# Fault diagnosis of a distributed knockout switch

Y.-J.Cheng
T.-H.Lee
W.-Z.Shen

*Indexing terms: Distributed knockout switch, Fault diagnosis procedure, Switch element array*

**Abstract:** The distributed knockout switch has multiple paths between any input and output pair and thus is inherently robust to faults without the need of adding any additional switch elements. However, to achieve fault tolerance, one has to first detect and locate the faults. The authors present an efficient fault diagnosis procedure to detect, locate, and identify the fault type of single switch element faults for the switch element array of the distributed knockout switch. To facilitate fault diagnosis, the operation of switch elements is slightly modified. The diagnosis procedure can locate most single switch element faults in two phases. Faults which cannot be located in two phases can always be located in a third phase. Binary search algorithms are developed to locate some kinds of single switch element faults in the third phase.

## 1 Introduction

Fault tolerance is an important design issue of any ATM switching system to improve system reliability. Without fault tolerent capability, a single fault can be disastrous to a switching system. To achieve fault tolerance, one has to first detect and locate the faults. Various fault diagnosis procedures have recently been proposed to detect and locate faults for different switching networks [1–5], which are considered to be candidate architectures in ATM switching systems.

A fault diagnosis procedure for the switch element array of the distributed knockout switch [6] was presented in [5]. Only two types of switch element (SE) faults (i.e., cross-stuck (CS) and toggle-stuck (TS)) and two kinds of link stuck-at faults (i.e., horizontal-stuck (HS) and vertical-stuck (VS)) were considered. Unfortunately, an SE with a CS fault may not be detected and located using that procedure. For example, consider an $n \times n$ SE array. Let $SE(i, j)$ denote the SE in the $i$th row and the $j$th column. If $SE(i, j)$ suffers a CS fault where $i < n$ and $j < n$, $SE(i + 1, j + 1)$ will be in the cross state too, because in that diagnosis procedure, the priority level of the cell entering from the north side is higher than that of the cell entering from the west side. As a consequence, the CS fault in $SE(i, j)$ is corrected by $SE(i + 1, j + 1)$ and the output becomes fault free. Furthermore, the SE array has to be partitioned into $2^{P-1} \times 2^{P-1}$ blocks and diagnosed separately, where $P$ represents the number of priority bits.

In this paper, we modify the operation of SE to facilitate fault diagnosis. With the modification, the whole SE array can be diagnosed together. In other words, one does not need to partition the SE array into smaller blocks and diagnose each block separately.

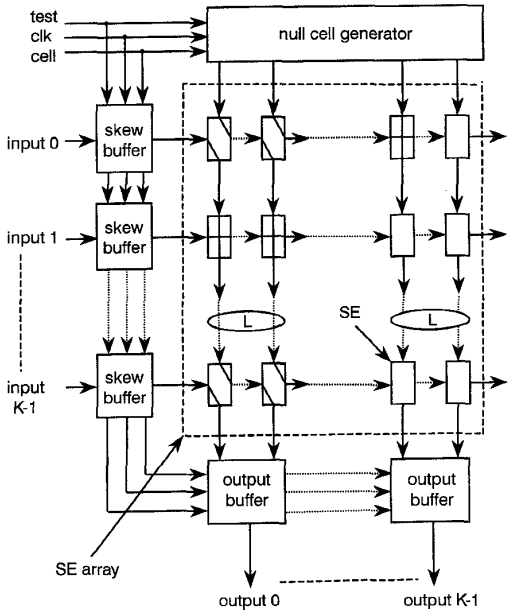


**Fig. 1** *Architecture of distributed knockout switch*

## 2 Operations of switch element

Fig. 1 shows the architecture of the distributed knockout switch proposed in [6]. Each SE can only be in the cross state or the toggled state. To facilitate fault diagnosis, we modify the operation of SE and the result is illustrated in Fig. 2, where $A_w$ and $P_w$ indicate, respectively, the address and the priority of cells input from the west side, $A_n$ and $P_n$ cells input from the north side. If $A_w \neq A_n$ or $P_w < P_n$, the SE is in the cross state and routes cells from the west side to the east side, and cells



*IEE Proceedings* online no. 19982136

Paper first received 12th March 1997 and in revised form 9th March 1998

Y.-J. Cheng is with the Applied Technology Laboratory, Telecommunication Laboratories, Chunghwa Telecom Co., Ltd., 12 Lane 551, Min-Tsu Road, Sec. 3 Yang-Mei, Taoyuan 326, Tawian, Republic of China, and also with the Institute of Electronics, National Chiao Tung University

T.-H. Lee is with the Department of Communication Enigneering, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 300, Taiwan, Republic of China

W.-Z Shen is with the Department of Electronic Engineering and Institute of Electronics, National Chiao Tung University

from the north side to the south side. If $A_w = A_n$ and $P_w \geq P_n$, the SE is in the toggled state and routes cells from the west side to the south side and cells from the north side to the east side. The modification is that an SE is set to be in the toggled state (rather than the cross state in the original design) when $A_w = A_n$ and $P_w = P_n$. With the modification, the input port in the lower position has a higher priority (i.e. if the cells of two input ports have the same priority values and are to be routed to the same output port, the SE array favours the cell of the lower position input port). The operations are so modified that one can easily set all SEs to be in the toggled state. A faulty SE with a CS fault can be detected and located in one phase after the modification.

