Proceedings of The Korea-Russia Joint Conference on
Advanced Sciences and Technologies. *June, 1998, Korea*

An I/O Bus-Based Dual Active Fault Tolerant Architecture for Good System Performance

Seung-Uk kwak*, Jeong-Il Kim*, Keun-Won Jeong*, Kyong-Bae Park**,
Kyong-In Kang***, Hyen-Uk Kim*, Kwang-Bae Lee*
* Dept. of Electronic Engineering, Myong-Ji University
** Dept. of Computer Science, Yeo-Joo Institute of Technology
*** Dept. of Information and Communication, Yeo-Joo Institute of Technology

Mailing Address : Prof. Kwang-Bae Lee
Dept. of Electronic Engineering, Myong-Ji University
San 38-2, Nam-Dong, Yongin City, Kyenggi-Do, 449-728, Korea
E-mail: kblee@wh.myongji.ac.kr

Abstract

In this paper, we propose a new fault tolerant architecture for high availability systems, where for module internal operations both processor modules perform the same tasks at the same time independently of each other while for module external operations both processor modules act actively. That is, operations of synchronization between dual processor modules except clock synchronization are requested only when module external operations are executed. The architecture can not only improve system availability by reducing system reintegration time but also reduce performance degradation problem due to frequent synchronization between dual processor modules. The clock unit consists of a clock generator and a clock synchronization circuit. This supplies a stable clock signal under clock unit disorder of any processor module or rapid clock signal variation. And this architecture achieves system availability and data credibility by designing as symmetrical form.

I. Introduction

Today, fault tolerant systems have been widely used in several areas such as defense system, banking system and telephone system, and so on.[1]-[3]

Modern society has required high quality of customer service, increasing need for high availability systems which remain alive as long as possible when a system fails. System availability and data credibility will become more important factors with the advance of information society.

Most high availability systems have been implemented in the basis of standby sparing architectures: warm standby sparing and hot standby sparing. Warm standby sparing architecture has been widely used for high availability systems because its implementation is relatively easy. This architecture, however, has possibility of losing data on fault occurrence and spreading false data on master changeover. In addition, frequent recovery block creation procedures under normal state to reduce data loss can drop the system performance significantly.

On the other hand, hot standby sparing architecture has the following advantages: rapid changeover from normal mode to spare mode on fault detection and protection from spreading of undetected false data over the whole system. However, this architecture requires frequent synchronization between the processor modules to maintain the same operation at the same time. Eventually, it dramatically increases system

overhead and leads to poor system performance. A new fault tolerant architecture to reduce such a severe performance problem is implemented in a commercial fault tolerant computer system ft-SPARC.[5] For the system, synchronization and data comparison between the processor modules is carried out only on I/O operations so that the system can avoid frequent synchronization tasks and improve its performance. Transient processor error can cause data incoherency problem or system malfunction for the dual processor module architecture because the error can be detected during only output operation. In addition, error master clock will make the system down ultimately.

In this paper, we present an I/O bus-based fault tolerant architecture using dual processor modules which is based on hot standby sparing architecture.

II. Symmetrical Processor Module

1. System Architecture

Dual processor module architectures have been widely accepted for commercial high availability system because they can provide high performance and high reliability at a low cost. As a fault tolerant architecture to meet such requirements, we propose a dual processor module architecture using I/O bus shown in fig 1.

