計算機アーキテクチャ　92－9
（1992．　1．　24）

拡張耐故障バッチャ網

陳 澤民　天野 英晴
慶應義塾大学 理工学部

エレメントに関する耐故障性を持つバッチャソータ FTB network (Fault Tolerant Batcher) をリンク故障に対処できるように拡張した EFTB network (Extended FTB) を提案する。EFTB は各エレメントのバイパス機構と冗長リンクを用いることにより、複数のリンク、エレメントが故障しても簡単な回避アルゴリズムでソーティング機能を維持することができる。故障時のルーティング法、VLSI チップ面積を考慮したところ、4096 入力で Survivability が Normal Batcher network より約 7 倍改善されることになった。

An extended Fault Tolerant Batcher Network

Chak-man Chan　Hideharu Amano
Faculty of Science and Technology, Keio University

An extended version of the Fault Tolerant Batcher (FTB) Network is given. This network is aimed to attain a higher survivability over the FTB by making an extension to the link fault tolerant capability. This network is shown to have a better survivability over the normal Batcher Network (about 7 times with 4096 inputs) and also the Fault Tolerant Batcher network with reasonable increase of hardware.

1 Introduction

The Batcher network [1] is a well known parallel sorter consisting of multistage connected switching elements. It is not only utilized as a hardware sorter, but also in packet switching fabrics because the combination of a Batcher network and most blocking multistage switching fabrics, such as the Banyan network or the Omega network, can overcome the problem of internal conflicts. Recently, many switching systems have been proposed for high speed packet switching based on the ATM (Asynchronous Transfer Mode), and some of them have a Batcher-Banyan switching fabric at the core [2].

In a practical switching system, a large scale Batcher network including thousands of switching elements is required to be implemented on a single VLSI chip. In such a large system, fault recovery is one of the critical issue. If the chip is operational even including some faulty links or switching elements, the die yield of the chip is much improved.

Many Fault recovery techniques for multistage interconnection networks with blocking feature have been well researched and many fault tolerant switching fabrics have been proposed [3]. However, these kinds of technique can not be applied to a Batcher network due their difference in nature; In a Batcher network, the permutation of all the output packets must be maintained in the recovery mechanism, while the recovery in a blocking network only aims to establish a path between any pair of a source and destination. Few work has been done on this kind of non-blocking network, and many of them are either dedicated to a specific network or require too much hardware [4] [5].

We have previously proposed a fault tolerant network called Fault Tolerant Batcher (FTB) Network. It is aimed to overcome element faults occurred in the network by adding addition hardware to the normal Batcher network. However, this network is not designed to cope with link faults, and packet will probably be lost with a link fault occurred somewhere in the network. In practice, the wiring area between elements cannot be ignored. In the large size of the network, it becomes larger than that for switching elements, and dominates the chip area.

To due with this problem, redundant links are required to substitute the broken links. Here, we propose an Extended FTB network which can tolerate both element fault as well as link fault.

2 Fault Tolerant Batcher network

A Batcher sorter consists of recursively connected bitonic sorters (Fig.1). Each bitonic sorter merges sorted half-size inputs and generates a full-size sorted result using multistage connected switching elements. Each switching element compares two inputs and transfers the larger one to the link marked '+'.

A faulty element causes the missorted result at the output of bitonic sorter. The output labels where missorted packets will be appeared can be known by the location of the faulty element. FTB(Fault Tolerant Batcher network) [6] provides a correction mechanism at the output of each bitonic sorter. [6] also proposes a diagnosis vector with which the location of the faulty element can be known. The correction mechanism is composed of multiplexors and insertors. It selects the missorted packets according to the diagnosis information, and inserts them into the appropriate position.