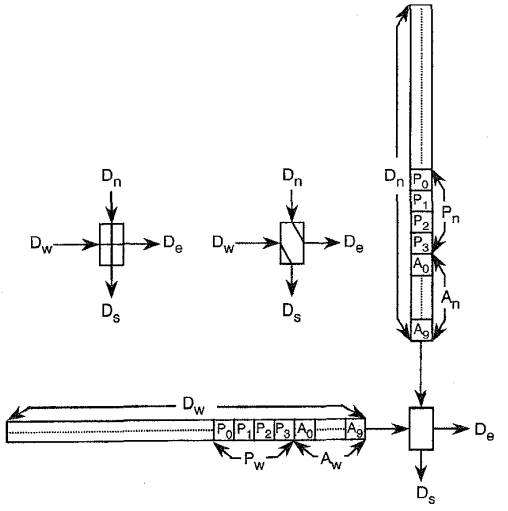


**Fig. 2** *Functional diagram of SE*
Cross state: $A_w \neq A_n$ or $P_w < P_n$
Toggled state: $A_w = A_n$ and $P_w \geq P_n$

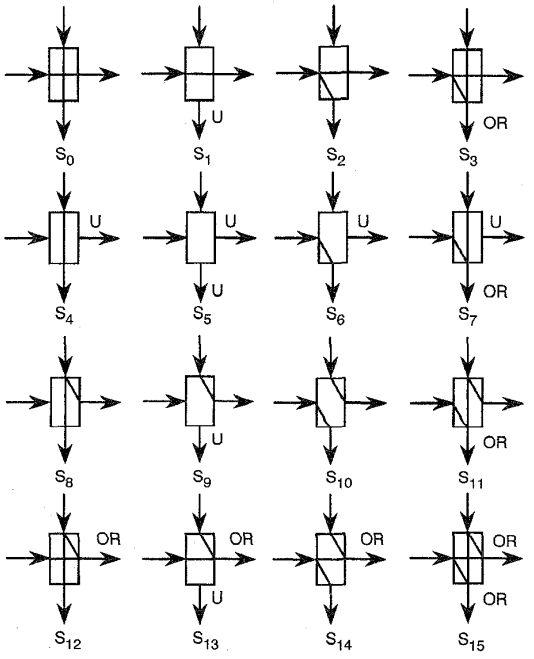


**Fig. 3** *Possible states of SE*
$U = s-a-0$ or $s-a-1$; OR = wire-OR output
$S_0$–$S_{15}$ represent states 0–15, respectively

## 3 Fault model

In this Section, we describe the fault model investigated in this paper. In general, there are 16 possible states for each SE, which are illustrated in Fig. 3. In this Figure, the output of an unidentified fault (U) can be either 0 or 1 (depending on circuit implementation) but is assumed to be constant. The 'OR' signal denotes logically bitwise wire-OR output. For an SE, only the cross state ($S_0$) and the toggled state ($S_{10}$) are valid states. Other states are invalid states. A faulty SE can be in any one of the 16 possible states. Thus, for an SE with two valid states, there are 256 possible state combinations. As in [4], we use the set $\{(s_1, s_2)|s_i \in S\}$ to describe the state combinations and refer to each $(s_1, s_2)$-pair as a functional state. Assume that the first valid state is $S_0$ and the second valid state is $S_{10}$. As a consequence, only the state combination $(S_0, S_{10})$ is the normal functional state and the other 255 state combinations are faulty functional states.

The 16 states shown in Fig. 3 can be partitioned into four sets, $B_1 = \{S_0, S_{10}\}$, $B_2 = \{S_2, S_8\}$, $U = \{S_1, S_4, S_5, S_6, S_9\}$, and $O = \{S_3, S_7, S_{11}, S_{12}, S_{13}, S_{14}, S_{15}\}$. Sets $B_1$ and $B_2$ contain states that result in only binary faults. Set $B_1$ contains states that suffer from a CS or TS fault. Set $B_2$ contains states that result in broadcast from the west or the north side. Set $U$ contains states that result in at least one unidentified fault which may be s-a-0 or s-a-1 but no wire-OR fault. Set $O$ contains states that result in at least one wire-OR fault.

## 4 Test vectors design

The test structure of a $K \times K$ switch with $L$ links per output port is shown in Fig. 4, where there is a test cell generator at each input port and a fault detector at each output port. We consider only single SE faults.