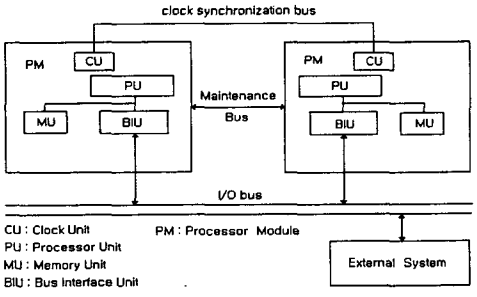

fig.1. Architecture of dual active processor modules

In the dual processor module architecture, both processor modules act as active state for normal mode. The only difference between both processor modules only any one module loads data on I/O bus when a module output operation is executed. It simplifies system function, thus improving system performance and system availability. A clock synchronization bus exists between the dual processor modules. Each processor module uses a synchronized clock signal that is made by clock signal from its own clock generator and clock signal transferred from the other processor module through the clock synchronization bus. It prevents the system-down that could occur if the architecture would use a common clock signal produced from a master clock generator and the master clock generator would be out of order. A maintenance bus between the dual processor modules is used for informing errors detected in a processor module and transferring control information required for system recovery after error detection. The maintenance bus is also used to support memory concurrent writes for system recovery.[4]

Each processor module consists of a processor unit, a local memory unit, a bus interface unit, a clock unit and a maintenance bus control unit. The processor unit has capability to detect transient processor error immediately. The bus interface unit includes DMA capability, data comparison logic, and parity checking logic. The data comparison logic is used to detect fault on module output operations while the parity checking logic is used on module input operation. The clock unit consists of a clock generator and a clock synchronization circuit.

2. System Operation

All processor modules perform the same operations at the same time with synchronized clock signal. For processor instructions, each processor module executes the instructions independently without any comparison between processor modules at full processor/memory bandwidth. Comparison between processor modules is done for only output instructions. It

makes the architecture achieve an enormous performance gain over other hot standby sparing architectures. When the system executes an I/O instruction, all processor modules have to perform the same input or output operation at the same time. If the processor modules request different operations at I/O access or fail to make any I/O operation, it means a fault occurs in the system. The fault detection will lead to system diagnosis and recovery activities. The cause of this fault is mainly some failure within the processor module element such as memory corruption or bus interface failure. The memory corruption can be fatal for dual processor module architectures because in most cases the system can not discern good module and failed module. The memory corruption can be greatly reduced by using the processor unit that can immediately detect transient processor errors. Delay error between the processor modules can be also detected at I/O access.

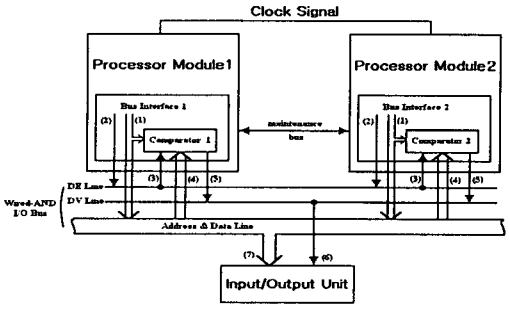

fig.2. Block diagram of output operation

In the case of a module output operation, both processor modules loads its data and asserts DE(data enable) signal on I/O bus. When the DE line is asserted, comparison between data from I/O bus and data from its own processor module is performed in each processor module. If there is no disagreement between two processor modules, which is the normal case, each processor module asserts DV(data valid) signal on I/O bus. The DV signals from two processor modules are wired-ANDed on I/O bus. It asserts DV line of I/O bus and the output processing continues.

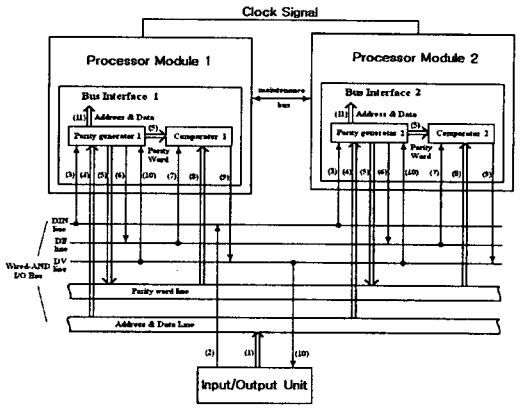

fig.3. Block diagram of input operation

In the case of a module input operation, each processor module reads data from I/O bus and generates parity word from the data. Then, it compares the generated parity word with the transferred parity word. If there is no error, each processor module asserts DE signal on I/O bus. The DE signals from two processor modules are wired-ANDed on I/O bus. When the DE line of I/O bus is asserted, each processor module transfers the data to its memory unit at the same time. If a processor module asserts DE signal but does not detect assertion of DE line within a predefined time, the processor module informs the other processor module of the fact by using the maintenance bus.

