MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LABORATORY FOR COMPUTER SCIENCE
CAMBRIDGE, MASSACHUSETTS

COMPUTATION STRUCTURES GROUP MEMO 214-1

A TEST STRATEGY FOR PACKET SWITCHING NETWORKS

by

WILLIE Y-P. LIM

This research was supported by the National Science Foundation under grant number MCS-7915255 and the Department of Energy under contract number DE-AC02-79ER10473.

(An abbreviated version of this paper was published in the Proceedings of the 1982 International Conference on Parallel Processing.)

MARCH 1982
(REVISED MARCH 1983)

A Test Strategy for Packet Switching Networks

Willie Y-P. Lim
MIT Laboratory for Computer Science
Cambridge, Massachusetts 02139

26 February 1982

Abstract

A test strategy for packet switching networks is described. The effect of a single stuck-at fault is either misdirected packets, missing packets, corrupted data in packets, or multiple packets. A fault can either prevent packet transmission or affect the integrity of the data sent in the packet and it is detected as one of 4 cases - both output ports of the switching element inaccessible to an input port, an output port inaccessible to an input port, an input permanently connected to an output port and erroneous packet length.

1. Introduction

Packet communication architecture has been discussed in the implementation of data-flow machines [1]. Such systems use packet switching networks for inter-processor connection. In [1] for example, the network used is composed of packet switching elements called 2 × 2 routers. Packet switching for another class of networks are discussed in [5]. Each packet is routed through the network using the information carried in the packet. Due to this distribution of the switching function, many packets can be simultaneously transmitted through the various stages of the network. When asynchronous or self-timed communication protocols are used, the testing of such networks requires new approaches. A strategy for testing such networks is described in this paper.

Fault diagnosis of networks has been studied by [4] for on-line fault diagnosis and by Wu and Feng [8]. The work in [8] dealt mainly with the fault diagnosis of networks in which the switching elements have single bit inputs. Wider inputs are used in packet communication. Furthermore the packet format and

[1] This research was supported by the National Science Foundation under grant number MCS-7915255 and the Department of Energy under contract number DE-AC02-79ER10473.

communication protocol used affect the test and fault diagnosis strategy. These two factors are first discussed in Section 2 while Section 3 described the type of packet switching network considered. A fault model for packet switching networks is presented in Section 4 and the test strategy and fault diagnosis of such a network are discussed in Sections 5 and 6 respectively. The conclusion of the paper is given in Section 7.

2. Packet Format and Packet Communication Protocol

A packet is a sequence of bits and is usually transmitted as a sequence of sub-units with each sub-unit being some fixed number of bits. For convenience, a sub-unit is referred to as a byte. The number of bits in a byte is usually determined by chip pin-out and communication bandwidth considerations. The information contained in a packet is composed of the destination address, the data to be sent and the length of the packet. Since only packet switched networks are considered in this paper, the destination address is necessary for routing the packets through the network. The packet length information can be included in the data transmitted, or an extra bit can be used to indicate which byte is the last one in the packet.

Packets are assumed to be transmitted using asynchronous communication protocols. The transmission of each packet or byte is accompanied by an event signalling the arrival of the packet or byte at the destination and each successful receipt of a packet must be acknowledged by the explicit sending of a control signal. For example, a special signal may be used to indicate the arrival of the packet, or the arrival event may be encoded in the data signal lines as in the "dual-rail" communication protocol [3].

3. A Packet Switching Network

The switching element in the network is a 2 × 2 router which receives packets at its two input ports and sends them out at its two output ports. The least significant bit of the address byte of the packet is used for selecting the output for sending the packet. Output ports can be independently selected by the input ports and if there is contention for an output port, only one of the input ports is connected while the other waits until the output port becomes free, i.e. the input is temporarily blocked. If there is no contention for an output port, then the packet transmission from an input port to an output port can proceed in parallel. The various input-output port configurations possible are shown in Figure 1. The least significant bit of the destination address byte having a value of 0 will cause output port 0 to be selected while output port 1 will be selected if that bit is 1.