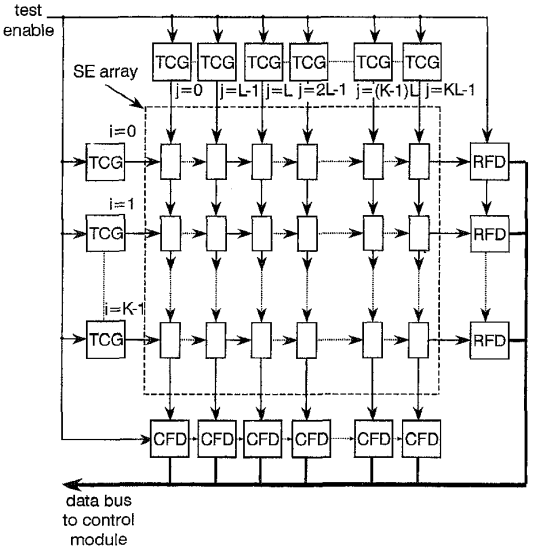


**Fig. 4** *Testing structure of a $K \times K$ with $L$ links per output port*
TCG = test cell generator
RFD = row fault detector
CFD = column fault detector

To diagnose, a test enable signal is sent to the skew buffers and the null cell generator. The skew buffers and the null cell generator then send cells to the SE array. The fault detectors at row and column outputs detect the fault and send the test results to the control

module, which analyses the test results to determine which SE is faulty. Before presenting the tests and the fault diagnosis procedure in details, we define the following notations. The data fields in phase I and phase II are slightly different. In phase I, it contains only one value while in phase II, it is partitioned into two parts, A and B.

$K$ = number of input/output ports.

$L$ = number of links per output port.

$RA_i$ = value of address field for the test cell entering from the $i$th row.

$RP_i$ = value of priority field for the test cell entering from the $i$th row.

$RD_i$ = value of data field for the test cell entering from the $i$th row in test I.

$RDA_i$ = value of part A of data field for the test cell entering from the $i$th row in test II.

$RDB_i$ = value of part B of data field for the test cell entering from the $i$th row in test II.

$CA_j$ = value of address field for the test cell entering from the $j$th column.

$CP_j$ = value of priority field for the test cell entering from the $j$th column.

$CD_j$ = value of data field for the test cell entering from the $j$th column in test I.

$CDA_j$ = value of part A of data field for the test cell entering from the $j$th column in test II.

$CDB_j$ = value of part B of data field for the test cell entering from the $j$th column in test II.

The purpose of test vector I is to verify all SEs in the cross state. To achieve this, the input cells are selected as follows.

### 4.1 Test I

(i) For $i = 0, 1, ..., K - 1$

Set $RA_i = LK + i$, $RP_i = 0$, and $RD_i = 2LK(i + 1)$

(ii) For $j = 0, 1, ..., LK - 1$

Set $CA_j = j$, $CP_j = 1$, and $CD_j = j + 1$

Notice that the address fields of the cells applied to column inputs are chosen to be different from those applied to row inputs. As a result, all SEs should be in the cross state if the SE array is fault free. Also, test I requires $\lceil \log_2((L + 1)K) \rceil$ bits in the address field where $\lceil x \rceil$ represents the smallest integer greater than or equal to $x$. The data field is used to locate and identify the fault type. The number of bits for the data field must be greater than or equal to $\lceil \log_2(K + 1) \rceil + \lceil \log_2(LK + 1) \rceil$.

The purpose of test II is to verify all SEs in the toggled state. To achieve this, the test cells are selected as follows.

### 4.2 Test II

(i) For $i = 0, 1, ..., K - 1$

Set $RA_i = 1$, $RP_i = 0$, $RDA_i = LK + i$, and $RDB_i = K - 1 - i$

(ii) For $j = 0, 1, ..., LK - 1$

Set $CA_j = 1$, $CP_j = 0$, $CDA_j = K - 1 - j$, and $CDB_j = K + j$

In test II, the address and priority fields of cells applied to column inputs are chosen to be the same as those applied to row inputs. Consequently, all SEs are supposed to be in the toggled state if the SE array is fault free. The data field is divided into two parts for identifying the faulty state. For the cells entering from the rows, part $A$ of the data fields ($DA$) are selected to be in the ascending order, while part $B$ ($DB$) are chosen to be in the descending order. The two parts of the data fields of the cells entering from the columns also have this property.

## 5 Fault diagnosis procedure

To diagnose a single fault, all SEs must be tested in both the cross state and the toggled state. Therefore, there are at least two phases in our diagnosis procedure. In phase I, all SEs are set in the cross state ($S_0$) using the test cells designed in test I. Table 1 summarises the results of phase I test, assuming $SE(a, b)$ is faulty. Notice that the faulty state can always be identified and the faulty SE can be located unless the faulty state is $S_1$ or $S_4$. If the faulty state is $S_4$ ($S_1$), then one knows which row (column) the faulty SE is in.

**Table 1: Results of phase I diagnosis, assuming $SE(a, b)$ is faulty**

| Faulty state | $RO_a$ | $CO_b$ |
|---|---|---|
| ⊕ $S_0$ | ≡ | ≡ |
| ‡ $S_1$ | ≡ | $CD_b$ = **0** or **1** |
| $S_2$ | ≡ | $CD_b = 2LK(a + 1)$ |
| $S_3$ | ≡ | $CD_b = (2LK(a + 1)) \vee (b + 1)$ |
| † $S_4$ | $RD_a$ = **0** or **1** | ≡ |
| $S_5$ | $RD_a$ = **0** or **1** | $CD_b$ = 0 or 1 |
| $S_6$ | $RD_a$ = **0** or **1** | $CD_b = 2LK(a + 1)$ |
| $S_7$ | $RD_a$ = **0** or **1** | $CD_b = (2LK(a + 1)) \vee (b + 1)$ |
| $S_8$ | $RD_a = b + 1$ | ≡ |
| $S_9$ | $RD_a = b + 1$ | $CD_b$ = 0 or 1 |
| $S_{10}$ | $RD_a = b + 1$ | $CD_b = 2LK(a + 1)$ |
| $S_{11}$ | $RD_a = b + 1$ | $CD_b = (2LK(a + 1)) \vee (b + 1)$ |
| $S_{12}$ | $RD_a = (2LK(a + 1)) \vee (b + 1)$ | ≡ |
| $S_{13}$ | $RD_a = (2LK(a + 1)) \vee (b + 1)$ | $CD_b$ = 0 or 1 |
| $S_{14}$ | $RD_a = (2LK(a + 1)) \vee (b + 1)$ | $CD_b = 2LK(a + 1)$ |
| $S_{15}$ | $RD_a = (2LK(a + 1)) \vee (b + 1)$ | $CD_b = (2LK(a + 1)) \vee (b + 1)$ |

