Fault Side-Effects in Fault Tolerant Multistage Interconnection Networks

T. Schwederski E. Bernath G. Roos
schwederski@mikroelektronik.uni-stuttgart.dbp.de
Institute for Microelectronics Stuttgart
Allmandring 30
W-7000 Stuttgart 80
F.R. Germany

W. G. Nation H. J. Siegel
hj@ecn.purdue.edu
Parallel Processing Laboratory
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-0501, USA

Abstract

Many fault tolerant (FT) multistage interconnection networks have been devised that tolerate one or more faults. It is shown that the effects of some realistic faults can propagate through multiple stages and cause non-faulty components to behave erroneously; such faults with *side effects* can cause "FT" networks to lose their fault tolerance capabilities even due to a single fault. The PASM prototype interconnection network is examined as a case study of realistic faults with side effects. A fault model that includes side effects is introduced. An enhanced Extra Stage Cube network is proposed that tolerates faults with or without side effects.

1. Introduction

Design of fault tolerant networks for parallel machines has been widely considered. The *multistage interconnection networks* are one important class, and many fault tolerant networks of this class have been devised [1]. Generally, multistage interconnection networks consist of two or more *stages* of *switches* that connect sources and destinations, e.g., [2,3]. *Fault tolerant (FT) networks* enhance the basic structure by adding redundancy to bypass faulty links and/or switches e.g., [1,4-7].

The fault models that are employed in these networks either assume that a faulty component cannot be used at all, or they assume a partial failure (e.g., a switch sends the correct data to the wrong destination) [1]. In all of these fault models, the faults are local and have no effect on non-faulty links or stages. For many realistic failures, this assumption is not valid. For example, in a study on fault identification of non-fault-tolerant multistage cube networks, Davis et al. [8] introduce a more general fault model that also includes stuck paths that might span the network. Thus, a single fault may have *fault side effects (FSEs)* on network components without hardware faults.

In many cases, the impact of such side effects will be detrimental. Stuck paths, for example, may prohibit processors from sending to or receiving from the network. This will be shown in a study of an actual FT network with realistic faults. To examine FSEs, a general model is presented, and the properties of FSE-tolerant networks are discussed. FT networks may fail in the presence of FSEs [9]; they can be modified by adding appropriate hardware such that they are FSE-tolerant. An enhanced version of the Extra Stage Cube FT network is proposed that achieves FSE-tolerance with little added hardware complexity. To illustrate the required overhead, the enhancements are discussed in the context of an actual prototype interconnection network.

This work was sponsored in part by the EUREKA research program PROMETHEUS, subprogram PROCHIP, by the Bundesministerium für Forschung und Technologie under contract TV 8926.3, and by the Office of Naval Research under grant number N00014-90-J-1483.

2. Case Study of the PASM ESC Network

The processing elements of the protoype of the PASM (Partitionable SIMD/MIMD machine) [10] parallel processing system system are connected by an Extra Stage Cube (*ESC*) network with 16 inputs and 16 outputs [1,11]. The ESC is designed to tolerate any single fault and is robust in the presence of multiple faults. Figure 1 shows an ESC network for N=8. The PASM prototype network is constructed from 2-by-2 interchange boxes and bypass switches; it is circuit switched, has a data path width of 16 bits with two parity bits, and is capable of broadcasts. Paths through the network are established and released by a path request (*PRQ*) / path grant (*PGRT*) mechanism. Sources and destinations will be referred to as PEs.

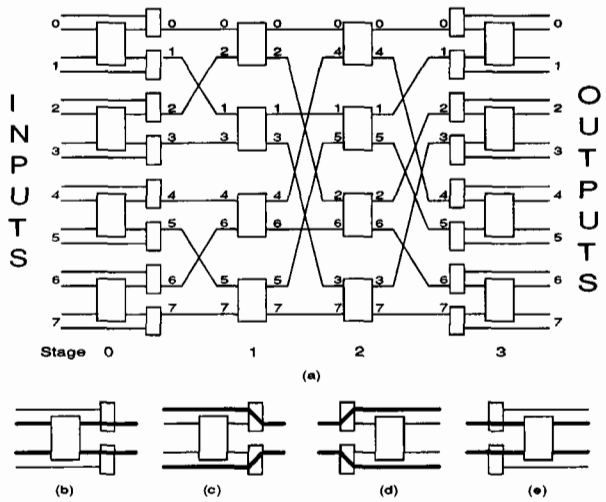

Figure 1: (a) Extra Stage Cube Network for N=8; (b) Input stage enabled; (c) Input stage disabled; (c) Output stage disabled; (e) Output stage enabled

