



# FAULT TOLERANT COMPUTING

*The 1989 Joint Symposium on  
Fault-Tolerant Computing  
JULY 18-20, 1989*

*Chongqing University, Chongqing, China*

江苏工业学院图书馆  
藏书章



**International Academic Publishers**  
A Pergamon-CNPIEC Joint Venture

**Pergamon Press**  
OXFORD · NEW YORK · BEIJING · FRANKFURT  
SAO PAULO · SYDNEY · TOKYO · TORONTO



**DISTRIBUTORS:**

U.K.	Pergamon Press plc, Headington Hill Hall, Oxford OX3 0BW, England
U.S.A.	Pergamon Press, Inc., Maxwell House, Fairview Park, Elmsford, New York 10523, U.S.A.
FEDERAL REPUBLIC OF GERMANY	Pergamon Press GmbH, Hammerweg 6, D-6242 Kronberg, Federal Republic of Germany
BRAZIL	Pergamon Editora Ltda, Rua Eca de Queiros, 346, CEP 04011, Sao Paulo, Brazil
AUSTRALIA	Pergamon Press (Australia) Pty Ltd, PO Box 544, Potts Point, NSW 2011, Australia
JAPAN	Pergamon Press, 5th Floor, Matsuka Central Building, 1-7-1 Nishishinjuku, Shinjuku-ku, Tokyo 160, Japan
CANADA	Pergamon Press Canada Ltd, Suite No 271, 253 College Street, Toronto, Ontario, Canada M5T 1R5
PEOPLE'S REPUBLIC OF CHINA	International Academic Publishers, Xizhimenwai Dajie, Beijing Exhibition Center, Beijing, 100044, People's Republic of China

---

Copyright © 1989 International Academic Publishers  
(A Pergamon-CNPIEC Joint Venture)

*All Rights Reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means: electronic, electrostatic, magnetic tape, mechanical, photocopying, recording or otherwise, without permission in writing from the publishers.*

*The book has been photographically reproduced from the best available copy. The papers were not refereed but were reviewed for their technical contents. Editing was restricted to matters of format, general organization and retyping. The editors assume no responsibility for the accuracy, completeness or usefulness of the information disclosed in this volume.*

First edition 1989

**Fault Tolerant Computing**

ISBN 7-80003-063-6/TN-4  
ISBN 0-08-037882-X

International Academic Publishers  
Pergamon Press

## GENERAL CHAIRMEN'S MESSAGE

With the wide applications of the computers to different areas and the rapid growth of the computing speed and memory capacity, the computing reliability, or more generally dependability, becomes the bottle-neck of the development for computer technology which attracts many researchers and practicers in the world to pay their attentions to this field and thus the discussion and communication among the scientists are needed in order to push forward the research on this topic.

In 1982 at the banquet of IEEE FTCS-12 in Santa Monica, Prof. Yoshihiro Tohma from Japan, Dr. Se June Hong from South Korea and Prof. Tinghuai Chen from China put their hands together and made a toast for their planning joint symposium to be hosted in turn in countries and regions in Asia. But at that time this symposium was only a dream. Many years passed until the joint symposium was proposed again by the Japanese colleagues in 1987. After this message was carried back by Prof. Yinghua Min and was discussed in the China side, the decisions for such joint symposium were made both in China and in Japan. Today we are proudly standing here to announce that '89 Joint Fault-Tolerant Computing Symposium is openning. This is the crystallization of the wishes, efforts and hard work of many Chinese and Japanese people.

This meeting is the first of the series of Joint Symposia on Fault-Tolerant Computing, sponsored by the Technical Committee on Fault-Tolerant Computing, Chinese Computer Federation and the Technical Committee on Fault-Tolerant Systems, IEICE. The first symposium is hosted by Chongqing University. We think, after many years it will be cherished that this symposium is a key milestone of the development on fault tolerance in the area of Asia-Pacific.

Since China is still a country in development, there are many difficulties in hosting an international conference, therefore we are extremely thankful to the program chairmen: Prof. Yinghua Min and Prof. Hideo Fujiwara for their active efforts and hard work under such difficult conditions. Sincere thanks are due to the referees who review the papers carefully, which guarantees the quality of the symposium. We would like to give thanks to all authors coming from Canada, India, Italy, Poland and USA through a long distance trip and also to all authors and participants for their outstanding work and warm concerns which make the symposium to be beneficial technically, professionally and socially.