∨ bitwise OR.
⊕ fault-free state.
≡ the output value is the same as fault-free value.
† only knows faulty SE is in the $a$th row (no information about which column the faulty SE is in).
‡ only knows faulty SE is in the $b$th column (no information about which row the faulty SE is in).
**0** an all-zero vector; **1** an all-one vector.

We now briefly explain the results summarised in Table 1. Let $RO_i$ and $CO_j$ denote the outputs received at row $i$ and column $j$, respectively. Also, let $FFRO_i$ and $FFCO_j$ denote the outputs received at row $i$ and column $j$ for a fault-free SE array. Let **0** indicate an all-zero vector and **1** indicate an all-one vector. Remember we partitioned (in Section 3) the 16 possible states into four sets $B_1$, $B_2$, $U$, and $O$. If $s_1 \in B_1$, the result is either fault free (if $s_1 = S_0$) or $CO_b$ and $RO_a$ are exchanged (if $s_1 = S_{10}$). Suppose $s_1 \in B_2$. If $s_1 = S_2$, $CO_b = FFRO_a$. Similarly, if $s_1 = S_8$, $RO_a = FFCO_b$. Consider the case $s_1 \in U$. In this case, either one or two **0**s or **1**s are received at the outputs. If $s_1 = S_1$ ($S_4$),

$CO_b$ = **0** or **1** ($RO_a$ = **0** or **1**). All the other outputs are fault free. If $s_1 = S_5$, $RO_a = CO_b$ = **0** or **1**. If $s_1 = S_6$ ($S_9$), $RO_a$ = **0** or **1** ($CO_b$ = **0** or **1**) and $CO_b = FFRO_a$ ($RO_a = FFCO_b$). Finally, assume $s_1 \in O$. In this case, either one or two wire-OR values are received at the outputs. If $s_1 = S_7$ ($S_{13}$), $RO_a$ = **0** or **1** ($CO_b$ = **0** or **1**) and the $b$th column ($a$th row) receives a wire-OR output. If $s_1 = S_3$ ($S_{12}$), only the $b$th column ($a$th row) receives a wire-OR output and the $a$th row ($b$th column) is fault free. The wire-OR value of data field depends on the row number and column number in our test vector design. Thus, according to the wire-OR output, one knows the location of the faulty SE. If $s_1 = S_{11}$ ($S_{14}$), the $b$th column ($a$th row) receives a wire-OR output and $RO_a = FFCO_b$ ($CO_b = FFRO_a$). If $s_1 = S_{15}$, both the $a$th row and the $b$th column receive wire-OR outputs.