Consider a stuck path. If any PRQ signal becomes stuck while in the asserted state, the path can no longer be dropped. Thus, at least one destination becomes permanently unreachable. Consider why the ESC cannot tolerate such a fault. During normal operation, the input stage is enabled and the output stage is disabled. To clarify the discussion, the ESC from Figure 1 is used as an example. Let the top box of stage 2 be faulty in such a way that it has established a path from its upper output (link 0) to network output 0, and can no longer drop this path. This is indicated in Figure 2a. While the output stage is enabled, PE 0 cannot be reached because the stuck path blocks all data transfers to PE 0 from both the upper and the lower input of the output stage box. Disabling the output stage does not alleviate the problem. When the output stage is disabled, PE

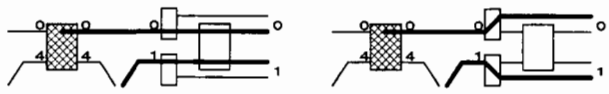

Figure 2: (a) Stuck path to PE 0 with bypass disabled; (b) Stuck path from PE 0 with bypass enabled

0 can only be reached from the top output of the faulty box that is blocked due to the stuck path (Figure 2b). Thus, PE 0 cannot be reached due to a single fault, and this single fault will therefore cause the ESC to lose its FT capabilities.

Stuck paths are an example of forward fault propagation, i.e., from the fault in the direction of the network outputs. Consider an example of backward propagation (i.e., from fault to network inputs). Assume that a PGRT signal is stuck at link K. Then, any path request that uses link K will immediately receive a PGRT, whether or not the path can be established to the destination. As a consequence, the source will change the data on the output link from the routing tag to the first data item. Because the path will, in general, not yet have been routed to the destination, an incorrect data item will be used for routing, an incorrect destination will be reached, and data will be lost.

3. FSE Network Models and Properties

3.1 Notation and Definitions

The previous discussion has shown that it is insufficient to model faults as localized entities because faults can propagate. The fault models that are commonly used for the design and analysis of FT networks must therefore be modified. To complement the fault model that describes the physical failures in the system, a *fault propagation (FP) model* is introduced.

The fault model provides an abstraction of a physical failure. A standard model is used that assumes that any network component can deviate from its specified behavior due to a single static hardware failure; transient faults are not considered. Examples of faults that fit this model are stuck-at-faults, open connections, etc. The immediate effect of such a fault could be data and control corruption or non-usability of a component (e.g., a link is broken and no data can pass the link). This could manifest itself in data not being transmitted, incorrect data being transmitted, data being transmitted to an incorrect destination, and spurious data generation. Spurious data generation would be especially problematic in packet-switched networks because it could cause portions of the network to saturate due to hot spot contention [12].

To simplify the following discussion, link faults are treated as switch faults, similar to the approach in [13]. Link faults need therefore not be considered because they can be accounted for by switch faults.

Now consider the FP model. To develop such a model, FSEs are classified; the terminology is defined first.

One of the most important properties of FSEs is their *reach*, i.e., to what extent a fault propagates through the network. Definition 1 formalizes this concept.

Definition 1: Let a fault occur at a switch at stage K. Let the fault have effects at a switch at stage F in the direction of the destination and at a stage G in the direction of the source. The *forward reach* R_F of an FSE is then given by ¦F–K¦, and the *backward reach* R_B is given by ¦K–G¦.

As shown in the case study in Section 2, FSEs may influence all network stages, and the maximum value of R_F or R_B may span all of the network (i.e., be M–1 for an M-stage network). The hardware implementation of switches, links, and protocols has a significant impact on R_F and R_B. One primary design goal of a network that is to tolerate FSEs is to minimize the maximum values of R_F and R_B, ideally to 0 (no side effects).

A second factor that determines the consequences of FSEs is their span, i.e., the number of switches at each stage that are influenced by the fault, as introduced in Definition 2.

Definition 2: The *span* S_L of an FSE at stage L is given by the number of switches at stage L that are affected by the fault.

Many FSEs will be restricted to a single forward or backward path as discussed in Section 2. For these, the span is one for all stages. A stuck broadcast path originating at the input stage could span the complete network at the output stage. Clearly, the span of FSEs should also be kept to a minimum.

