics 84
3.6 Switch Structure 84
3.7 VLSI Implementation 86
3.8 Comparison with other Architectures 86
3.9 Summary and Discussion 88

CHAPTER 4. MOD-TREE: A MODULAR FAULT TOLERANT BINARY TREE ARCHITECTURE
4.1 Modular Tree Architecture 91
4.1.1 Basic Concept of MOD-tree 92
4.2 MOD-trees based on 2-node structures 95
4.2.1 Realization of 2-node structure 95
4.2.2 Obtaining full binary tree from P3-MOD-tree 97
Local Reconfiguration Scheme 100
Global Reconfiguration Scheme 103
4.2.3 Switch Structure 110
4.2.4 Reliability Analysis of the P3-MOD-tree 110
4.2.5 Some comments on Realization of 2-node structure 114
4.3 MOD-trees based on 4-node structures 114
4.3.1 Realization of 4-node structure 114
4.3.2 Reconfiguration of P7-MOD-tree to full binary tree 116
Local Reconfiguration Scheme 122
Global Reconfiguration Scheme 125
4.3.3 Some Comments on P7-MOD-tree 127
4.3.4 Two variations of P7-MOD-tree architecture 129
VLSI layout 131

4.5 Comparison with other architectures 131
4.6 Summary and Discussion 133

Appendix 4.1 All possible configurations of a P3-module and a P7-module 135
Appendix 4.2 The detailed assignment phase of algorithm P7-GLOBAL-RECON 137
Appendix 4.3 Global reconfiguration algorithm for MOD-tree I 140
Appendix 4.4 Local Reconfiguration of the AR-module of MOD-tree II 142

CHAPTER 5. SUBTREE ORIENTED FAULT TOLERANCE IN BINARY TREE MULTIPROCESSOR ARCHITECTURE
5.1 Introduction 144
5.2 Minimum critical fault set 147
5.3 Obtaining non-faulty subtrees 150
5.4 Improvements in subtree availability 151
5.4 Summary and Discussion 159

CHAPTER 6. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK
6.1 Summary of results 160
6.2 Recommendations for future work 165

References 167
Curriculum vitae 178