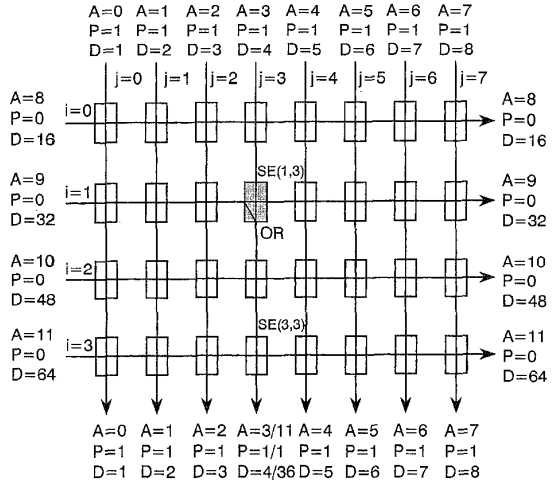


**Fig.5** *Phase I diagnosis for a 4 × 4 switch with L = 2: result of faulty state $S_3$*
OR = wire-OR output

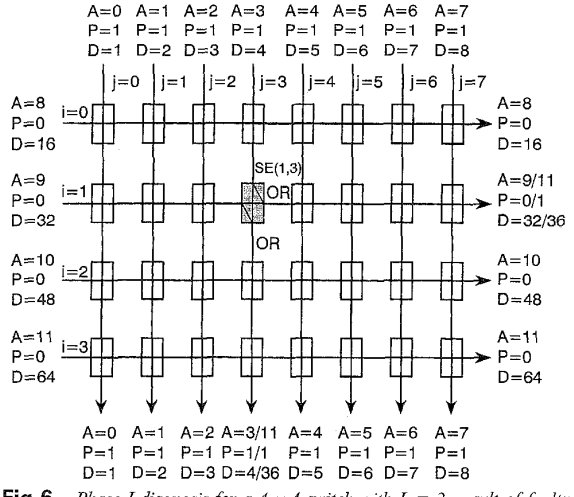


**Fig.6** *Phase I diagnosis for a 4 × 4 switch with L = 2: result of faulty state $S_{15}$*
OR = wire-OR output

Examples of phase I diagnosis for a 4 × 4 switch with $L$ = 2 are illustrated in Figs. 5 and 6 for faulty states $S_3$ and $S_{15}$, respectively. In these Figures, $SE(1, 3)$ is assumed to be faulty. In Fig. 5, $D$ = 4/36 in the output of the third column means that the fault-free value of the data field is 4 and the faulty value becomes 36. Notice that the faulty value of the data field of the third column depends on the row number of the faulty SE. Although the value of the address field after wire-OR ($A$ = 11) is the same as the address value of the test vector entering from the third row, the priority value after wire-OR is greater than its priority value. Therefore, $SE(3, 3)$ remains in the cross state and the wire-OR vector can be propagated to the output. In Fig. 6, the wire-OR vector can be propagated to the output. Therefore, by observing the values of data field received by the output ports, one can determine which row and column the faulty SE is in.

In phase II diagnosis, all SEs are set in the toggled state ($S_{10}$) using the test cells designed in test II. There are $LK \times K$ SEs for a $K \times K$ switch with $L$ links per output port. We define a special diagonal, denoted by DIAG, to be the set of SEs such that $SE(i, j) \in$ DIAG if and only if (iff) $j - i = LK - K$. Three regions are considered separately below.

### 5.1 Region 1: j - i = LK - K

Table 2 shows the results for the case $SE(i, j) \in$ DIAG. If $s_2 \in B_1$, the result is either fault free (if $s_2 = S_{10}$) or $RO_{K-1} = FFCO_{LK-1}$ and $CO_{LK-1} = FFRO_{K-1}$ (if $s_2 = S_0$). Suppose $s_2 \in B_2$. If $s_2 = S_2$, $RO_{K-1} = FFCO_{LK-1}$ and all other outputs are fault free. Similarly, if $s_2 = S_8$, then $CO_{LK-1} = FFRO_{K-1}$. Consider the case $s_2 \in U$. In this case, either one or two **0**s or **1**s are received by the outputs. If $s_2 = S_1$, $CO_b$ = **0** or **1** because the address field in the test vector was set to 1 which is different from that in **0** or **1**. Also, we have $RO_{K-1} = FFCO_{LK-1}$ because the link of the east side receives the cell entering from the west side and the address field is not changed. In addition, there are $K - a - 1$ outputs which receive vectors different from the fault-free vectors. Notice that the number of those outputs depends on the row number of the faulty SE. Therefore, one can determine which row the faulty SE is in. The situation for $s_2 = S_4$ is similar to that for $s_2 = S_1$. If $s_2 = S_5$, $RO_a = CO_b$ = **0** or **1**. If $s_2 = S_6$, $RO_a$ = **0** or **1**. In addition, there are $LK - b - 1$ row outputs which receive vectors different from the fault-free vectors. Since the number of those outputs depends on the column number of the faulty SE, one knows its location. The situation for $s_2 = S_9$ is similar to that for $s_2 = S_6$. Finally, assume $s_2 \in O$. In this case, either one or two wire-OR vectors are received by the outputs. If $s_2 = S_7$, $RO_a$ = **0** or **1** and the ($LK$ − 1)th column receives a wire-OR output because the address field is not changed. To detect the fault of a wire-OR output, the data field is partitioned into two parts. The values of part A and part B in the data field are assigned in the ascending order and the descending order, respectively. Therefore, the value of wire-OR output is changed in either part A or part B. The situation for $s_2 = S_{13}$ is similar to that for $s_2 = S_7$. If $s_2 = S_3$, the ($LK$ − 1)th column receives a wire-OR output and $RO_{K-1} = FFCO_{LK-1}$. The situation for $s_2 = S_{12}$ is similar to that for $s_2 = S_3$. If $s_2 = S_{11}$, only the ($LK$ − 1)th column receives a wire-OR output. All the other outputs are fault free. The situation for $s_2 = S_{14}$ is similar to that for $s_2 = s_{11}$. If $s_2 = S_{15}$, both the ($K$ − 1)th row and the ($LK$ − 1)th column receive wire-OR output vectors.