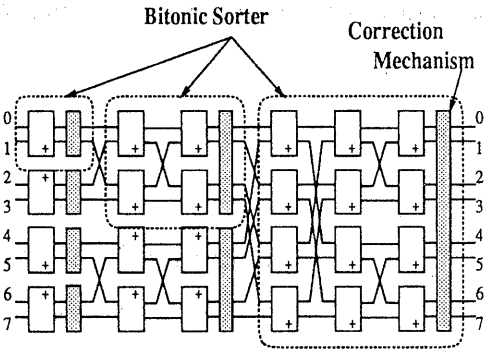

Figure 1: Structure of the FTB

3 Extended FTB

Since a disconnection at links in the network is not under consideration in the FTB, only one faulty link in the network can cause the whole system down. In order to overcome this problem, alternative links are required to prevent the lost of packet which are routed to the faulty link.

Usually, using alternative link would lead to a missorting of packets. Fortunately, the correction mechanism in the FTB can cope with the missorting of packets caused by faulty elements. Therefore, a simple additional bypass mechanism attached to alternative

links can transform a link fault into an element fault which can then be saved by FTB fault recovery mechanism. The network which provides such a mechanism is called Extended Fault Tolerant Batcher network, or EFTB. This network is defined using the following definitions.

Definition 1 *A switching element is a 2-input, 2-output structure used to compare the value of the two packets. The bigger one will be routed to the output with a plus (+) sign. Their output label is different by one bit. Normally its output is determined automatically according to the value of the packet. It can also be set to "straight" mode or "exchange" mode compulsory by external signal.*

Definition 2 *The bypass mechanism is a structure composed of multiplexors and de-multiplexors, which is called "box" in this paper. It is used to provide extra paths for the network.*

Definition 3 *A switching unit is composed of a switching element and bypass mechanism including 5 boxes, except for stage 0, which contains only 4 boxes. Two of them are placed at the input of the switching element, one for each input. And two of them are placed at the output of the switching element, one for each output.*

For stage $1 \leq k < n$, where n is the total stage in the corresponding bitonic sorter, there is the center box. One of the two links are used to join the input box and output box together to form a bypass between the switching element. The other link is joined across the neighboring switching element such that packets can be transfer between switching elements with their labels different at the second last bit.

For stage 0, the input boxes and output boxes are joined together directly, forming two paths which bypass the switching element.

The switching unit is a basic component of the EFTB.

Definition 4 *Every input/output of a switching unit is characterized by its "label". A label is the address of the path given by a multi digit binary number.*

Switching units are joined together by two sets of links: namely, primary link and secondary link.

Definition 5 *Primary Path is the set of path joining the switching units of different stages together by means of perfect shuffle connection. For every $1 \leq k < n$, where n is the total number of stage, label*

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1})$$

is joined to label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1}, a_k)$$

Definition 6 *Secondary Path is the set of path joining the switching units of different stages together by means of perfect shuffle with the last bit reversed. For every $1 \leq k < n$, label*

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1})$$

is joined to label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, \overline{a_{k-1}}, a_k)$$

A correction mechanism is attached to the end of the each bitonic sorter as that in the FTB.

Definition 7 *Every bitonic sorter is joined together by two shuffle connections: One of them is a normal shuffle connection, and the other is a shuffle connection which is joined to the switching element with the last bit of the label reversed.*

In a normal bitonic sorter, switching units are joined by cube connection. For easy labeling, shuffle connection have been chosen since cube connection and shuffle connection is shown to be topologically equal.

Using the definitions above, we can construct the EFTB as shown in fig.2. For the sake of simplification, the correction mechanism in the EFTB is omitted. A bypass mechanism for the alternative link is attached to each link of the switching element. Unlike the switching element, boxes for bypass mechanism are set to the straight or exchange mode by the control signals from outside, and change the mode only when fault is detected. The structure of these boxes are very simple because each of them contains only a multiplexor and de-multiplexor. As a result, the hardware of each box is about one sixth of that of a switching element.

A detail evaluation on hardware will be made in the later part of this paper.

For the connection between the switching units, two sets of link are used: The primary link (the bold lines in fig.2) and the secondary link (doted lines in fig.2).

The bypass switch connects element

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1})$$

and

$$p'_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, \overline{a_{k-1}})$$

to their output.