We hope that everybody will have a nice meeting and a nice stay in Chongqing.

### General Co-Chairmen

Tinghuai Chen  
Chongqing University  
China



Yoshiaki Koga  
National Defense Academy  
Japan



## PROGRAM CHAIRMEN'S MESSAGE

There is a large number of applications for which computer failure is unacceptable. Even in less critical applications we have come to depend more on shared computing services, making dependability increasingly important. It is true in Japan. It is also becoming true in China because of the rapid increase in the use of computers in the recent years.

The two program chairmen recognized that a joint symposium on fault-tolerant computing would be valuable and significant for Chinese and Japanese scientists and engineers, and also be beneficial to developments in the state of the art. This suggestion was transferred to TC FTC of CCF by Yinghua Min who was invited to attend the 17th Japan FTC workshop in 1987, and transferred to TC FTS of IEICE as well, and then the decision of '89 JFTCS was made. The preparation work ran smoothly under the leadership of the general chairmen since then.

The digest you have in your hand is the first of the joint symposium on FTC. Its contents were obtained by the process of selection of program chair and committee, call for papers, the program committee meeting in Tokyo, 1988, distribution of papers to referees, evaluation of reports and paper selection at the program committee meetings separately on both sides, and final revision. Even though the '89 JFTCS is the first China-Japan joint symposium for the Chinese and Japanese scientists and engineers to present and discuss state-of-the-art developments in fault-tolerant computing and their applications, the symposium welcomes presentation and participation of people from elsewhere all over the world. The '89 JFTCS accepts 49 papers from China, Japan, India, Italy, Poland, Canada and the United States. Selection of the papers was based on originality, relevance to the symposium, and geographical distribution of the authors to encourage more people to participate in these technical interactions. Due to the limitation of time and space, many good papers unfortunately could not be included in the program.

The 49 papers deal with not only traditional aspects but also some new techniques, such as fault-tolerance in parallel computing, techniques for tolerating software and hardware design faults as well as operator errors, and fault-tolerance techniques in VLSI.

As an introduction to recent developments in research and application of fault-tolerant computing in China, a panel session, "FTC in China", is included in the program. Some experts are invited to present their work to reflect the developments of FTC in China.

We cannot sufficiently thank the many referees and program committee members for their tremendous contributions. Many of them were asked to read a large number of papers in a very short period of time. For those whose papers were not accepted, please accept our sincere regrets that the reviewing cycle does not allow iterations that in many cases would surely have resulted in acceptance.

Finally, we thank Prof. Kozo Kinoshita, Chair of TC FTS of IEICE, and Prof. Wei Daozheng, Chair of TC FTC of CCF, for their steady support and encouragement, and Prof. Tinghuai Chen, and Yoshiaki Koga, the general chairmen of '89 JFTCS for their cooperation.

#### Program Co-Chairmen

Yinghua Min  
Academia Sinica  
China



Hideo Fujiwara  
Meiji University  
Japan



# THE 1989 JOINT SYMPOSIUM ON FAULT-TOLERANT COMPUTING

Sponsored by  
Technical Committee on Fault-Tolerant Computing,  
Chinese Computer Federation,  
China

Co-sponsored by  
Technical Committee on Fault-Tolerant Systems  
Institute of Electronics, Information and Communication Engineers, JAPAN

Hosted by  
Chongqing University, China

Under the Auspices of:  
National Natural Science Foundation of China  
STRATUS Computer (HK) Limited  
Nan-Feng Computer Company, Kaili, Guizhou, China

## '89 JFTCS SYMPOSIUM COMMITTEE

### **General Chairman**

Tinghuai Chen  
Computer Research Institute  
Chongqing University  
Chongqing, Sichuan  
China 630044

### **Program Chairman**

Yinghua Min  
Center for  
Fault-Tolerant Computing  
CAD Laboratory  
Institute of Computing Technology  
Academia Sinica,  
Beijing, China 100080