**Table 2: Results of phase II diagnosis, assuming $SE(a, b)$ is faulty and $b - a \neq LK - K$**

| Faulty state | $RO_a$ | $RO_{K-1}$ | $CO_b$ | $CO_{LK-1}$ |
|---|---|---|---|---|
| * $S_0$ | ≡ | $RDA_{K-1} = K$<br>$RDB_{K-1} = LK - 1$ | ≡ | $CDA_{LK-1} = K - 1$<br>$CDB_{LK-1} = LK$ |
| $S_1$ | ≡ | $RDA_{K-1} = K$<br>$RDB_{K-1} = LK - 1$ | **0** or **1** | NO = $K - a - 1$ |
| * $S_2$ | ≡ | $RDA_{K-1} = K$<br>$RDB_{K-1} = LK - 1$ | ≡ | ≡ |
| * $S_3$ | ≡ | $RDA_{K-1} = K$<br>$RDB_{K-1} = LK - 1$ | ≡ | $CDA_{LK-1} = K \vee (K - 1)$<br>$CDB_{LK-1} = (LK - 1) \vee LK$ |
| $S_4$ | **0** or **1** | NO = $LK - b - 1$ | ≡ | $CDA_{LK-1} = K - 1$<br>$CDB_{LK-1} = LK$ |
| $S_5$ | **0** or **1** | x | **0** or **1** | x |
| $S_6$ | **0** or **1** | NO = $LK - b - 1$ | ≡ | ≡ |
| $S_7$ | **0** or **1** | NO = $LK - b - 1$ | ≡ | $CDA_{LK-1} = K \vee (K - 1)$<br>$CDB_{LK-1} = (LK - 1) \vee LK$ |
| * $S_8$ | ≡ | ≡ | ≡ | $CDA_{LK-1} = K - 1$<br>$CDB_{LK-1} = LK$ |
| $S_9$ | ≡ | ≡ | **0** or **1** | NO = $K - a - 1$ |
| ⊕ $S_{10}$ | ≡ | ≡ | ≡ | ≡ |
| * $S_{11}$ | ≡ | ≡ | ≡ | $CDA_{LK-1} = K \vee (K - 1)$<br>$CDB_{LK-1} = (LK - 1) \vee LK$ |
| * $S_{12}$ | ≡ | $RDA_{K-1} = K \vee (K - 1)$<br>$RDB_{K-1} = (LK - 1) \vee LK$ | ≡ | $CDA_{LK-1} = K - 1$<br>$CDB_{LK-1} = LK$ |
| $S_{13}$ | ≡ | $RDA_{K-1} = K \vee (K - 1)$<br>$RDB_{K-1} = (LK - 1) \vee LK$ | **0** or **1** | NO = $K - a - 1$ |
| * $S_{14}$ | ≡ | $RDA_{K-1} = K \vee (K - 1)$<br>$RDB_{K-1} = (LK - 1) \vee LK$ | ≡ | ≡ |
| * $S_{15}$ | ≡ | $RDA_{K-1} = K \vee (K - 1)$<br>$RDB_{K-1} = (LK - 1) \vee LK$ | ≡ | $CDA_{LK-1} = K \vee (K - 1)$<br>$CDB_{LK-1} = (LK - 1) \vee LK$ |

x – don't care for these output values because the faulty SE can be located.
NO – number of outputs which receive a vector different from the fault-free ones.
* providing information about a set of SEs in a diagonal.

### 5.2 Region 2: $j - i > LK - K$

Table 3 shows the results for the case $SE(i, j)$ in the right-hand side of DIAG. If $s_2 \in B_1$, the result is either fault free (if $s_2 = S_{10}$) or $RO_{LK+a-b-1} = FFRO_{LK+a-b}$ and $RO_{LK+a-b} = FFRO_{LK+a-b-1}$ (if $s_2 = S_0$). Suppose $s_2 \in B_2$. If $s_2 = S_2$, $RO_{LK+a-b-1} = FFRO_{LK+a-b}$ and all other outputs are fault free. Similarly, if $s_2 = S_8$, $RO_{LK+a-b} = FFRO_{LK+a-b-1}$. Consider the case $s_2 \in U$. In this case, either one or two **0**s or **1**s are received at the outputs. If $s_2 = S_1$, then $CO_b = \mathbf{0}$ or **1** because the address field in the test vector was set to 1 which is different from that in **0** or **1**. Also, we have $RO_{LK+a-b-1} = FFRO_{LK+a-b}$ because the link of the east side receives the cell entering from the west side and the address field is not changed. In addition, there are $K - a - 1$ outputs which receive vectors different from the fault-free ones. Again, the number of those outputs depends on the row number of the faulty SE and, thus, one can determine which row the faulty SE is in. The situation for $s_2 = S_4$ is similar to that for $s_2 = S_1$. If $s_2 = S_5$, then $RO_a = CO_b = \mathbf{0}$ or **1**. If $s_2 = S_6$, then $RO_a = \mathbf{0}$ or **1**. In addition, there are $LK - b - 1$ row outputs which receive vectors different from the fault-free vectors. Based on the number of outputs which receive vectors different from the fault-free ones, one knows the location of the faulty SE. The situation for $s_2 = S_9$ is similar to that for $s_2 = S_6$. Finally, assume $s_2 \in O$. In this case, either one or two wire-OR vectors are received by the outputs. If $s_2 = S_7$, $RO_a = \mathbf{0}$ or **1** and the $(LK + a - b)$th row receives a wire-OR output because the address field is not changed. The situation for $s_2 = S_{13}$ is similar to that for $s_2 = S_7$. If $s_2 = S_3$, the $(LK + a - b)$th row receives a wire-OR output and $RO_{LK+a-b-1} = FFRO_{LK+a-b}$. The situation for $s_2 = S_{12}$ is similar to that for $s_2 = S_3$. If $s_2 = S_{11}$, only the $(LK + a - b)$th row receives a wire-OR output. All the other outputs are fault free. The situation for $s_2 = S_{14}$ is similar to that for $s_2 = S_{11}$. If $s_2 = S_{15}$, both the $(LK + a - b - 1)$th row and the $(LK + a - b)$th row receive wire-OR output vectors.