3. System Recovery

In the case that a processor module detects a fault for module internal operations, the processor module informs the other processor module of the fact. Then, the good processor module takes the failed processor module off-line and continues its suspended task in a single mode while the failed processor module is tested for permanent failures.

After a processor module failure is removed by means of hardware repair or replacement, the contents of the operating processor module's memory are copied onto the recovered processor module's memory in order for two processor modules to hold exactly the same memory

contents. The memory copy takes place without disturbing the normal operation in the system. Memory contents modified by the operating processor module during the copy are recopied iteratively. The iterative recopy process should be stopped when only small amount of memory contents modified remain. At that time, the system completes its copy without any interruption. When the copy is complete, the operating processor module informs the other processor module of the fact. Then, both processor modules are halted for a while and restarted in synchronization.

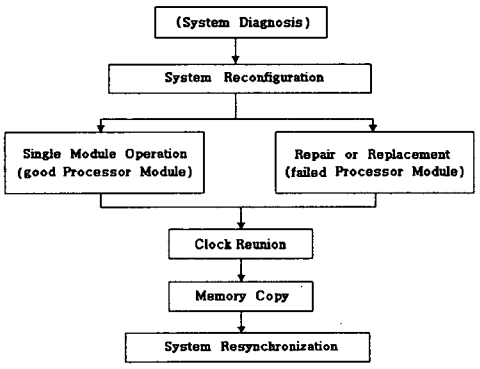

fig.4. State diagram of recovery operation

III. Unavailability Evaluation

We use a continuos-time Markov model to compute system unavailability of the proposed architecture and consider only the steady-state solution here. For simple evaluation, we do not consider any software errors and make some following assumptions.

1) There does not exist any undetectable error in the system.
2) Processor and clock errors and 99.99% of power supply error are immediately detected on error occurrence.
3) Memory/bus interface and remaining power supply errors are detected on I/O operation.
4) Any error does not occur when the system is resynchronized.
5) Module delay error does not exist.
6) When the system is down due to errors in both processor modules, the system recovers both processor modules simultaneously and then returns to the normal state.

A Markov model for our architecture based on the above assumptions is shown in figure 5.

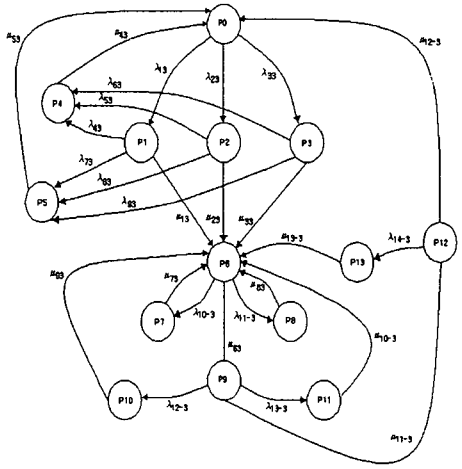

fig.5. Continuous-time Markov model for the proposed architecture

We use Lambda output data for SUN-2/50 workstation in [2] to extract approximate failure rate values. Power failure ratio shown in figure 6. represents ratio of power failure rate over total system failure rate including power failure rate. Each state is following.[5]

P0 : a normal state
P1 : a state where a failure has been detected on non-I/O operation(except clock error)
P2 : a state where an error has been detected on I/O operation
P3 : a state where clock error has been detected
P4 : a state where an error has been immediately detected in a normal module during transformation as a single module
P5 : a state where an error has been detected by frequent checking in a normal module during transformation as a single module

P6 : a state where is operated as a single module and is performed hardware repair and replacement for processor module failure

P7 : a state where a failure has been immediately detected in a single module

P8 : a state where an error has been detected by only frequent checking

P9 : a state where data recovery is performed and hardware repair is completed for processor module failure