### **General Co-Chairman**

Yoshiaki Koga  
Department of Electrical Engineering  
National Defence Academy  
1-10-20 Hayamizu  
Yokosuka 239, Japan

### **Program Co-Chairman**

Hideo Fujiwara  
Department of Computer  
Science  
Meiji University, Japan  
1-1-1 Higashi-mita, Tama-ku  
Kawasaki 214, Japan



## PROGRAM COMMITTEE MEMBERS

Yongzhang Chen  
Computer Research Institute  
Chongqing University  
Chongqing, Sichuan  
China 630044

Eiji Fujiwara  
Department of Computer Science  
Tokyo Institute of Technology  
2-12-1 O-okayama, Meguro-ku  
Tokyo 152, Japan

Kiyosi Furuya  
Department of Electrical  
and Electronic Engineering  
Chuo University  
1-13-27 Bunkyo-ku  
Tokyo 112, Japan

Bidyut Gupta  
Computer Science Unit  
Indian Statistical Institute  
203 B.T. Road  
Calcutta, 700035  
India

Jiandong Lu  
Department of Computer  
Science & Engineering  
Huazhong University of  
Science & Technology  
Wuhan, Hubei  
China 430074

Takashi Nanya  
Department of Computer Science  
Tokyo Institute of Technology  
2-12-1 O-okayama, Meguro-ku  
Tokyo 152, Japan

Laung-Terng Wang  
Center for Reliable Computing  
Computer Systems Laboratory  
Stanford University  
Stanford, CA 94305-4055  
USA

Shiyi Xu  
Department of Computer Science  
Shanghai University of  
Science and Technology  
Shanghai, China 201800

Shigeru Yamada  
Department of Industrial  
and Systems Engineering  
Hiroshima University  
Higashi-Hiroshima-shi 724  
Japan

Teruhiko Yamada  
Department of Electronics  
and Communications  
Meiji University  
1-1-1 Higashi-mita, Tama-ku  
Kawasaki 214, Japan

Xiaozong Yang  
Department of Computer  
Engineering  
Harbin Institute of Technology  
Harbin, China

Youguang Yuan  
Wuhan Digital Engineering  
Institute  
Wuhan, China

Yixin Zhu  
Shenzhen Science & Industry Park  
Corporation  
Shenzhen, China

## '89 JFTCS REFEREES

CHEN, Junliang  
CHEN, Yongzhang  
CHEN, Zhaolong  
DENG, Yaping  
DONG, Hao  
FAN, Chengrong  
FANG, Binxing  
FUJIWARA, Eiji  
FUJIWARA, Hideo  
FURUYA, Kiyoshi  
HASHIZUME, Masaki  
HU, Mou  
HUANG, He  
HUANG, Kaiyuan  
HUANG, Weikang  
ITOH, Hideo  
KINOSHITA, Kozo  
KOGA, Yoshiaki  
LI, Jintao  
LI, Meng  
LI, Xiangyu  
LIANG, Yewei  
LING, Xieting  
LIU, Jiasong  
LU, Jiandong  
MIN, Yinghua  
NAITO, Sachio  
NAKAMICHI, Matsuroh  
NANYA, Takashi  
TAKAMATSU, Yuzo

PAN, Yuqi  
SHEN, Li  
SHENG, Yunhuan  
TAMAMOTO, Hideo  
TAN, Ying  
TAN, Zhongping  
TONG, Changzhong  
WANG, Lei  
WANG, Yingxu  
WEI, Daozheng  
XU, Shiyi  
XUE, Shijun  
YAMADA, Shiyeru  
YAMADA, Teruhiko  
YANG, Shiyuan  
YANG, Xiaozong  
YANG, Ying-Wa  
YUAN, Youguang  
ZENG, Shide  
ZENG, Wenbin  
ZHANG, Hanying  
ZHANG, Huanguo  
ZHANG, Jie  
ZHANG, Kenong  
ZHAO, Guonan  
ZHENG, Chongxun  
ZHENG, Jun  
ZHU, Ronghui  
ZHU, Yixin