If a network is to tolerate FSEs, the propagation of faults must be stopped by some appropriate hardware. The required properties of such hardware are given in Definition 3:

Definition 3: An *isolation point* is a network component that is constructed such that it passes data and control signals freely if set to the operating state, and stops data, control, and fault propagation if set to the disconnecting state.

Isolation points can be located inside the interconnection network, where they might separate links between switches, or they can be located at sources and destinations, where they stop faults from affecting processors or memories. Isolation points may be constructed from hardware devices (e.g., buffers that can be disabled), or they may be conceptual only. In the DR network [14], for example, spare PEs are provided. By not using spare PEs, these are effectively acting as isolation points, so that such a PE can be modelled as being separated from the network by an isolation point, even though such a separation does not exist in the physical sense. For simplification, explicit isolation points will be used throughout the discussion.

Because isolation points may be hardware devices that are set by some control logic, faults of isolation points must be considered. It is assumed that both isolation points and their control lines may fail. Thus, an isolation point can stop fault propagation only if the isolation point and the control logic are fault free, and the control logic is not affected by fault propagation.

3.2 Fault Side-Effect Model

To determine the effects of fault propagation, appropriate models must be developed. These may be very general and permit propagation with large reach and span, or may be restricted. In this paper, a rather general propagation model is assumed. Other fault propagation models are discussed in [9].

Fault Propagation (FP) Model: A fault has side effects that have maximum reach and maximum span, both in forward and backward directions, along all paths that can be reached from the fault until a source, a destination, or a

disconnecting isolation point is encountered. It is assumed that all elements that may be affected by faults do not operate correctly.

Most failures will not have such significant effects, but this model is well suited to illustrate the cababilities of an enhanced ESC to tolerate FSEs.

3.3. Properties of FSE-Tolerant Networks

Before a network can be examined for its capability to tolerate FSEs, the meaning of such a property must be defined.

Definition 4: Let an FT interconnection network maintain property P in the presence of a single fault without side effects. This property could be full access, dynamic full access, etc. [1]. If the network maintains property P under the assumption that faults propagate according to fault propagation model FP, it can tolerate FSEs under the FP propagation model.

In the presence of a fault, a network can always be modelled such that the network is partitioned into a faulty network portion *FNP* that contains the fault and all its propagation paths, and the operational, fault-free portion *ONP*. In the best possible case, FNP would contain only the faulty switch. Then, no fault propagation occurs, and conventional fault modelling suffices. In the worst case, the network could contain, for example, all network output ports so that no messages could be passed through the network (e.g., stuck broadcast path at the input stage of a network without isolation points).

Because FNP contains all fault propagation paths, and fault propagation stops at isolation points only, all connections between ONP and FNP are by definition separated by isolation points. Paths from FNP and ONP can reach sources and destinations, and these connections may be isolation points or not. Theorem 1 states under what conditions a network can tolerate FSEs.

Theorem 1: A fault-tolerant interconnection network with fault-tolerance property P can tolerate FSEs under the FP model if and only if the fault-free network portion ONP provides property P, and all paths connecting FNP to sources or destinations are separated by disconnecting isolation points (see Figure 3).

Proof:

If: Because all fault propagation paths entering or leaving FNP are isolated, ONP is not affected by faults, and no operational PE is affected by a fault. Because ONP provides property P, the network can tolerate FSEs under the FP model. q.e.d.

Only-If: (a) If the fault-free portion ONP does not provide property P, the network loses the fault tolerance capabilities for which it was designed.

(b) Recall from Section 3.1 that all spare PEs are considered to be isolated from the network. It is assumed that all used PEs are required for property P. If any path that connects FNP to sources or destinations is not isolated, faults will propagate from FNP to a used PE, and this affected PE will malfunction. Because all PEs not needed for property P are isolated, at least one PE required for property P malfunctions. Thus, property P is not maintained. q.e.d.

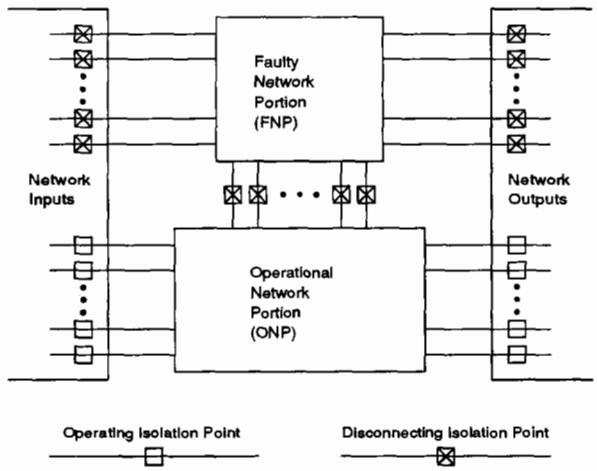

Figure 3: Conceptual view of network partitioning required for fault propagation tolerance

4. Enhancing the Extra Stage Cube Network

In this section, enhancements to the Extra Stage Cube network are discussed to illustrate techniques that can be employed to achieve FSE tolerance. Minor modifications to the ESC are suggested that take advantage of its inherent architectural properties. Let the Enhanced ESC (*EESC*) have the same basic structure as the ESC (i.e., each PE has two links into the network as shown in Figure 1) with the following differences.