The packet switching network has the same interconnection structure as the baseline network [6, 7]. Figure 2 shows the structure of a 16 × 16 network. Each router in the network is connected to another router or a processor through links. For a network with N input ports and N output ports, there are $\log_2 N$ stages of

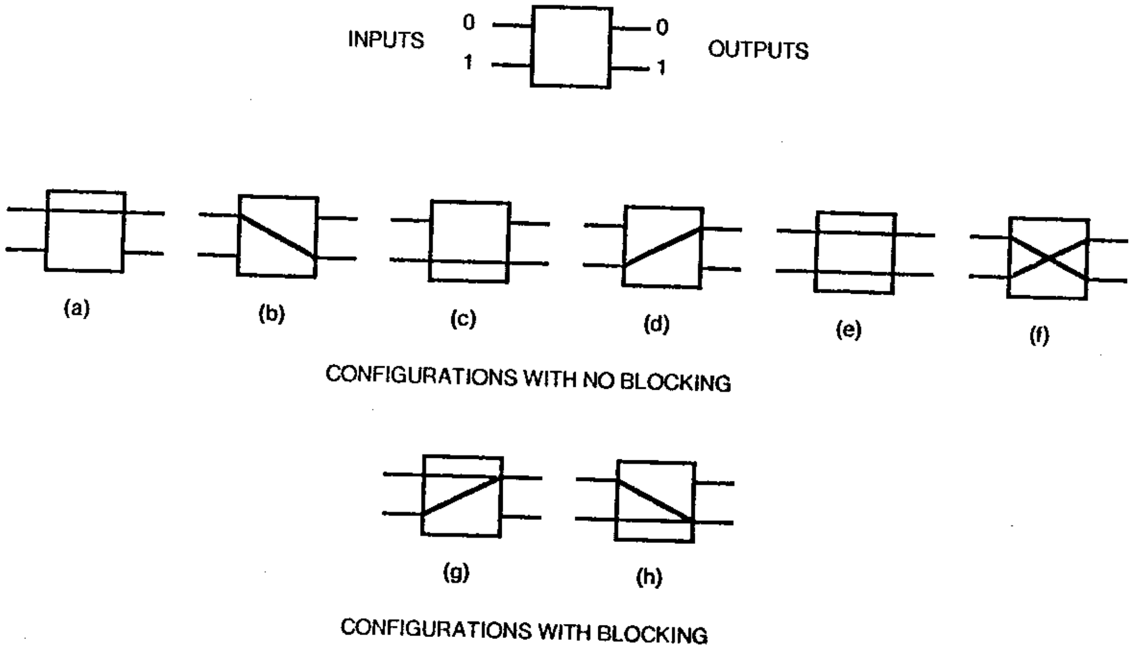


Figure 1. Port Configurations of the 2 × 2 Router

routers and $1 + \log_2 N$ levels of links. The ports of the routers in each stage are numbered from top to bottom starting with 0 at the top. If $P_s\ P_{s-1} \dots P_1\ P_0$ with $s = (\log_2 N) - 1$, is the bit representation of the port number, then the router number in that stage is given by the value of the bit string $P_s\ P_{s-1} \dots P_1$, i.e. by dropping the least significant bit of the port number. With this network structure, destination address bytes of the same value will route packets to the same output port of the network. The number of output ports that can be addressed, i.e. the network size, is fixed by the number of bits in the destination address byte. Port and router numbers are important for identifying ports and routers within a given stage during testing or fault diagnosis. The network is used for connecting N processors and the numbers in parentheses shown in Figure 2 are the processor numbers connected to the network ports.

