

Some Properties for a Class of Multistage Interconnection Networks

¹S.K.Hui, ²K.Seman, Senior Member IEEE, and ³J.Yunus

Faculty of Electrical Engineering
Universiti Teknologi Malaysia
81300 Skudai, Johor, Malaysia

Tel: (607)-5505106 Fax:(607)-5566272
Emails: ¹sk_hui@hotmail.com
²kzaman@suria.fke.utm.my
³jasmy@suria.fke.utm.my

Abstract-Derivations of some properties for multistage interconnection(MIN) will be presented in this paper. These derivations are focused on baseline network, data manipulator(modified version), flip network, indirect binary n -cube network, omega network, and regular SW banyan network with spread and fanout of 2($S=F=2$). The derivations contain two parts; part 1 is a supplementary proof of equations (1) and (2) in [8], part 2 is a proof of connectivity between stages for various MIN as mentioned above. We may use the proof of part 2 to build fault-tolerant feature in various MIN networks.

1. INTRODUCTION

The asynchronous transfer mode(ATM) is the transport technique for the broadband ISDN recommended by CCITT. It requires fast packet (cell) switching. Among the switches have been proposed, self-routing switches which apply a banyan network are suitable candidates because of their inherent parallel processing ability. Among the banyan networks that were included in this category were baseline network[1], data manipulator(modified version)[2], flip network[4], indirect binary n -cube network[5], omega network[6], and regular SW banyan network with spread and fanout of 2($S=F=2$)[3],[7]. Each of these networks consists of a set of N input terminals and the set of N output terminals are two disjoint sets of terminals. All these networks are capable of connecting an arbitrary input terminal to an arbitrary output terminal. But simultaneous connections of more than one terminal pair may result in conflicts in the communication path within the logic cells.

This paper evaluates the properties among these networks. The paper is divided into two main parts. *Part I* was a supplementary proof of equations (1) and (2) in [8]. In [8], the authors have escaped some important steps in their derivations, so part I serves as a supplementary reading for that purpose. *Part II* is a new property that not yet explored by other researchers. The authors of this paper believe that it

may have practical implication that lead to constructing fault-tolerant networks among the MIN.

For further clarification, please refer to [1] for all the network figures and mathematical notations that were mentioned in this paper.

2. PART 1'S PROOF

Before the proof of equations (1) and (2) in [1], one property of baseline network should be understood first. Let us call it reverse property of baseline network. The property stated that:

In phase 1 test (all the 2x2 SE in direct connection), a network with the source terminal = $p_1 p_{l-1} p_{l-2} \dots p_0$ and destination terminal = $z_1 z_{l-1} z_{l-2} \dots z_0$, then $z_{l-i} = p_i$.

In phase 2 test (all the 2x2 SE in crossed connection), a network with the source terminal = $p_1 p_{l-1} p_{l-2} \dots p_0$ and destination terminal = $z_1 z_{l-1} z_{l-2} \dots z_0$, then $z_{l-i} = \overline{p_i}$.

Phase 1 test

For a network source terminal $A = a_l a_{l-1} \dots a_0$ and destination terminal $Z = z_l z_{l-1} \dots z_0$. The in-path link in level $i+l$ will be $(z_l \dots z_{l-i+1} a_l \dots a_{i+1} z_{l-i})_{i+1}$. We use input terminal (source terminal) as $p_1 p_{l-1} \dots p_0$. Therefore, we have

in-path link in level $i+l$ is = $\underbrace{(z_l z_{l-1} \dots z_{l-i+1})}_{i \text{ bus}} \underbrace{(p_l \dots p_{i+1} z_{l-i})}_{(l-i+1) \text{ bus}}_{i+1}$

Let $i=0$,

Input at $(p_l \dots p_0)_0 = \text{Input at } (p_l \dots p_1 z_l)_1$

but $z_l = p_0$, therefore,

$$\text{Input at } (p_l \dots p_0)_0 = \text{Input at } (p_l \dots p_1 p_0)_1 \quad (2.1)$$

Let $i=1$,

$$\text{Input at } (p_l \dots p_0)_1 = \text{Input at } (z_l p_l \dots p_2 z_{l-1})_2$$

but $z_l = p_0$ and $z_{l-1} = p_1$, therefore,

$$\text{Input at } (p_l \dots p_0)_1 = \text{Input at } (p_0 p_l \dots p_2 p_1)_2 \quad (2.2)$$