In this network configuration, it can be shown that multiple paths exists between each source and destination.

Theorem 1 *In the EFTB network configuration, between each stage there exist 2 paths for each output label of a switching element. That is, for any stage*

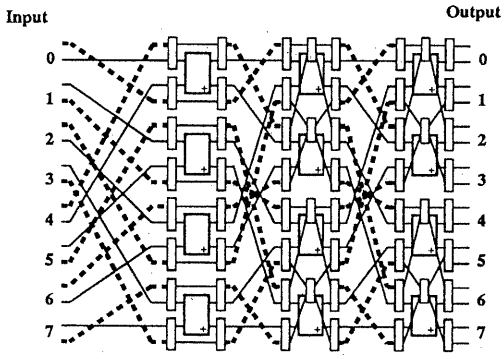

Figure 2: Basic structure of the Extended Fault Tolerant Batcher

$1 \leq k < n$, there exist 2 paths between switching element with output label

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1})$$

and switching element with output label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1}, a_k)$$

Proof:

Let's consider the paths in stage k. The label of the element in stage k is

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1})$$

By definition 5, the primary link is formed by perfect shuffle connection so that it is joined to the the element in $k+1$ with label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1}, a_k)$$

On the other hand, when the secondary path is used, element in stage k with label

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1})$$

is joined to the element with label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, \overline{a_{k-1}}, a_k)$$

and then through the bypass mechanism to the element with label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, a_1, \ldots, a_{k-1}, a_k)$$

Because there are no common element in the two paths, it follows that the two paths is mutual independent with each other; that is, two disjoint paths.□

Lemma 1 *There exist 2 paths between every source and destination if the input permutation is a direct mapping with same source and destination label.*

Proof:

It follows directly from the proof of theorem 1; There are the Primary path as well as the secondary path between each stage, which provide two disjoint paths in between the source and destination if all the elements are set to the straight position. This can be done if and only if they have the same source and destination label.

If the source and destination label are different, the secondary path may be fail because routing will be intervened by the switching element, other than perfect shuffle only. □

Follow from this lemma, it can be shown that multiple paths exist between the source and destination. This is a necessary condition to recover a link fault. The mechanism of fault recovery will be shown in the next section.

4 Fault Recovery

In the fault free case, EFTB uses the primary link, and packets are routed in the same manner as in the Batcher network.

When a fault is detected, the following strategy will be used: The EFTB will figure out the location of the fault and try to transform it into an element fault which can be resolved in the FTB. Any kind of fault in the EFTB will be resolved into "element straight stuck" fault or "element exchange stuck" fault. After that, correction mechanism, which is composed of comparators and multiplexors, are used to correct those packets which are missorted at the end of the bitonic sorter.

A fault other than an element fault in the EFTB is either a link fault or malfunction of the bypass mechanism. First, let us consider the following lemmas:

Lemma 2 *A link fault can be transformed into an element fault by using the secondary path and the bypass mechanism.*

Proof:

Let the link fault be in stage k. The output label of this broken primary link is

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1})$$

The packet using this label will be routed to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1}, a_k)$$

On the other hand, packets using the secondary link will be route from

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1})$$

to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, \overline{a_{k-1}}, a_k)$$

and through the bypass mechanism to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1}, a_k)$$

Since the switching element of stage $k+1$ is bypassed, the last bit of the label is fixed. This can be treated as an element fault.□

According to theorem 1, we have the primary path functions the same as the secondary path with the bypass mechanism, except that the last bit is fixed to "1" or "0", which is corresponding to an "element straight stuck".

Lemma 3 *A fault at the input/output box of the bypass mechanism can be transformed into an element fault using the secondary path and the bypass mechanism.*

Proof:

Let's consider the box fault in stage $k+1$. For the input box fault, the same argument could be used as in lemma 2. The input label of stage $k+1$ is equal to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1}, a_k)$$

which is linked to the output label of stage k whose label is

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1})$$

This path can be avoided by using the secondary path of stage k, from

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1})$$

to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, \overline{a_{k-1}}, a_k)$$

and through the bypass mechanism to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1}, a_k)$$

For the output box fault, both the primary and the secondary link of the neighboring box can be used to substitute the link connected to the faulty box.