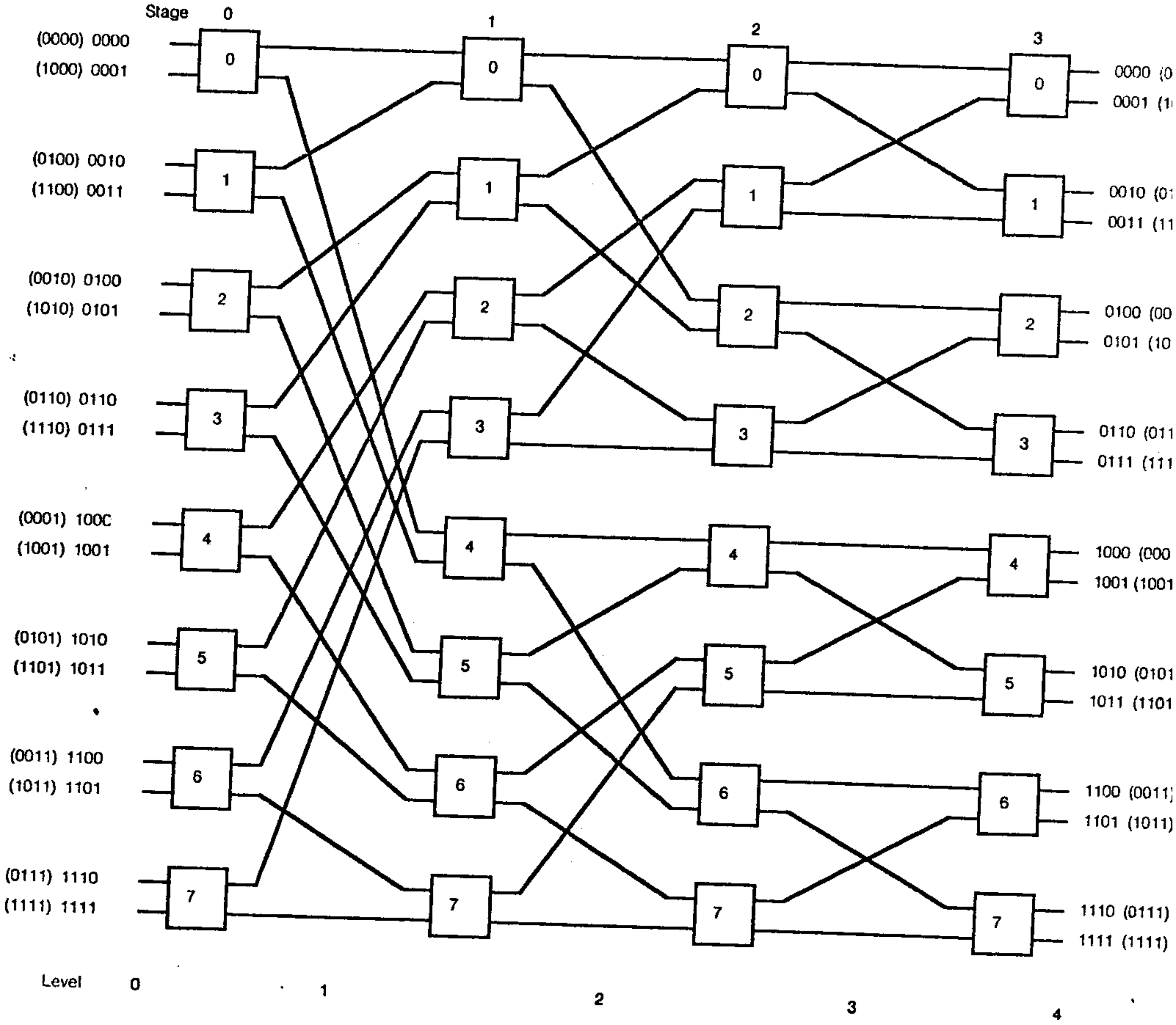


Figure 2. 16 × 16 Network of 2 × 2 Routers

4. Fault Model

One or more of the following effects will be produced by every single stuck-at fault occurring in a link or router of the network –