(a) All PEs can disconnect themselves from any bypass or interchange box input/output, independently of the bypass state. Such an isolation can be accomplished, for example, by disabling appropriate tristate buffers or ignoring port values.

(b) An enabled bypass isolates the bypass-to-interchange box connection, and a disabled bypass isolates the bypass-to-PE connection.

(c) The output K of an input stage interchange box and the input K of an output stage interchange box can operate as isolation points controlled by PE K.

(d) The bypass and stage enable-lines can operate as disconnecting isolation points so that a fault in a bypass can be stopped from propagating into the PEs.

The model of a 2-by-2 input stage interchange box is depicted in Figure 4. All isolation points required by the above modifications are indicated in the figure. The necessary control lines from PEs to bypasses and interchange boxes are also shown. In Section 5, simple hardware will be proposed that fits these requirements.

Theorem 2: The EESC can tolerate FSEs under the FP model.

Proof: Link faults are treated as PE, interchange box, or bypass faults. PE failures need not be considered because the ESC is not designed to operate under such faults. Thus, only interchange box and bypass failures must be discussed. Five cases must be distinguished: (1) faults in input stage interchange boxes, (2) faults in output stage interchange boxes, (3) faults in interior interchange boxes, (4) faults in input stage bypasses, and (5) faults in output stage bypasses.

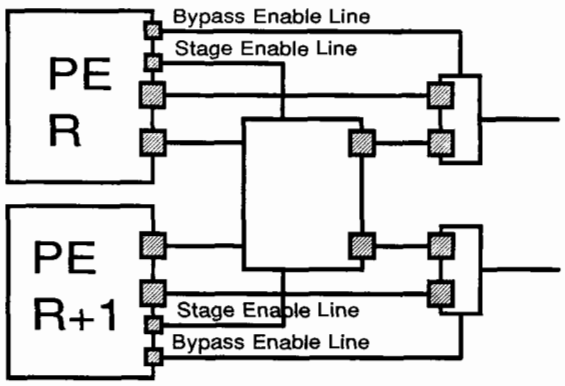

Figure 4: Enhanced ESC input stage and bypass circuits. Shaded squares are isolation points

(1) Faulty input stage interchange box (Figure 5a). By enabling the input stage bypass and isolating the PE outputs that connect to the input stage interchange box, the fault cannot propagate to PEs or into the network. Modifications (a) and (b) are utilized. With respect to Theorem 1, FNP contains the faulty interchange box, and ONP contains the remaining network. All connections between ONP and FNP, and to the PEs connected to FNP, are isolated. ONP provides property P because the fault with propagation paths is equivalent to a single input stage box fault under which the ESC assumptions are valid.

(2) Output stage interchange box fault. Similar to (1).

(3) Faulty interior interchange box (Figure 5b). Without loss of generality, assume that the fault occurred at an interchange box with even numbered inputs. By enabling the input stage and the output stage, and isolating all even numbered input stage outputs and all even numbered output stage inputs, all even numbered interior boxes are disconnected (upper middle rectangle labelled "E" in Figure 5c). Enhancements (b) and (c) are required. Thus, FNP contains all even numbered interior boxes and all even numbered PE bypasses. ONP contains the input stage interchange boxes, the output stage interchange boxes, odd numbered PE bypass circuits, and all interior interchange boxes with odd numbered input links. FNP is isolated from ONP and from the PEs by construction. ONP provides property P because a PE can first send its data to the odd numbered input stage interchange box output, and then route to its destination through the rest of the network. (This is analogous to the way the original ESC avoids a single non-propagating interior interchange box fault.) Thus, the complete path stays within ONP, and Theorem 1 holds true.