# Table of Contents

<b>General Chairmen's Message</b> .....	iii
<b>Program Chairmen's Message</b> .....	v
<b>'89 JFTCS Symposium Committee</b> .....	ix
<b>Program Committee Members</b> .....	xi
<b>'89 JFTCS List of Referees</b> .....	xiii
 <b>Session 1A: Fault-Tolerant Hardware (I)</b>	
Chair: Yang Xiaozong, Harbin Institute of Technology, China	
On Multiplication of Processors for Fault-Tolerance .....	2
<i>Kiyosi Furuya</i>	
The Researches in Fault Tolerant Data Flow Architecture .....	8
<i>Wang Lei, Tan Ying and Hang Wen Yu</i>	
A Fast Design for Fault Tolerant Hardware Clock Synchronisation .....	12
<i>Sandeep Pagey, G Venkatesh and S.S.S.P. Rao</i>	
 <b>Session 1B: Fault Simulation and Modelling</b>	
Chair: Hideo Fujiwara, Meiji University, Japan	
Algorithmic TPG Oriented Fault Simulator: ALOFAS .....	20
<i>Yuzo Takamatsu and Kozo Kinoshita</i>	
A Unified Fault Model .....	26
<i>Liang Yewei</i>	
Critical Path Tracing .....	32
<i>Luo Dongjin and Wei Daozheng</i>	
 <b>Session 2A: Fault-Tolerant Hardware (II)</b>	
Chair: Yoshiaki Koga, National Defence Academy, Japan	
DSMS-Fault-Tolerance for Dual STD Microcomputer System .....	38
<i>Lu Jiandong, Li Jianlin and Huang Changhua</i>	
On the Minimum Repair Cost for VLSI/WSI Memory Chips with Redundancy .....	43
<i>Wei-kang Huang and F. Lombardi</i>	
A Unified Model and the Optimal Design of Fault-Tolerance for Multiprocessor Network .....	50
<i>Rong Yao, Tai Kang and Tinghuai Chen</i>	
 <b>Session 2B: Fault Detection (I)</b>	
Chair: Wei Daozheng, Academia Sinica, China	
A New Method for Analog Fault Diagnosis with Tolerance .....	58
<i>Yan Li and Shibai Tong</i>	
A Novel Approach to Detection of ROM with High Fault Coverage .....	63
<i>Shiyi Xu and Jianhua Gao</i>	
Diagnostic Testing of NAND Logic Circuits under Totally Observable Condition .....	69
<i>Kozo Kinoshita, Wen Xiaoqing, and Sudhakar M. Reddy</i>	

### **Session 3A: Fault Tolerant Software (I)**

Chair: Kiyoshi Furuya, Chuo University, Japan

Dynamic Redundancy at the Task Level .....	76
<i>Li Weihua and Yuan Youguang</i>	
Reverse Computing—An Engineering Approach to Software Fault-Tolerance .....	80
<i>Chen Junliang</i>	
The Design and Implementation of A General Multi-Processor System with Fault Tolerant Capability .....	85
<i>Yang Xiaozong, Fang Binxing, Wang Tianzuo and Hu Mingzeng</i>	

### **Session 3B: Fault Detection (II)**

Chair: Yuzo Takamatsu, Ehime University, Japan

A Parallel Algorithm for Seeking the Set of the Shortest Test Sequences (STS) .....	92
<i>Wang Wenzhang, Tian Shaohuai and Wang Ronghua</i>	
Logic Probability Calculation and Applications .....	99
<i>Zheng Chongxun and Zhang Kenong</i>	
Test Generation for Sequential Circuits Using Structured Test Table .....	104
<i>Matsuroh Nakamichi and Arthit Thongtak</i>	

### **Session 4A: Fault-Tolerant Software (II)**

Chair: Chen Junliang, Beijing University of Posts &  
Telecommunications, China

An Intelligent Fault-Tolerant Method for Computer Message Process Systems .....	112
<i>Guangwei Chen and Cunyi Zhao</i>	
Availability Measurement Based on Software Reliability Growth Model .....	117
<i>Hiroshi Ohtera, Shigeru Yamada and Hiroyuki Narihisa</i>	
N-Version Programming and Its Implementation .....	123
<i>Gao Yuan and Liu Guichun</i>	