### 5.3 Region 3: $j - i < LK - K$

Table 4 shows the results for the case $SE(i, j)$ in the left-hand side of DIAG. If $s_2 \in B_1$, the result is either fault free (if $s_2 = S_{10}$) or $CO_{K-a+b-1} = FFCO_{K-a+b}$ and $CO_{K-a+b} = FFCO_{K-a+b-1}$ (if $s_2 = S_0$). Suppose $s_2 \in B_2$. If $s_2 = S_2$, $CO_{K-a+b-1} = FFCO_{K-a+b}$ and all other outputs are fault free. Similarly, if $s_2 = S_8$, $CO_{K-a+b} = CO_{K-a+b-1}$. Consider the case $s_2 \in U$. In this case, either one or two **0**s or **1**s are received by the outputs. If $s_2 = S_1$, then $CO_b = \mathbf{0}$ or **1** because the address field in the test vector was set to 1 which is different from that in **0** or **1**. Also, we have $CO_{K-a+b-1} = FFCO_{K-a+b}$ because the link of the east side receives the cell entering from the west side and the address field is not changed. In addition, there are $K - a - 1$ outputs which receive vectors different from the fault-free vectors. Based on the number of those outputs, one can determine which row

**Table 3: Results of phase II diagnosis, assuming $SE(a, b)$ is faulty and $b - a > LK - K$**

| Faulty state | $RO_a$ | $RO_{LK+a-b-1}$ | $RO_{LK+a-b}$ | $CO_b$ |
|---|---|---|---|---|
| * $S_0$ | ≡ | ROT0 | ROT1 | ≡ |
| $S_1$ | ≡ | ROT0 | NO = $K - a - 1$ | **0** or **1** |
| * $S_2$ | ≡ | ROT0 | ≡ | ≡ |
| * $S_3$ | ≡ | ROT0 | ROT3 | ≡ |
| $S_4$ | **0** or **1** | NO = $LK - b - 1$ | ROT1 | ≡ |
| $S_5$ | **0** or **1** | x | x | **0** or **1** |
| $S_6$ | **0** or **1** | NO = $LK - b - 1$ | ≡ | ≡ |
| $S_7$ | **0** or **1** | NO = $LK - b - 1$ | ROT3 | ≡ |
| * $S_8$ | ≡ | ≡ | ROT1 | ≡ |
| $S_9$ | ≡ | ≡ | NO = $K - a - 1$ | **0** or **1** |
| ⊕ $S_{10}$ | ≡ | ≡ | ≡ | ≡ |
| * $S_{11}$ | ≡ | ≡ | ROT3 | ≡ |
| * $S_{12}$ | ≡ | ROT2 | ROT1 | ≡ |
| $S_{13}$ | ≡ | ROT2 | NO = $K - a - 1$ | **0** or **1** |
| * $S_{14}$ | ≡ | ROT2 | ≡ | ≡ |
| * $S_{15}$ | ≡ | ROT2 | ROT3 | ≡ |

ROT0: row output type 0
$RDA_{LK+a-b-1} = LK + a - b - 1/LK + a - b$ and
$RDB_{LK+a-b-1} = K - a + b/K - a + b - 1$
ROT1: row output type 1
$RDA_{LK+a-b} = LK + a - b/LK + a - b - 1$ and
$RDB_{LK+a-b} = K - a + b - 1/K - a + b$
ROT2: row output type 2
$RDA_{LK+a-b-1} = LK + a - b - 1/(LK + a - b - 1) \vee (LK + a - b)$ and
$RDB_{LK+a-b-1} = K - a + b/(K - a + b) \vee (K - a + b - 1)$
ROT3: row output type 3
$RDA_{LK+a-b} = LK + a - b/(LK + a - b) \vee (LK + a - b - 1)$ and
$RDB_{LK+a-b} = K - a + b - 1/(K - a + b - 1) \vee (K - a + b)$

**Table 4: Results of phase II diagnosis, assuming $SE(a, b)$ is faulty and $b - a < LK - K$**

| Faulty state | $RO_a$ | $CO_b$ | $CO_{K-a+b-1}$ | $CO_{K-a+b}$ |
|---|---|---|---|---|
| * $S_0$ | ≡ | ≡ | COT1 | COT0 |
| $S_1$ | ≡ | **0** or **1** | COT1 | NO = $K - a - 1$ |
| * $S_2$ | ≡ | ≡ | COT1 | ≡ |
| * $S_3$ | ≡ | ≡ | COT1 | COT2 |
| $S_4$ | **0** or **1** | ≡ | NO = $LK - b - 1$ | COT0 |
| $S_5$ | **0** or **1** | **0** or **1** | x | x |
| $S_6$ | **0** or **1** | ≡ | NO = $LK - b - 1$ | ≡ |
| $S_7$ | **0** or **1** | ≡ | NO = $LK - b - 1$ | COT2 |
| * $S_8$ | ≡ | ≡ | ≡ | COT0 |
| $S_9$ | ≡ | **0** or **1** | ≡ | NO = $K - a - 1$ |
| ⊕ $S_{10}$ | ≡ | ≡ | ≡ | ≡ |
| * $S_{11}$ | ≡ | ≡ | ≡ | COT2 |
| * $S_{12}$ | ≡ | ≡ | COT3 | COT0 |
| $S_{13}$ | ≡ | **0** or **1** | COT3 | NO = $K - a - 1$ |
| * $S_{14}$ | ≡ | ≡ | COT3 | ≡ |
| * $S_{15}$ | ≡ | ≡ | COT3 | COT2 |

