ON THE DESIGN OF MULTI-MICROPROCESSOR SYSTEM FOR REAL-TIME APPLICATIONS

by

AMITABHA SINHA

A thesis submitted in partial fulfilment of
the requirements of
the degree of
DOCTOR OF PHILOSOPHY

to the

INDIAN INSTITUTE OF TECHNOLOGY, DELHI

JANUARY 1983

To my parents

CERTIFICATE

Certified that the thesis "On The Design of Multi-Microprocessor System For Real-Time Applications", which is being submitted by Mr. Amitabha Sinha for the award of the degree of Doctor of Philosophy to the Indian Institute of Technology, Delhi, is a record of student's own work carried out by him under our supervision and guidance. The matter embodied in this thesis has not been submitted for the award of any other degree or diploma.

New Delhi
January, 1982

(M. IBRAMSHA)
Assistant Professor
Centre for Computer Science and Engineering
I.I.T., Delhi.

(J.C. SHOURI)
Professor
Centre for Computer Science and Engineering
I.I.T., Delhi

ACKNOWLEDGEMENT

It is a great pleasure to express my sense of gratitude to Dr. M. Ibramsha and Prof. J.C. Shouri for their able guidance and help throughout the investigation. I am specially indebted to Dr. M Ibramsha for his continual encouragement, keen interest and many valuable discussions during the period of my research work.

I wish to extend my thanks to Prof. R. Arockiasamy for introducing me the various real-time applications of Multi-Microprocessor Systems.

Thanks are also extended to Dr. J.G. Deshpande and Dr. B.N. Jain for their co-operation throughout the preparation of this thesis.

I am thankful to Dr. S.P. Rana for his keen interest in the present work. The co-operation received from my friends Arun, Nirmal, Niraj, Subhash and Prasad are sincerely appreciated.

My family members have in no small way contributed towards the completion of this work. To Subhra, my wife, is reserved my deepest appreciation for her patience, forebearance and understanding.

Finally, thanks are also due to Mr. Ranjit Kumar and Mr. Aloke Biswas who painstakingly typed and sketched the diagrams of the manuscript respectively.

Sinha

(A. SINHA)
Centre for Computer Science
and Engineering,
Indian Institute of Technology
Hauz Khas, New Delhi-110016.

Abstract

There exists a class of real-time algorithms which needs very high computing speed beyond the levels achievable by current technology. If an algorithm can be rewritten so that its inherent parallelism can be taken advantage of by parallel computers like SIMD, MIMD and Systolic machines, we can achieve the necessary levels of computing speed. As such, developing algorithms that can run on parallel computers is still an art in contrast to sequential programming which is very well developed. Real-time application programs are invariably developed as sequential algorithms having very little parallelism.

In this thesis we have considered the problem of designing a computer architecture other than SIMD, MIMD and Systolic which facilitates the speeding up of sequential algorithms. Sequential algorithms like Fast Fourier Transform (FFT) can be executed at very high speeds by a pipelined computer system. Algorithms like FFT can be divided into a number of stages such that any given stage communicates with the immediate next stage. It may not be possible to split a general sequential algorithm into such stages. In this thesis we generalize the concept of pipeline architecture to accommodate any sequential algorithm.

In the proposed design a given sequential algorithm is split into the desired number of stages such that the time of execution of each stage is equal to the data arrival time. Each stage so identified is executed by one microprocessor. If such splitting is done at logical interface, a processor needs to communicate to one or more of its successor processors partial results only. However, such splitting can not be carried out always at logical interfaces in the interest of maximizing the speed. This necessitates additionally the communication of processor status to the succeeding processor. For the currently available microprocessors, transfer of processor status information from one processor to another is time consuming. To overcome this limitation the concept of 'Virtual Pipelining' is introduced.