### **Session 4B: Test Generation**

Chair: Kozo Kinoshita, Hiroshima University, Japan

Analysis of Parallel Processing for Test Generation in a Distributed System .....	128
<i>Hideo Fujiwara and Tomoo Inoue</i>	
A Product-Oriented Strategy of Test Pattern Generation for Programmable Logic Arrays .....	134
<i>Jintao Li and Yinghua Min</i>	
Minimum Test-Sequence for Delay and Stuck-Open Fault Detection of a Fanout-Free 2-level Circuit .....	144
<i>Hideo Itoh</i>	

### **Session 5A: Distributed Systems**

Chair: Takashi Nanya, Tokyo Institute of Technology, Japan

A Support for Robust Replication in a Distributed Object Environment .....	152
<i>A. Corradi, L. Leonardi</i>	
The DFTSNA Distributed Fault-Tolerant System: Architecture, Fundamental Principle and Implementation .....	159
<i>Jie Xu, Qingnan Li, Shize Huang and Jiangfeng Xu</i>	
Measurement and Modeling of Computer System Failures .....	163
<i>Ravishankar K. Iyer and Dong Tang</i>	

**Session 5B: Design Verification**

Chair: Shiyi Xu, Shanghai University of Science and Technology,  
China

An ETG PLA Designer .....	166
<i>Yinghua Min</i>	
Hazard Analysis under Time-Ordered Input Transitions .....	172
<i>S.D. Sherlekar and P.S. Subramanian</i>	
Logic Simulator for Microprocessor Based Circuits Using Timing Charts .....	178
<i>Masaki Hashizume, Toshinori Inoshita and Takeomi Tamesada</i>	

**Session 6A: Network**

Chair: Sachio Naito, Nagaoka University of Technology, Japan

An Approach for Single Fault Detection and Location in Baseline Interconnection Networks with a Constant Number of Tests .....	186
<i>Wei-kang Huang and F. Lombardi</i>	
An Optimal Fault-Tolerant Routing Algorithm for Dual-Ring Networks .....	192
<i>Binxing Fang, Xiaoming Li and Wenmin Wang</i>	
Dynamic-Full-Access Fault-Tolerance in Extra Stage Cube Interconnection Network .....	196
<i>Yaping Deng</i>	

**Session 6B: Built-in Self-Testing**

Chair: Zhang Hanying, Beijing Institute of Control Engineering,  
China

An LFSR-Based Built-In Self-Test Scheme for Embedded RAMs .....	204
<i>Teruhiko Yamada and Hiroshi Nakajima</i>	
On the Equivalence Between IE-TYPE and EE-TYPE Signature Analyzers .....	210
<i>Han Shuliang and Chen Zhaolong</i>	
The Optimal Input Probability Distribution in Random Testing .....	216
<i>Xiangdong and Zhangzuan</i>	

**Panel: Fault-Tolerant Computing in China**

Chair: Yinghua Min, Academia Sinica, China

**Session 7A: Computer Control**

Chair: Lu Jiandong, Huazhong University of Science &  
Technology, China

A Bus Comparator for a Fail-Safe Dual-Microcomputer System .....	222
<i>Yuan Chunxin and Zhou Zhibang</i>	
The Fault-Tolerant Design of Microprocessors in the Control System of Railway Cab Signal .....	227
<i>Ning Bin and Wang Xishi</i>	
A Test Algorithm for SLCs in Digital Switching Systems .....	232
<i>Yingxu Wang</i>	

**Session 7B: Self-Checking**

Chair: Ying-Wa Yang, Shaanxi Microelectronics Research, China

A Self-Checking Restoring Device for QMR Systems .....	238
<i>Hiroyuki Mori, Hiroki Sawada, and Yoshiaki Koga</i>	
The Design of Strongly Fault-Secure and Strongly Code-Disjoint Combinational Circuits .....	245
<i>Takashi Nanya and Masatoshi Uchida</i>	
Unequal Error Protection Codes for Fault-Tolerant Computer Systems .....	251
<i>Chen Zhi, Fan Pingzhi and Jin Fan</i>	