P10 : a state where an error has been immediately detected in a single module during data recovery

P11 : a state where an error has been detected by only frequently checking in a single module during data recovery

P12 : a state where data recovery is completely and system reconfiguration is performed

P13 : a state where an error has been detected during system reconfiguration

We can solve for the system unavailability by computing P1 + P2 + P3 + P4 + P5 + P7 + P8 + P9 + P10 + P13

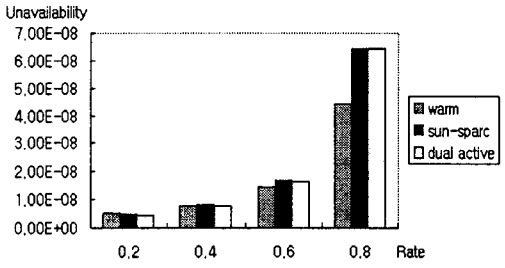

fig.6. System unavailability over power failure ratio

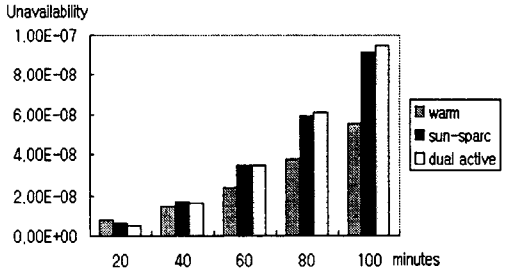

fig.7. System unavailability as a function of module replacement time

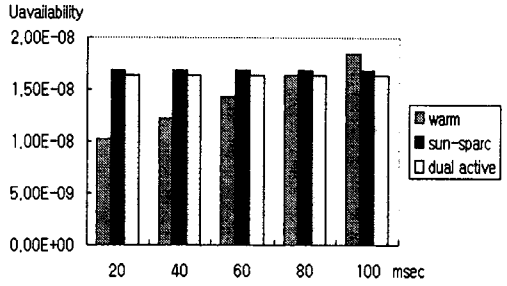

fig.8. System unavailability as a function of hardware repair time

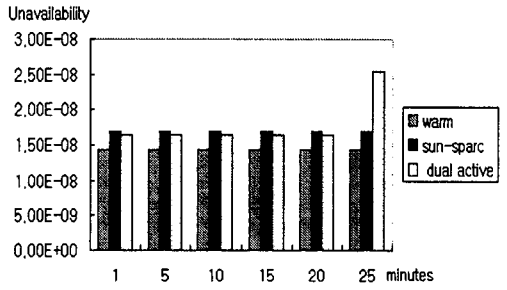

fig.9. System unavailability as a function of data recovery time

IV. Conclusions

Conventional high availability systems based on dual processor modules do not sufficiently satisfy such a social requirement because they have some problems in performance, data credibility or availability. We, hence, present a fault tolerant architecture using dual processor modules to reduce such problems. The dual processor modules perform the same operation at the same time independently of each other except when I/O instructions are executed. Because this architecture performs comparison between two processor modules through I/O bus for only module output operations, that can minimize degradation by frequently synchronization between dual processor modules. It provides performance gain over most hot standby sparing architectures by using a new method to reduce system reconfiguration time. In addition, it improves

system availability because system diagnostics time can be greatly reduced. Also, we could remove non-synchronization problem by using system clock to average clock signal generated by each clock unit. However, use of such processor unit will increase the system hardware cost.

References

[1] B. W. Johnson, *"Design and Analysis of Fault -Tolerant Digital Systems"*, Addison Wesley, 1989.

[2] D. P. Siewiorek and R. S. Swarz, *"Reliable Computer Systems"*, Digital Press, 1992.

[3] J-C Laprie et al., *"Definition and Analysis of Hardware and Software Fault-Tolerant Architectures"*, Computer, July 1990.

[4] Roland E. Best,*"Phase-Locked Loop"*, McGraw Hill, Third Edition, 1997

[5] ft-SPARC Technical Description, Integrated Micro Product Ltd., 1995.