The secondary link of the neighboring box can be shown to be the same as the primary path as follow:

Let's consider the the switching element in stage k. Let the output label of the upper box be

$$p_k(a_k, a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1})$$

It is joined to

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1}, a_k)$$

by perfect shuffle connection of the primary path in stage $k+1$.

On the other hand, the upper box can be joined to the element in stage $k+1$ with label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, \overline{a_{k-1}}, a_k)$$

through the secondary path, which is then joined to the element with label

$$p_{k+1}(a_{k+1}, \ldots, a_n, a_0, \ldots, a_{k-1}, a_k)$$

of stage $k+1$ through the bypass mechanism.□

It follows that the EFTB can tolerate element faults as well as multiple link faults, 2 at most in every switching unit. Fault recovery is impossible if more than 3 out of 4 links are found to be failed.

5 Routing

The routing mechanism of the EFTB is basically the same as in the FTB. However, when faults are occurred in the network, the routing mechanism has to be modified. Here, we define some rules to represent the routing scheme.

Definition 8 *A route is represented by a routing function. It is a "series of mode" of elements get through by the packet in sequence.*

"Switching element" and "box" included in each path. The mode of a switching element is denoted by a capital letter:

- *'C' normal compare mode*
- *'E' compulsory exchange mode*
- *'S' compulsory straight mode*

The mode of a box is denoted by a small letter:

- *'s' straight mode*

◇ *'e' exchange mode*

'$\bar{s}$' and '$\bar{e}$' are used to denote the mode of the "center box", that is, the box in between two boxes.

Firstly, let us consider the fault-free case, that is, network operating in normal condition. In this case, only primary link will be used. All the boxes are set to the straight position. All the bypass mechanism and correction mechanism will be made inactive. The routing scheme can be represented by the following following function:

$$r_i(\ldots, s, C, s, s, C, s, \ldots), \quad \forall i$$

where i is the number of route. Packets are self-routing according to their headers. Everything works like a normal Batcher network.

5.1 Single fault

The fault recovery mechanism of the EFTB is capable to tolerate a single fault (faulty switching unit) in every diagnosis group. A fault may be occurred in all part of the network, including switching elements, links or boxes of the bypass mechanism.

According to definition 3, 2 paths exist between every input and output with the same label. In case of switching element fault, the faulty switching element can be bypassed using the extra path of the bypass mechanism. The box is set to exchange position by external control signal by the following function:

$$r_i(\ldots, e_k, \bar{s}_k, e_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots)$$

where the subscript k represents the stage where the fault occurred, and i represents the 2 routes get through the faulty element. And the routing function is

$$r_j(\ldots, s_k, C_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots) \quad \forall j \neq i$$

for other paths.

The ability to tolerate a switching element fault is better in the EFTB than in the FTB since the fault model in the EFTB included internal blocking of the element, which is not under consideration in the FTB.

When a fault is occurred in the input/output box of the bypass mechanism, according to lemma 3 it is at first transformed into "element fault" and then it is recovered using the correction mechanism. The routing mechanism is different in both cases. For a faulty input box, The paths of the packets getting through the switching unit containing the faulty input box will be

$$r_i(\ldots, e_{k-1}, s_k, \bar{e}_k, e_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots)$$

and for all other paths the routing function is

$$r_j(\ldots, s_k, C_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots) \quad \forall j \neq i$$

In case of a faulty output box, The paths of the packets getting through the switching unit containing the faulty output box will be

$$r_{i_1}(\ldots, e_k, \bar{s}_k, s_k, e_{k+1}, C_{k+1}, s_k, \ldots)$$

and

$$r_{i_2}(\ldots, s_k, E_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots)$$

For all other paths, the routing function is

$$r_j(\ldots, s_k, C_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots) \quad \forall j \neq i_1 \neq i_2$$

5.2 Multiple fault

Since at most one fault can be tolerated in every bitonic sorter, the EFTB network as a whole is multiple fault tolerant. A *fault*, in this case, is referred to a *faulty switching unit*. A switching unit, as defined in definition 3, is composed of a switching element, bypass mechanism and a few boxes. Multiple fault tolerant is permitted *within* a switching unit in some cases. The routing function for the EFTB with a multiple faulted switching unit is given.