**Session 8A: System Diagnosis**

Chair: Kaiyuan Huang, Chongqing University, China

Characterization of Connection Assignment of t-Diagnosable Systems with Repair .....	256
<i>Shize Huang, Jie Xu and Jiangfeng Xu</i>	
Distributed Diagnosis Strategy in Hybrid Fault Situations .....	260
<i>Wojciech E. Kozłowski and Henryk Krawczyk</i>	
System Diagnosis with Given Syndrome .....	267
<i>Tinghuai Chen, Hengchang Guo and Tai Kang</i>	

**Session 8B: Retry, Recovery and Reconfiguration**

Chair: Yongzhang Chen, Chongqing University, China

A Hierarchical Reconfiguration Strategy for Bus-Based Multiprocessors .....	274
<i>Ashish Pancholy, Fidel Muradali and Vinod K. Agarwal</i>	
A Cache-Aided Multiprocessor Rollback Recovery Scheme .....	280
<i>Kun-Lung Wu and W. Kent Fuchs</i>	
Combinational Circuits for Complement Data Retry and Alternate Data Set for Multiple Faults .....	286
<i>Sachio Naito</i>	
A New Fault-Tolerant Method for Memory .....	292
<i>Wang Shu-hua, Wang Dong-sheng and Wen Yong-cheng</i>	

**Late Paper**

Measurement and Modeling of Computer System Failures .....	298
<i>Ravishankar K. Iyer and Dong Tang</i>	
Author Index .....	305

**Session 1 A**

**Fault-Tolerant Hardware ( I )**

**Chair: Yang Xiaozong**

**Harbin Institute of Technology, China**

# On Multiplication of Processors for Fault-Tolerance

Kiyoshi FURUYA

Faculty of Sci. & Eng., Chuo University.  
Bunkyo-ku 1-13-27, Tokyo, 112 Japan.

**ABSTRACT** The paper discusses the strategies of constituting fault-tolerant systems due to multiplication-and-voting. We show comparisons of two schemes, one based on hardware and the other based on software. Problems associated with synchronization and voting that are critical in fault-tolerant computing systems based on software are further investigated. Then, an interprocessor communication scheme employing small amount of memories is proposed.

## 1. Introduction

Although various fault-tolerant computing systems have ever been designed [1],[2], characteristics and problems associated with those principles must be investigated to make clear the general design methodologies and the way of estimating reliabilities.

Methods of constructing fault-tolerant systems by the multiplication-and-voting fall into two classes depending on the way of realization: one based on hardware and the other based on software. We first show comparisons of those two schemes on various aspects. Synchronization and voting are realized by interprocessor communications in the scheme based on software. We investigate problems associated with the communication, making clear the conditions imposed on system configurations. Finally, we discuss various communication schemes on the performance and the hardware requirement, and propose a scheme in which small amount of memories are used.

## 2. Multiplication scheme based on Hardware and That based on Software

To constitute a computing system by multiplication-and-voting, required functions for providing fault-tolerance could be realized either based on hardware or on software. The typical examples of the actual systems are FTMP [3] and SIFT [4],[5], respectively.

The multiplication scheme based on hardware is realized by appending voters for corresponding signal lines of multiplied processors or computing systems. In most actual fault-tolerant systems, however, voters are also multiplied in the form of Triplicated-

TMR to eliminate hard cores.

In the scheme based on software, on the other hand, voting is realized by confronting results of each processor with those of other two (or more) processors. Therefore, an application program must be divided into appropriate length of tasks, and the results of a task have to be justified before proceeding to the next task. (It is possible to postpone until the results are actually referred.) If an inconsistency is detected, it will be recorded in an error table. Then, after some repetitions of tasks and voting phases, system diagnosis along with system reconfiguration are performed referring the error records. The typical scheduling of the operation is shown in Figure 1. Processors have to be synchronized, possibly by exchanging flags values, every after tasks and voting phases to make sure that all active processors have completed them and to depress the skew within a predetermined range. Synchronizations are also required after system diagnoses and reconfigurations.

We now show comparisons on the various features of multiplication scheme based on hardware and that based on software. Discussions are summarized in Table 1.