ooooo

CONTENTS

		Page
CHAPTER 1:	INTRODUCTION	1
1.1	Introduction	1
1.2	Microprocessors and their capabilities	3
1.3	Motivation behind the design of digital system using LSI	4
1.4	LSI based modular Architecture, a new way of Computer design	5
1.5	Different types of parallel processing machines	5
1.6	Survey of the related work	8
1.7	Overview of the thesis	17
CHAPTER 2:	GENERALIZING THE CONCEPT OF A CONVENTIONAL PIPELINED SYSTEM	19
2.1	Introduction	19
2.2	Pipelining system	20
2.3	Partitioning of linear algorithm into sub-processes	21
2.4	Strategy for splitting a given linear algorithm into segments	23
	2.4.1 Segmentation using the variable reference table	27
2.5	Pipelined processing for tree structured algorithm	35
	2.5.1 Discussion and proposed solution	36
	2.5.2 Pass I: Construction of variable table for a given path	41
	2.5.3 Highlights of Pass II	42
2.6	Handling the structured network algorithms	44
2.7	Handling the loop algorithms	49
2.8	Overhead for data transferring	52
2.9	Conclusion	53

		Page
CHAPTER 3:	DATA COMMUNICATION PIPELINED SYSTEM	54
3.1	Introduction	54
3.2	Discussion on splitting of a given linear algorithm using the concept of data communication among the non-neighbouring sub-processes	55
	3.2.1 Strategy to develop the algorithm for splitting a given linear algorithm on the basis of data communication among the non-neighbouring sub-processes	56
	3.2.2 Pass II (Algorithm to extract the segment from the source program)	57
	3.2.3 Procedure for extracting the segment from the source program	59
	3.2.4 Pass III (Algorithm to modify the name of the variables)	62
3.3	Handling the tree structured algorithms using the concept of data communication	63
	3.3.1 Strategy to split a given tree structured algorithm	65
	3.3.2 Pass II	66
	3.2.3 Pass III	71
3.4	Handling the structured network algorithms	72
3.5	Handling the loop algorithms	72
3.6	Processes to processor assignment and their configuration	73
	3.6.1 Definition and lemmas	73
3.7.	Calculation of maximum data arrival rate	81
3.8	Conclusion	82
CHAPTER 4:	VIRTUAL PIPELINING	83
4.1	Introduction	83
4.2	Virtual pipelining	84

		Page
	4.2.1 Description of the proposed system	86
	4.2.2 Switching network	88
	4.2.3 Binary n cube network	91
4.3	Testing the feasibility of 'binary n cube network' for the proposed system	92
4.4	Influence of delay involved in the above multistage inter-connection network on the proposed design	95
4.5	Control scheme for 'binary n cube network'	96
4.6	Memory overhead for controlling the inter-change boxes by table look up technique	98
4.7	Data manipulator network	99
	4.7.1 Control overhead to implement the required switching connection by data manipulator network	101
4.8	Design of data communication network after incorporating the processor switching scheme	102
	4.8.1 Operation of the system after connecting the data memory to each processor and to evaluate its performance	104
	4.8.2 Solution of 'delay problem' in inter-connection network	106
	4.8.3 Drawback of the above approach	107
4.9	Conclusion	108
CHAPTER 5:	CONCLUSION	109
APPENDIX A.1	ALGORITHM OF PASS II FOR SPLITTING A LINEAR ALGORITHM USING CONVENTIONAL PIPELINING	113
APPENDIX A.2	ALGORITHM FOR PATH NUMBERING	120

Page

APPENDIX A.3 ALGORITHM OF PASS II FOR SPLITTING A LINEAR ALGORITHM USING DATA COMMUNICATION PIPELINING CONCEPT 123

APPENDIX A.4 ALGORITHM OF PASS III FOR SPLITTING A LINEAR ALGORITHM USING DATA COMMUNICATION PIPELINING CONCEPT 131

APPENDIX A.5 ALGORITHM OF PASS II FOR SPLITTING A TREE STRUCTURED ALGORITHM USING DATA COMMUNICATION PIPELINING CONCEPT 133

A.5.1 Algorithm of module 1 133

A.5.2 Algorithm of module 2 137

A.5.3 Algorithm of module 3 142

APPENDIX S.1 151

APPENDIX S.2 162

APPENDIX S.3 172

REFERENCES 182

ooooo