1) misdirected packets,

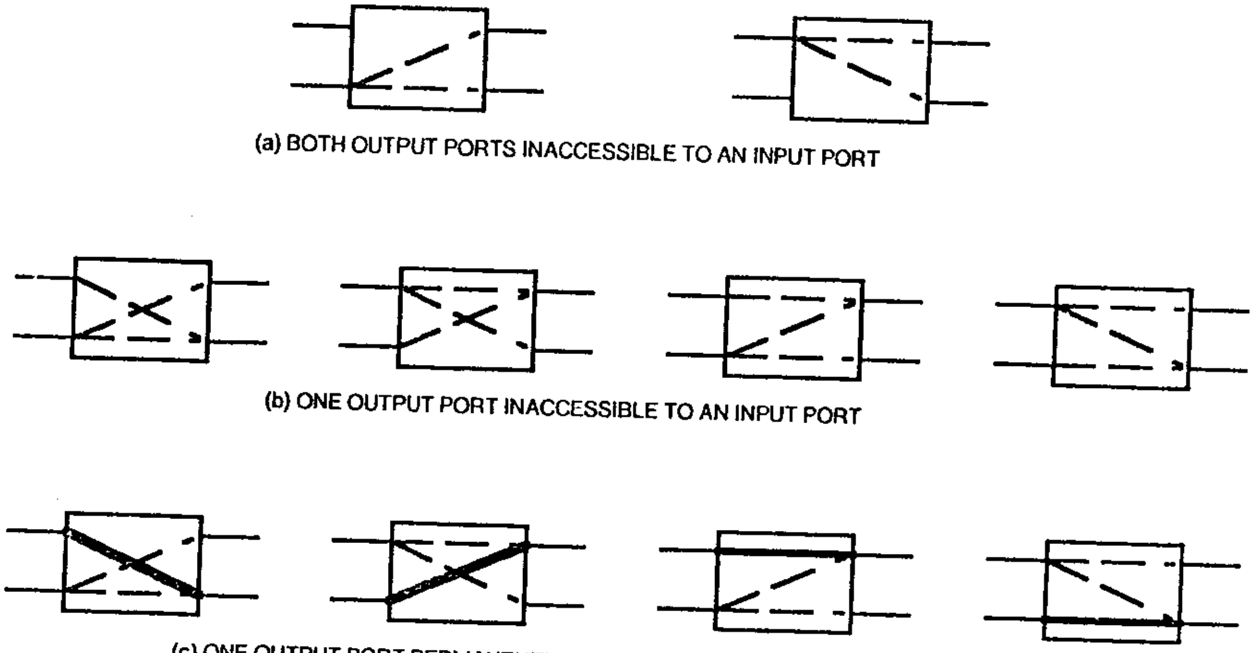
2) missing packets,

3) corrupted data in packets, and,

4) multiple packets being received.

The types of faults occurring in the network can be divided into two classes. The first of these is the class of faults that affects the asynchronous communication protocol. Examples of faults in this class include the packet acknowledge or packet control signals being stuck at one of the logical values or a switching element being stuck in some erroneous state due to a fault occurring inside it. The effect of this class of faults is missing packets, i.e. no packet is received when one or more is expected. This occurs when the packets fail to arrive within some specified time which is larger than the normal packet transmission time. The packets are held up somewhere in the network due to faults. The other class of faults affect the integrity or interpretation of data in a packet. A stuck-at fault occurring in a link, for example, can cause packets to be misdirected due to an erroneous destination address bit being used. Or, the occurrence of an internal fault in a switching element can cause the address bit to be interpreted wrongly. If the full address space available is not used, a fault in a link need not necessarily cause packets to be misdirected. We may get instead, erroneous data in the received packet.

Figure 3 shows the various faulty router configurations. The dashed lines indicate the connections that are operable while the dark lines indicate the connection being permanently fixed. Case (a) in the figure is for faults that prevent packet transmission through an input port while case (b) is for faults that prevent packet transmission through an input-output port pair. The third case is for faults that caused an input to be permanently connected to an output port. Note that a connection is said to be good if packets can be sent through using the asynchronous communication protocol. Hence the case of corrupted data in a received packet at the proper destination is not shown. Neither is the case where the faulty router sends packets to two output ports considered. This is because in order that the router be able to send packets to two output ports, the fault must make it behave like a fork in sending the packet arrival signal out and like a merge in receiving the packet acknowledge signals. We assume that the design of the router is such that this will cause packet transmission to hang. In one implementation of the router described in [2], this type of failure will never occur under the single stuck-at fault assumption.