Let $i=2$,

$$\text{Input at } (p_0 p_l \dots p_1)_2 = \text{Input at } (p_0 z_{l-1} p_l \dots p_3 z_{l-2})_3$$

but $z_{l-1} = p_1$ and $z_{l-2} = p_2$, therefore,

$$\text{Input at } (p_0 p_l \dots p_1)_2 = \text{Input at } (p_0 p_1 p_l \dots p_3 p_2)_3 \quad (2.3)$$

By observing equations (2.1),(2.2) and (2.3), we can conclude that

$$\text{Input at } (p_l p_{l-1} \dots p_0)_0 = \text{Input at } (p_0 p_l \dots p_{l-2} p_l \dots p_{l-1})_i$$

for phase 1, $0 < i \leq l+1$

which is the same as equation (1) in [1]. Next we will prove the equation (2) in [1].

Phase 2 test

For a network source terminal $A = a_l a_{l-1} \dots a_0$ and destination terminal $Z = z_l z_{l-1} \dots z_0$. The in-path link in level $i+l$ will be $(z_l \dots z_{l-i+1} a_l \dots a_{l-i+1} z_{l-i})_{i+l}$. We use input terminal(source terminal) as $p_l p_{l-1} \dots p_0$. Therefore, we have

$$\text{in-path link in level } i+l \text{ is } = \underbrace{(z_l z_{l-1} \dots z_{l-i+1})}_{i \text{ bits}} \underbrace{(p_l \dots p_{l-i+1} z_{l-i})}_{(l-i+1) \text{ bits}}_{i+l}$$

Let $i=0$,

$$\text{Input at } (p_l \dots p_0)_0 = \text{Input at } (p_l \dots p_1 z_l)_1$$

but $z_l = p_0$, therefore,

$$\text{Input at } (p_l \dots p_0)_0 = \text{Input at } (p_l \dots p_1 p_0)_1 \quad (2.4)$$

Let $i=1$,

$$\text{Input at } (p_l \dots p_0)_1 = \text{Input at } (z_l p_l \dots p_2 z_{l-1})_2$$

but $z_l = p_0$ and $z_{l-1} = p_1$, therefore,

$$\text{Input at } (p_l \dots p_0)_1 = \text{Input at } (p_0 p_l \dots p_2 p_1)_2 \quad (2.5)$$

Let $i=2$,

$$\text{Input at } (p_0 p_l \dots p_2 p_1)_2 = \text{Input at } (p_0 z_{l-1} p_l \dots p_3 z_{l-2})_3$$

but $z_{l-1} = p_1$ and $z_{l-2} = p_2$, therefore,

$$\text{Input at } (p_0 p_l \dots p_2 p_1)_2 = \text{Input at } (p_0 p_1 p_l \dots p_3 p_2)_3 \quad (2.6)$$

Let $i=3$,

$$\text{Input at } (p_0 p_l p_l \dots p_3 p_2)_3 = \text{Input at } (p_0 p_l z_{l-2} p_l \dots p_4 z_{l-3})_4$$

but $z_{l-2} = p_2$ and $z_{l-3} = p_3$, therefore,

$$\text{Input at } (p_0 p_l p_l \dots p_3 p_2)_3 = \text{Input at } (p_0 p_l p_2 p_l \dots p_4 p_3)_4 \quad (2.7)$$

By observing equations (2.4),(2.5) and (2.6), and (2.7) we can conclude that

$$\text{Input at } (p_l p_{l-1} \dots p_0)_0 = \text{Input at } (p_0 p_l \dots p_{l-2} p_l \dots p_i p_{l-1})_i$$

for phase 2, $0 < i \leq l+1$

which is the same as equation (2) in [1].