In lemma 3, we have shown that any 2 of the 4 paths of a switching unit can be used for routing. It follows that for any faulty switching unit in stage k where $1 \leq k < n$, one of the following routing functions can be employed:

◇ For link fault on both secondary link,

$$r_i(\ldots, s_k, C_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots)$$

for both routes.

◇ For link fault on both primary link,

$$r_i(\ldots, s_k, C_k, e_k, e_{k+1}, C_{k+1}, s_{k+1}, \ldots)$$

for both routes.

◇ In case of a primary link fault at one output and a secondary link fault at the other output,

$$r_{i_1}(\ldots, s_k, C_k, e_k, s_{k+1}, \bar{s}_{k+1}, e_{k+1}, \ldots)$$

for one route and

$$r_{i_2}(\ldots, s_k, C_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots)$$

for the other.

For other non-faulty switching units, the routing function is

$$r_j(\ldots, s_k, C_k, s_k, s_{k+1}, C_{k+1}, s_{k+1}, \ldots) \quad \forall j \neq i$$

Besides multiple link fault, other combination of link fault and element fault can also be tolerate. For instance, an switching element can be coupled with a bypass link fault; a secondary link fault can be occurred at the same time with an element fault. In these cases, the routing function is unchanged and is the same as a link fault or an element fault.

6 Evaluation

The cost-performance of the EFTB network can be estimated in terms of network survivability and hardware involved. Simulation is performed to compare the survivability and hardware involved between the normal Batcher network, FTB and EFTB.

6.1 Hardware consideration

We have considered the hardware of the EFTB in terms of total chip surface area. The hardware is estimated by measuring the total area occupied by the wiring diagram of the layout of the network using CAD (fig.3).

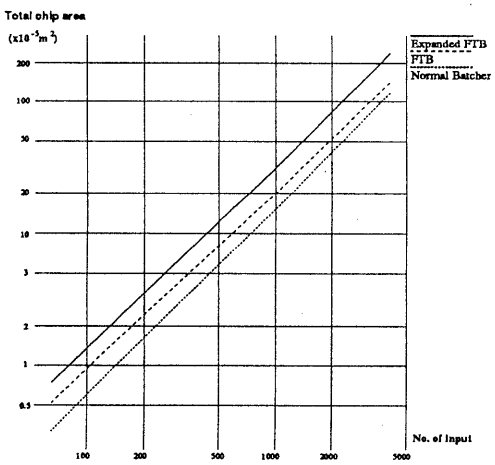

Figure 3: Comparison on hardware

From fig.4, we can notice that the hardware of EFTB is about 2 times the hardware of normal Batcher network. In spite of this, such among of hardware increase is consider reasonable as the survivability of the system is significant enhanced as the number of input grows.

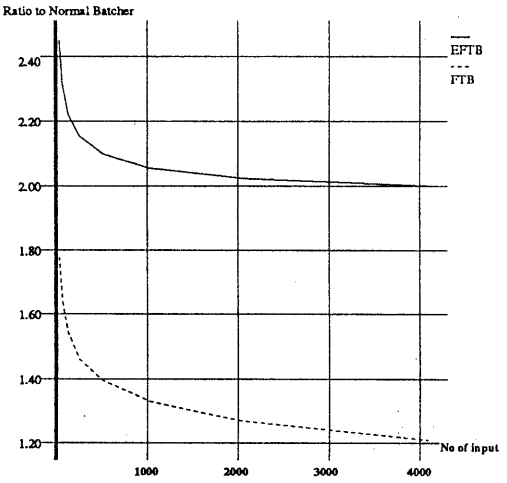

Figure 4: Ratio of hardware to Normal Batcher Network

6.2 Survivability

Survivability is defined as the ability to restore network service in the event of component failure. An component failure could be a faulty link or faulty element. We assume that the probability of the failure is proportional to the surface area of the layout. The survivability can be calculated based on the above mentioned hardware.