(4) Faulty input stage bypass (Figure 5c). Assume that the fault occurs at an even-numbered bypass. Then, the fault is isolated by enabling input and output stages, isolating the bypass connections of all even-numbered PEs (data and enable lines), isolating all even-numbered input stage interchange box outputs, and isolating all even-numbered output stage interchange box inputs. Clearly, the partitioning in this case is identical to that in the case of an interior fault as illustrated by Figure 5c. The only difference is the need to isolate the PE outputs that connect to the even-numbered bypasses. Thus, the same proof as for case (3) applies.

(5) Faulty output stage bypass. Similar to (4). q.e.d.

In contrast to the original ESC fault assumptions, the EESC under the FP model is no longer robust in the presence of multiple faults. As a simple example, consider two faulty interior boxes, one with even, one with odd link numbers. In this case, all even numbered interior boxes belong to FNP because of the first fault, and all odd ones belong to FNP due to the second fault. Thus, no interior interchange box is left in ONP, and thus no paths through ONP are possible.

The total available network bandwidth as compared to the original ESC assumptions is reduced. This is due to the necessity of utilizing only half of the network in case of a single error, which halves the available bandwidth. In the original ESC, only those paths that would have to use the faulty interchange are rerouted, so that only a small loss in bandwidth occurs due to a single fault.

5. EESC Implementation in the PASM Prototype

To illustrate the hardware overhead that is required to enhance the ESC, the PASM protoype interconnection network is examined as an example to demonstrate one approach. The enhancements discussed in Section 4 do not require modification of interior boxes. Only the bypasses, input and output stages, and the PE interfaces need to be discussed. Consider the

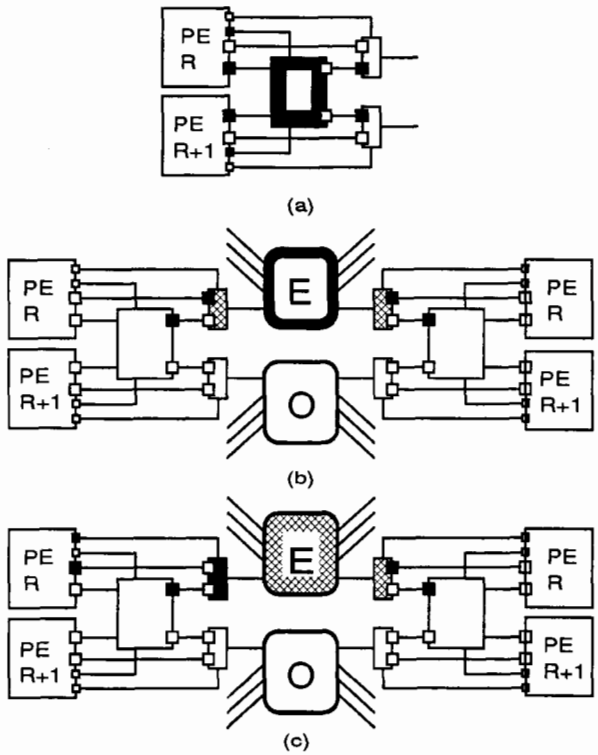

Figure 5: (a) Isolating an input stage interchange box fault; (b) Isolating an interior stage interchange box fault; (c) Isolating an input stage bypass fault. Rectangles labelled "E" represent the subnetwork of all interior interchnage boxes whose links have even numbers. "O" is the same for those with odd numbers

input stage and input bypass structure as shown in Figure 5. In the input stage, four sets of buffers (drivers) (B_UU, B_LU, B_UL, and B_LL) implement the required 2-by-2 crossbar switch. By providing two box enable lines UP_En and LW_En that connect to the box control logic, the buffers can be disabled so that they serve as isolation points; this feature is required to isolate the input stage interchange box from bypass failures. Each bypass contains two buffers that can be enabled through a bypass enable line. Thus, input stage faults can be isolated, and all requirements that are needed to make the ESC FSE-tolerant are met.

This hardware structure serves to illustrate two points. First, little overhead is sufficient in many cases to avoid the fault propagation problem; the hardware of Figure 6 has negligible overhead as compared to a solution that does not consider fault propagation. Secondly, it shows that fault propagation must be carefully considered during the design phase. For example, if no fault propagation were considered, the bypass buffers BYBU and BYBL could be merged with the input stage box buffers. While this does not affect functionality, the input stage would lose its capability to tolerate fault propagation [9].