3. PART 2'S PROOF

For the six networks architecture we mentioned above, we found all of the networks have one common property. The property stated that: *two switch elements(SEs) in stage i will connect directly to stage i+1 and the connections is independent of i and N where $0 \leq i < l$, $l = \log_2 N$, N is the number of input terminals. An example is given; in figure 3.0, S_{00} and S_{0l} in stage 0 are connected directly to two same SE, S_{10} and S_{1l} in stage 1.*

The connectivity of baseline network was defined as:

$$\alpha_i^0[(p_l p_{l-1} \dots p_1)_i] = (p_l \dots p_{l-i+1} p_{l-i} \dots p_2)_{i+1}$$

for link $(p_l p_{l-1} \dots p_1 0)_{i+1}$, $0 \leq i < l$

(3.0)

and

$$\alpha_i^1[(p_l p_{l-1} \dots p_1)_i] = (p_l \dots p_{l-i+1} p_{l-i} \dots p_2)_{i+1}$$

for link $(p_l p_{l-1} \dots p_1 1)_{i+1}$, $0 \leq i < l$

(3.1)

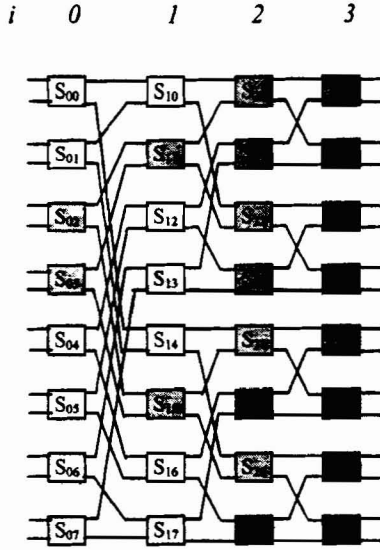


figure 3.0:
baseline network

the α_i 's describe the interconnection by mapping a switching element in stage i to two switching elements in stage $i+1$, one element per link "out of" the switching element in stage i .

By calculating the number of bits, we have the following new label of number of bits in the right hand side of that equation.

$$\alpha_i^0[(p_1 p_{l-1} \dots p_1)_i] = (\underbrace{p_1 \dots p_{l-i+1}}_{(i+1) \text{ bits}} \underbrace{0 p_{l-i} \dots p_2}_{(l-i-1) \text{ bits}})_{i+1} \quad (3.2)$$

$$\alpha_i^1[(p_1 p_{l-1} \dots p_1)_i] = (\underbrace{p_1 \dots p_{l-i+1}}_{(i+1) \text{ bits}} \underbrace{1 p_{l-i} \dots p_2}_{(l-i-1) \text{ bits}})_{i+1} \quad (3.3)$$

the label of number of bits in right hand of equations 3.2 and 3.3 contains two fields, let us called it right bit field and left bit field. Right bit field refers to right hand label and left bit field refers to left hand label. The left bit fields contained $(i+1)$ bits, since $0 \leq i < l$, the numbers of bits will keep on increasing from 1 bit to l bits when i increase from 0 to $(l-1)$ bits.

When $i=0$, equations 3.2 and 3.3 will be

$$\alpha_0^0[(p_1 p_{l-1} \dots p_1)_0] = (0 p_1 p_{l-1} \dots p_2)_1 \quad (3.4)$$

$$\alpha_0^1[(p_1 p_{l-1} \dots p_1)_0] = (1 p_1 p_{l-1} \dots p_2)_1 \quad (3.5)$$

By observing the right hand side of equations 3.2 and 3.3 carefully, the 0 in equation 3.2 and 1 in equation 3.3 will shift gradually to right side until it

becomes the least significant bit. It happens when $i=l-1$, by this time equations 3.2 and 3.3 will be

$$\alpha_0^0[(p_1 p_{l-1} \dots p_1)_0] = (p_1 p_{l-1} \dots p_2 0)_1 \quad (3.6)$$

$$\alpha_0^1[(p_1 p_{l-1} \dots p_1)_0] = (p_1 p_{l-1} \dots p_2 1)_1 \quad (3.7)$$

therefore 0 and 1 in left bit fields in equations 3.2 and 3.3 are independent of the value of i and l . For every value of i , 0 and 1 in the right hand of the equations 3.0 and 3.1 will cause two same switching element connected to their respective switching elements. Let us take some examples to illustrate our idea.