**Basic Configurations:** In the scheme based on hardware, processors and memories can be multiplied respectively. On the other hand, in the scheme based on software, a processor requires a dedicated memory. Faults in a processor and the dedicated memory are indistinguishable. Therefore, they have to be multiplied intrinsically in one united body. Fault isolation in software based system can be

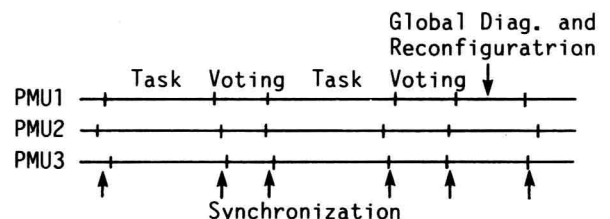


Figure 1. An Operation of a Triplicated System based on Software Voting.

achieved by prohibiting write operation of a dedicated memory from other processors.

**Synchronization:** In order to supply signals to a hardware voter simultaneously, processors must be synchronized tightly in clock level. Since the reliability of a common clock generator give direct inference on that of the whole system, its fault-tolerance is essential. It is usually achieved by multiplying clock generators and interconnecting them with a phase locked loop. However, such synchronization is turning to be more and more difficult as the speed of processors increases.

In the scheme based on software, on the other hand, loose synchronization in the software level allowing with a certain range of skew is sufficient.

**Hardware Overhead:** The scheme based on hardware requires a hardware voter for each bit of signals that should be checked. Consequently, voters account for a major portion of hardware overhead. The absence of testability of voting elements often incurs a problem. Further, such fault isolation mechanisms like bus-isolation gates of the FTMP are also required in most systems. In stead of the hardware voters, the scheme based on software requires inter-processor communication mechanism. As discussed more detail in latter section, its effective realization is a crucial issue.

**Overhead in Time Domain:** In the scheme based on the hardware, the operational delay of voters will cause the overhead in time domain. System reconfigurations may also incur the overhead. In the scheme based on software, on the other hand, the most serious overhead in time domain comes from the interprocessor communication.

**Reconfiguration Scheme:** In the scheme based on hardware, the system reconfiguration is achieved physically by precluding faulty units and including spare ones. On the other hand, in the scheme based on software, it can be achieved logically. However, if a faulty processor is left operational, it may disturb other active processors by absurdly accessing to the communication mechanism. Therefore, it is better to preclude faulty units not only logically but also physically as soon as possible.

**Restrictions on Software:** In the scheme based on hardware, no restriction is imposed principally on software, since voting operation is applied for predetermined signal lines. However, when only a particular set of signals are voted, application programs must be designed so as to transmit critical data to the voted signal lines. Actually, in the FTMP system, processors have dedicated cache

memories in which most operation are carried out, and voting is applied only for data on the redundant buses connecting processor units and main memory units. Therefore, data to be voted must be transferred to or from main memory units.

In the scheme based on software, application programs have to be divided into appropriate length of tasks to take vote on the results of computations. If the task is too long, system reliability will be decreased due to an occurrence of intolerable faults. On the contrary, if the task is too short, it will suffer heavy overhead in time domain from synchronization and voting. Thus, the length of task must be determined appropriately regarding the trade-off between reliability and overhead in time domain.

If an application program is divided into tasks at arbitrary points, large amount of data including results of computation and program status words will be required to be voted. So that, an application programs should be divided at appropriately in order to increase the throughput. For example, for a program which repeats similar routines, it is quite natural to assign a repetition to a task. A relatively short off-line program could be assigned to a task. However, the appropriate partitioning of an arbitrary program is a difficult problem. The similar problem can be found in setting

Table 1. Comparisons of Multiplication Schemes, one based on Hardware and the other based on Software.

	Mechanisms for Providing Fault-Tolerance	
	Hardware	Software
Processor and Memory	Multiplicated in One United Body or Respectively	Multiplicated in One United Body
Synchronization	Clock Level	Software Level
Additional H/W Overhead	Voters and Fault Isolation Mechanism	Communication Mechanism
Overhead in Time Domain	Slight	Crucial
Reconfiguration	Physical	Physical or Logical
Restrictions on Software	Slight	Programs must be divided into Tasks.
Correlated Err.	Tend to Suffer	Hard to Suffer
N-version Programming	Incompatible	Compatible
Analog Signal Processing	A common quantized value must be used.	Flexible
Lying Problem	Serious	More Serious
Actual Systems	FTMP	SIFT