6. Conclusion

The concept of fault side effects was introduced. It was shown that these side effects may render "fault tolerant" networks inoperable, even as a result of a single fault. A fault model that includes fault side effects was presented and properties of FSE-tolerant networks were described. The capability of some networks for handling fault side effects can be enhanced significantly with little hardware overhead. It is important during the network design and implementation to consider features required to eliminate fault propagation. The model and implementation example given here will aid the network architects and designers to judge the fault tolerance of a particular network in its capabilities in the presence of fault side effects.

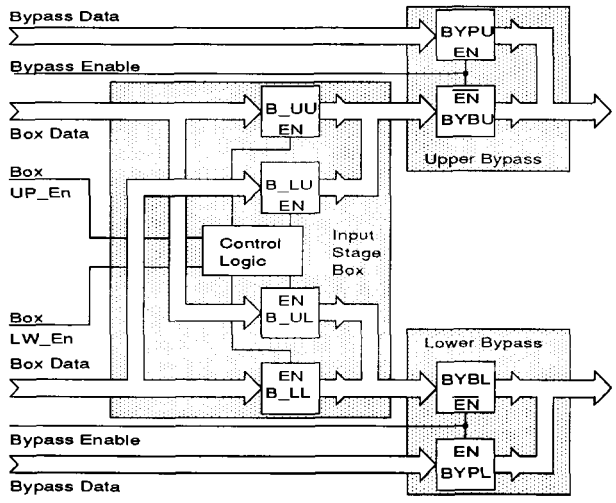

Figure 6: Implementation detail of an enhanced ESC input stage and bypass in the PASM prototype

References

[1] G. B. Adams III, D. P. Agrawal, and H. J. Siegel, "A survey and comparison of fault-tolerant multistage interconnection networks," *Computer*, Vol. 20, No. 6, June 1987, pp. 14-27.

[2] G. J. Lipovski and M. Malek, *Parallel Computing: Theory and Comparisons*, John Wiley and Sons, Inc., New York, NY, 1987.

[3] H. J. Siegel, W. G. Nation, C. P. Kruskal, and L. M. Napolitano, "Using the multistage cube network topology in parallel supercomputers," *Proceedings of the IEEE*, Vol. 77, No. 12, December 1989, pp. 1932-1953.

[4] K. Padmanabhan and D. H. Lawrie, "A class of redundant path multistage interconnection networks," *IEEE Transactions on Computers*, Vol. C-32, No. 12, December 1983, pp. 1099-1108.

[5] D. S. Parker and C. S. Raghavendra, "The gamma network," *IEEE Transactions on Computers*, Vol. C-33, No. 4, April 1984, pp. 367-373.

[6] S. M. Reddy and V. P. Kumar, "On fault-tolerant multistage interconnection networks," *1984 International Conference on Parallel Processing*, August 1984, pp. 155-164.

[7] J. P. Shen and J. P. Hayes, "Fault-tolerance of dynamic-full-access interconnection networks," *IEEE Transactions on Computers*, Vol. C-33, No. 3, March 1984, pp. 241-248.

[8] N. J. Davis IV, W. T.-Y. Hsu, and H. J. Siegel, "Fault location techniques for distributed control interconnection networks," *IEEE Transactions on Computers*, Vol. C-34, No. 10, October 1985, pp. 902-910.

[9] T. Schwederski, E. Bernath, G. Roos, W. G. Nation, and H. J. Siegel, *An analysis of fault side-effects in fault tolerant multistage interconnection networks*, IMS Technical Report TB01-91, May 1991.

[10] H. J. Siegel, T. Schwederski, J. T. Kuehn, and N. J. Davis IV, "An overview of the PASM parallel processing system" in *Computer Architecture*, D. D. Gajski, V. M. Milutinovic, H. J. Siegel, and B. P. Furht, eds., IEEE Computer Society Press, Washington, DC, 1987, pp. 387-407.

[11] G. B. Adams III and H. J. Siegel, "The extra stage cube: a fault-tolerant interconnection network for supersystems," *IEEE Transactions on Computers*, Vol. C-31, No. 5, May 1982, pp. 443-454.

[12] G. F. Pfister and V. A. Norton, "'Hot spot' contention and combining in multistage interconnection networks," *IEEE Transactions on Computers*, Vol. C-34, No. 10, October 1985, pp. 933-938.

[13] A. Varma and C. S. Raghavendra, "Reliability analysis of redundant-path interconnection networks," *IEEE Transactions on Reliability*, Vol. 38, April 1989, pp. 130-137.

[14] M. Jeng and H. J. Siegel, "Design and analysis of dynamic redundancy networks," *IEEE Transactions on Computers*, Vol. C-37, No. 9, September 1988, pp. 1019-1029.