(a) BOTH OUTPUT PORTS INACCESSIBLE TO AN INPUT PORT

(b) ONE OUTPUT PORT INACCESSIBLE TO AN INPUT PORT

(c) ONE OUTPUT PORT PERMANENTLY CONNECTED TO AN INPUT PORT

Figure 3. Faulty Configurations of the 2 × 2 Router

5. Test Strategy

The test strategy proposed here involves checking that packets can be sent through the input ports of every router in the network. To do this, the two test phase approach used in [8] is used. Since each phase involves a single test as each router is tested by the transmission of a single packet through it, we call the two phases, tests 1 and 2. In test 1, each router input port is checked to see if it can be connected directly to the corresponding output port – input port 0 to output port 0 and input 1 to output port 1. All routers are set up as in (e) of Figure 1 by using the proper destination addresses. Test 2 checks to see if each input port can be connected to the output port across from it – input port 0 to output port 1 and input port 1 to output 0. This means that all routers are set up as in (f) of Figure 1. Note that in both cases packet transmission through each input port of the router is independent of each other.

In each test, exactly one specially formatted packet is sent from a source to a destination and there are exactly N such source-destination pairs that will be communicating concurrently. Hence if the network is working properly, each processor will send and receive exactly one packet. The format of the packet used depends to some extent on the router implementation. In any case, source address is also included in the packet. The source address in the received packet is checked to make sure that the packet received is sent by the expected source. Some test bit patterns are also sent to check for stuck-at faults in the data bits. This test pattern is composed of two bytes; the first of which is an alternating sequence of 0's and 1's, while the second is the same sequence rotated by 1. The width of the test pattern is the same as the width of the data path of the byte serial transmission used. In those implementation where the Last Byte bit is used, the length of the packet is also included to check for stuck-at faults in that bit. If the packet length is fixed during the test, the length information need not be sent as data in the packet.

With this test strategy, if case (a) of Figure 3 occurs, the effect will be two missing packets – one for each test. In this case a stuck-at fault occurring in the attached input link of the router cannot be distinguished from one that occurs inside the router. Case (b) will have the effect of a missing packet in one of the tests. In case (c), the effect will be a missing packet and more than one packet received by a destination in one test. If a fault occurs that causes packets of the wrong lengths to be sent, the destination will see a shortened packet and it as well as some other destinations may receive additional packets.

5.1 Contention Tests

In addition to the above, each router must be checked to see that it functions properly in the blocked configurations (cases (g) and (h) of Figure 1). In these cases, each input port is sending a packet to the same output port at the same time. Only the packet from one of the input ports is allowed to go through while the packet at the other input port wait until the first input port is done. To ensure that no input port is left waiting indefinitely, the mechanism for arbitrating the contention must be fair. Hence if there is a packet pending at the unconnected input port, the packet will be sent through once the input port being served is done. A subsequent packet at the latter will not be served until after the pending input port is served. It is common to use a hardware arbiter to handle this. The performance of the arbiter is determined by a number of characteristics. One of these characteristics is the minimum temporal separation between the requests from the input ports for connection to the same output port. If the actual separation is less than this minimum, then the arbiter will treat the two requests as occurring at the same time. That is, a contention is detected by the arbiter. Hence the the minimum temporal separation between the requests for connection of contending input ports should be small to minimize the occurrence of contentions. This is necessary as it has been found in [1] that when contention occurs, the arbiter can take an arbitrarily long time to resolve the contention, thus slowing down packet transmission through the router. Another characteristic of the arbiter that is important is the mean resolution time – i.e. the time it takes the arbiter to resolve a contention. The smaller the mean resolution time the faster it takes the arbiter to resolve and hence the better is its performance. Determining such characteristics of the arbiter is extremely tedious as it involves controlling the difference in the arrival times of the requests to the order of tens of picoseconds! It is suggested that the measurements be done at installation time and not when the router is being used since setting up the elaborate measurement instruments would be very difficult when the router is in the routing network.