If $i=0$, $l=3$, equations 3.0 and 3.1 will be

$$\alpha_0^0[(p_3 p_2 p_1)_0] = (0 p_3 p_2)_1 \quad (3.8)$$

$$\alpha_0^1[(p_3 p_2 p_1)_0] = (1 p_3 p_2)_1 \quad (3.9)$$

Let us randomly substitute $p_3=1$ and $p_2=1$, we will have

$$\alpha_0^0[(11 p_1)_0] = (011)_1 \quad (3.10)$$

$$\alpha_0^1[(11 p_1)_0] = (111)_1 \quad (3.11)$$

Since value of p_1 can either be 0 or 1, therefore we have switch element 110, 111 connected to 011 and 111 from stage $i=0$ to stage $i=1$. The complete permutation of p_3 , p_2 and p_1 will make a complete connections from stage $i=0$ to $i=1$.

If $i=1$, $l=3$, equations 3.0 and 3.1 will be

$$\alpha_0^0[(p_3 p_2 p_1)_1] = (p_3 0 p_2)_2 \quad (3.12)$$

$$\alpha_0^1[(p_3 p_2 p_1)_1] = (p_3 1 p_2)_2 \quad (3.13)$$

Let us randomly substitute $p_3=1$ and $p_2=1$, we will have

$$\alpha_0^0[(11 p_1)_1] = (101)_2 \quad (3.14)$$

$$\alpha_0^1[(11 p_1)_1] = (111)_2 \quad (3.15)$$

Since value of p_1 can either be 0 or 1, therefore we have switch element 110, 111 connected to 101 and 111 from stage $i=1$ to stage $i=2$. The complete permutation of p_3 , p_2 and p_1 will make a complete connections from stage $i=1$ to $i=2$.

If $i=2$, $l=3$, equations 3.0 and 3.1 will be

$$\alpha_0^0[(p_3 p_2 p_1)_2] = (p_3 p_2 0)_3 \quad (3.16)$$

$$\alpha_0^1[(p_3 p_2 p_1)_2] = (p_3 p_2 1)_3 \quad (3.17)$$

Let us randomly substitute $p_3=1$ and $p_2=1$, we will have

$$\alpha_0^0[(11p_1)_2] = (110)_2 \quad (3.18)$$

$$\alpha_0^1[(11p_1)_2] = (111)_2 \quad (3.19)$$

Since value of p_1 can either be 0 or 1, therefore we have switch element 110, 111 connected to 110 and 111 from stage $i=2$ to stage $i=3$. The complete permutation of p_3, p_2 and p_1 will make a complete connections from stage $i=2$ to $i=3$. This ends our proof for baseline network.

The proofs for the rest of the networks are quite similar to baseline. Therefore we have completed the proofs of our properties.

4.0 CONCLUSION

Since we know that two SE in stage i will connect directly to two same SE in stage $i+1$. Therefore the six MINs we have mentioned above inherently provided a feature of fault-tolerant, i.e. if one SE in stage i in faulty stage, we can use some components to bypass that faulty SE and redirect the cells to the normal SE.

References

- [1] C.L. Wu and T.Y. Feng, "On a Class of Multistage Interconnection Networks", *IEEE Trans. Comput.*, vol c-29, no 8, pp. 694-702, August 1980.
- [2] T.Y. Feng, "Data manipulating functions in parallel processors and their implementations," *IEEE Trans. Comput.*, vol. C-23, pp. 309-318, March 1974.
- [3] G.L. Lipovski and A. Tripathi, "A reconfigurable varistructure array processor", *Proc. Int. Conf. Parallel Processing*, pp. 165-174, 1977.
- [4] K.E. Batcher, "The flip network in STARAN," *Proc. Int. Conf. Parallel Processing*, pp. 65-71, 1976.
- [5] M.C. Pease, III, "The indirect binary n-cube microprocessor array," *IEEE Trans. Comput.*, vol. C-26, pp. 458-473, May 1977.
- [6] D.K. Lawrie, "Access and alignment of data in an array processor," *IEEE Trans. Comput.*, vol c-24, pp. 1145-1155, December 1975.

[7] L.K. Goke and G.J. Lipovski, "Banyan networks for partitioning multiprocessing systems," *Proc. 1st Annu. Comput. Architecture Conf.*, pp. 21-28, December 1973.

[8] T.Y. Feng and C.L. Wu, "Fault-Diagnosis for a Class of Multistage Interconnection Networks", *IEEE Trans. Comput.*, vol. c-30, No 10, October 1981.