The survivability of EFTB and FTB are compared with the normal Batcher network. The result is shown in the following graph. The survivability of the normal Batcher network is 1 since a single link fault or an element fault is fatal. According to this graph, the survivability of the EFTB is about 7 times better than the normal Batcher network with 4096 inputs. Also, we can notice that the improvement of the survivability of the FTB tends to be saturated as the number of input increase, and reach an upper limit of 5. This can be explained by the fact that FTB cannot cope with link fault, and the percentage of link increase with growing network size, where link fault is dominant.

7 Conclusion

The EFTB makes up for the weak points of FTB, which can cope with both link faults and element faults. Moreover, the multi-fault tolerant ability of the network is improved thanks to the existence of multiple

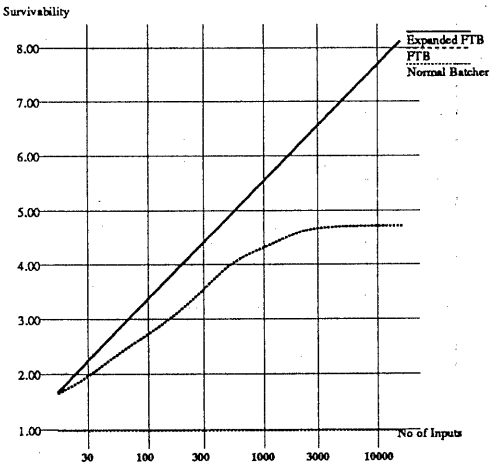

Figure 5: Comparison on survivability

paths between every stage.

On the other hand, hardware is increased significantly to about 2 times the normal batcher sorter, independent of the scale of the network. But considering the cost-performance gained by the EFTB, such amount of hardware increase is acceptable.

Acknowledgment

I would like to thank Dr. Tomohiro Kudoh (Tokyo Engineering University), Dr. Tetsuro Kimura and Dr. Takuya Terasawa (Keio University) for their valuable advice and support. I would also like to thank Mr. Yulu Yang for his assistance on network layout.

References

[1] K.E.Batcher. Sorting networks and their applications. *Proc. of Spring Joint Computer Conference*, 1968,pp.307-314.

[2] A.Huang and S.Knauer. Starlite: A wideband digital switch. *Proc. of 1984 Globecom Conf.*, 1984.

[3] D.P.Agrawal G.B.Adams III and H.J.Siegel. A survey and comparison of fault-tolerant multistage interconnection networks. *IEEE Computer*, June. 1987, pp.14-27.

[4] N.Arakawa H.Karasawa, H. Amano. A fault diagnosis method for the batcher network. *JTC-CSCC*, '89,pp.487-492.

[5] Eli Upfal Shay Assaf. Fault tolerant sorting network. *SIAM, Transaction On Discrete Mathematics*, 1990.

[6] H.Amano. A fault diagnosis and recovery technique for a batcher sorter. *Proc. of Inter. Conf. on Parallel Processing*, 1990,I-pp.441-444.

[7] K.E.Batcher. On bitonic sorting networks. *Proc. of Inter. Conf. on Parallel Processing*, 1990,I-pp.376-379.

[8] S.Horiguchi. Systolic sorter for wsi implementation. *Proc. IEEE International Conference on Wafer Scale Intergration*, Jan, 1989, pp.151-160.

[9] F.A.Tobagi and T.Kwok. Fast packet switch architectures and the tandem banyan switching fabric. *Proc. NATO Workshop on Architecture and Performance Issues of High-Capacity Local and Metropolitan Area Networks, Sophia Antipolis, France*, June 25-27, 1990.