Another test that needs to be done is that of checking the integrity of a connection once it has been established. Under no circumstances must packets from another input port be able to prematurely break the connection between an input port and an output port. That is, once an input port is blocked, it must remain blocked until packet transmission through the previously established path has been completed. Each router is tested in the following manner. For each of the configurations (g) and (h) of Figure 1, each input port is connected in turn to the output port. A path is established by sending the address byte of a packet from an input port to an output port. Then transmission of the packet is suspended, without breaking the path, by holding the transmission of the next byte of the packet. Special hardware may have to be added to the router to do this properly. An attempt is then made to send a packet from the other input port to the same output port of the router. This packet transmission should be blocked by the router. The original suspended packet transmission is then continued. The packet received at the destination should not contain any bytes that belong to the blocked packet. Furthermore if the two packets are sent to the same destination, the blocked packet must be received latter than the unblocked one if no further blocking occurs in the rest of the transmission path through the network. The test is then repeated by reversing the roles of the two input ports, i.e. the one that is not blocked previously becomes the blocked one and vice versa. This test is done a total of four times, two for each of the configurations (g) and (h) of Figure 1, for each router. For the network, the procedure is to test a stage of routers at a time. This is done by establishing $\frac{N}{2}$ parallel paths through the network at one time. For example if stage 0 is being tested, packets are sent such that for each router in that stage, input port 0 is connected to output port 0. Then packets are sent from input port 1 to output port 0 of each router and the packets received at the destinations are checked to see that the packets from the odd numbered input ports are blocked and only the packets from the even numbered ports are received by their respective destinations. Half of the parallel paths used in test 1 are used – i.e. the ones for even numbered input ports. The test is then repeated by using the other half of test 1 – i.e. the paths for odd numbered input ports are established. Similarly the two halves of test 2 are used. Hence all four possible blocking combinations for each router in that stage are checked. This is repeated for all stages of the network – i.e. it is repeated a total of $\log_2 N$ times. This test should be done when the routing network successfully pass tests 1 and 2. This will insure that stuck-at faults that are detected earlier will not interfere with the test; since the test only checks the blocking capability of the routers.

6. Fault Diagnosis

If a fault is detected in the test, the fault diagnosis strategy described in [9] is used to identify the faulty router. However, it is important to note that the strategy given in [9] deals only with single bit input lines, while in this paper we are dealing with multiple signal lines carrying bytes observing some asynchronous communication protocol. Hence, instead of getting faulty output patterns, we get one of the effects described in Section 4. For example, the logically unidentified output value (open circuit) "$-$" and logically erroneous value (two independent logic signals being tied together) "φ" correspond to the effects of missing packets and multiple packets, respectively.

6.1 Both Output Ports Inaccessible to an Input Port

Since in this case a fault occurring in a link cannot be distinguished from one that occurs in the connected router, the fault is assumed to be in a link. Once the link is located, further tests are then done to locate the actual fault. Since a link is on exactly one path for each test, the set of links that are on the faulty path can be identified as follows. Each link is identified by the number of the input port that it is connected to. In test 1, if $P_s\ P_{s-1} \cdots P_1\ P_0$
is the link that is connected to the source processor then the link at the output side of the i-th stage is $P_0\ P_1 \cdots P_i\ P_s \cdots P_{i+1}$, where $0 \le i \le s$. Similarly for test 2, the link at the output side of the i-th stage is $\overline{P}_0\ \overline{P}_1 \cdots \overline{P}_i\ P_s \cdots P_{i+2}\ P_{i+1}$. In test 1, the link at the output side of the i-th stage is identified by rotating, to the right by 1, the rightmost $s-i+1$ bits of the link number of the previous stage while in test 2, the process is the same except that the least significant bit of the $s-i+1$ bits is complemented before rotating the bits to the right by 1. To identify the "faulty" link, the source addresses for the two tests are obtained from the destination addresses of the processors that did not receive a packet. Note that the source-destination addresses are related as follows: for test 1, the addresses are bit reversals of each other and for test 2 they are the complement of the bit reversals of each other. The set of links of the path is determined for each test. The "faulty" link is the intersection of the two link sets. Two tests are required for locating the "faulty" link and to determine if the fault is in the link or the router, one more test is necessary. This test involves checking to see if packet arrivals and packet acknowledgments can detected at the input port of the router. The absence of the former means that the link is bad and the absence of the latter means that the router is bad.

6.2 An Output Port Inaccessible to an Input Port

For this case, there is only one destination that receives no packets for both tests. To locate the faulty router, a binary search is done. The objective of the search is to identify the stage in which the router is located. Knowing the path and the stage, the router can be pinpointed. A search tree with each of the stages of the network as leaves is constructed. Starting at the root, the stages 0 to $\frac{s}{2}$ (if s is even) or $\frac{s-1}{2}$ (if s is odd) will be in the left subtree and the rest of the stages in the right subtree. The left subtree is set up to be of the same configuration as the test in which the fault occurs while the right subtree is set up in the same configuration as the other test. The network is then tested. If no faulty response is obtained then the fault is in the right subtree; otherwise it is in the left subtree. This process is repeated for the faulty subtree until the stage is located. The number of tests required is of the order log(log N).

6.3 An Output Port Permanently Connected to an Input Port

In this case one of the tests will give two faulty responses – missing packet and multiple packets at two distinct destinations. From the test at which the fault occurs, the fault type can be determined – for test 1, the left two cases of (c) in Figure 3 and for test 2 the other two cases. At most 2 tests are required to locate the router as this case is similar to that discussed in Section 6.1.

6.4 Erroneous Packet Length

For this case, the destination will receive a shorter than expected packet. Since the fault may occur in a link or a router, depending on whether the Last Byte bit is used or not, the situation is similar to that discussed in Section 6.1. A fault has the effect of sending fragments of the packets through the network. More than one destination may receive multiple packets; all but one of these will receive a normal packet followed by at least one erroneous packet. The remaining one destination will receive one shortened packet followed possibly by some erroneous packets. The faulty path is identified by the latter since it is the proper destination and is guaranteed to receive at least the destination address byte of the packet. Each test will give a faulty path and the intersection of the set of links or routers in the two paths is the faulty link or router.

7. Summary

A test strategy for packet switching networks has been presented. The strategy is developed for byte serial packet communication using an asynchronous communication protocol. It has been shown that the effect of a single stuck-at fault can be classified into misdirected packets, missing packets, corrupted data in packets, or multiple packets. There are basically two types of faults – those that prevent packet transmission and those that affect only the integrity or the interpretation of the data sent in the packet. The presence of a fault in the network will show up as one of 4 cases – both output ports of the switching element not accessible to an input port, an output port not accessible to an input port, an input port permanently connected to an output port and erroneous packet length. An approach for fault location is also presented and it is shown that the number of tests required is either constant or of the order of log (log N).

8. Acknowledgements

The author is indebted to Professor Jack Dennis as well as to Andy Boughton and Dean Brock for their criticisms and comments during the writing of this paper.

9. References

[1] T.J. Chaney and C.E. Molnar, "Anomalous Behavior of Synchronizer and Arbiter Circuits," *IEEE Transactions on Computers*, vol. C-22, no. 4, April 1973, 421-422.

[2] J. Dennis, G. Boughton, and C. Leung, "Building Blocks for Data Flow Prototypes," *Proceedings of 1980 Symposium on Computer Architecture*, LaBaule, France, May 1980, 1-8.

[3] J. Lilienkamp, "The Development of a Prototype Router: Design, Implementation, and Test Procedures," Laboratory for Computer Science, M.I.T., CSG Memo 199, September 1980.

[4] C. Mead and L. Conway, Introduction to VLSI Systems, Addison-Wesley, Reading, MA, 1980.

[5] J. Narraway and K-M. So, "Fault diagnosis in inter-processor switching networks," *Proceedings of the IEEE International Conference on Circuits and Computers*, ICCC 80, October 1980, 750-753.

[6] A. Tripathi and G. Lipovski, "Packet Switching in Banyan networks," *Proceedings of the 6th Annual Symposium on Computer Architecture*, April 1979, 160-167.

[7] C. Wu and T. Feng, "On a class on multistage interconnection networks," *IEEE Transactions on Computers*, vol. C-29, August 1980, 694-702; also published as "Routing techniques for a class of multistage interconnection networks," in the *Proceedings of the 1978 International Conference on Parallel Processing*, G. Lipovski, Editor, August 1978, 197-205.

[8] C. Wu and T. Feng, "The reverse-exchange interconnection network," *IEEE Transactions on Computers*, vol. C-29, September 1980, 801-811; also in the *Proceedings of the 1979 International Conference on Parallel Processing*, O. Garcia, Editor, August 1979, 160-174.

[9] C. Wu and T. Feng, "Fault-diagnosis for a class of multistage interconnection networks," *IEEE Transactions on Computers*, vol. C-30, October 1981, 743-758; also in the *Proceedings of the 1979 International Conference on Parallel Processing*, O. Garcia, Editor, August 1979, 269-278.