COT0: column output type 0
$CDA_{K-a+b-1} = LK + a - b/LK - b - 1$ and
$CDB_{K-a+b-1} = K - a + b - 1/K - a + b$
COT1: column output type 1
$CDA_{K-a+b} = LK + a - b - 1/LK + a - b$ and
$CDB_{K-a+b} = K - a + b/K - a + b - 1$
COT2: column output type 2
$CDA_{K-a+b-1} = LK + a - b/(LK + a - b) \vee (LK + a - b - 1)$ and
$CDB_{K-a+b-1} = K - a + b - 1/(K - a + b - 1) \vee (K - a + b)$
COT3: column output type 3
$CDA_{K-a+b} = LK + a - b - 1/(LK + a - b - 1) \vee (LK - a + b)$ and
$CDB_{K-a+b} = K - a + b/(K - a + b) \vee (K - a + b - 1)$

the faulty SE is in. The situation for $s_2 = S_4$ is similar to that for $s_2 = S_1$. If $s_2 = S_5$, $RO_a = CO_b = \mathbf{0}$ or **1**. If $s_2 = S_6$, then $RO_a = \mathbf{0}$ or **1**. Besides, there are $LK - b - 1$ outputs which receive vectors different from the fault free ones. Thus, according to the number of outputs which receive vectors different from the fault-free vectors, one knows the location of the faulty SE. The situation for $s_2 = S_9$ is similar to that for $s_2 = S_6$. Finally, assume $s_2 \in O$. In this case, either one or two wire-OR vectors are received by the outputs. If $s_2 = S_7$, $RO_a = \mathbf{0}$ or **1** and the $(K - a + b)$th column receives a wire-OR output because the address field is not changed. The situation for $s_2 = S_{13}$ is similar to that for $s_2 = S_7$. If $s_2 = S_3$, the $(K - a + b)$th column receives a wire-OR output and $CO_{K-a+b-1} = FFCO_{K-a+b}$. The situation for $s_2 = S_{12}$ is similar to that for $s_2 = S_3$. If $s_2 = S_{11}$, only the $(K - a + b)$th column receives a wire-OR output. All the other outputs are fault free. The situation for $s_2 = S_{14}$ is similar to that for $s_2 = S_{11}$. If $s_2 = S_{15}$, both the $(K - a + b - 1)$th column and the $(K - a + b)$th column receive a wire-OR output value.

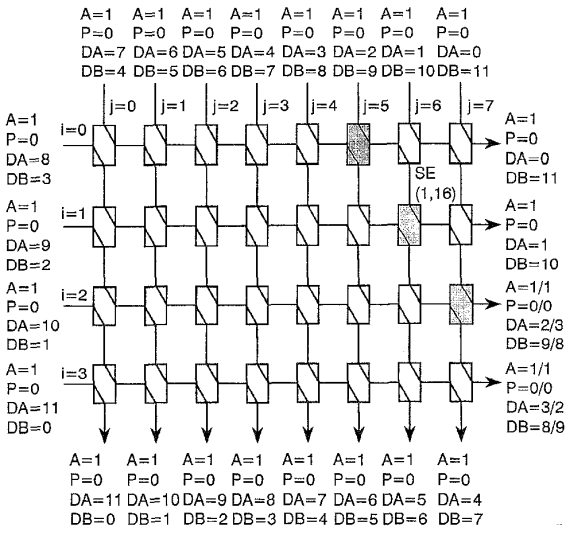


**Fig. 7** *Phase II diagnosis for a 4 × 4 switch with $L = 2$: result of faulty state $S_0$*
*$U = s - a - 0$ or $s - a - 1$*

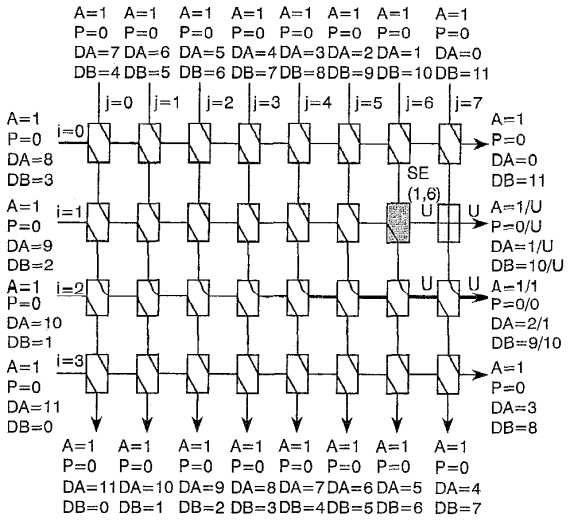


**Fig. 8** *Phase II diagnosis for a 4 × 4 switch with $L = 2$: result of faulty state $S_6$*
*$U = s - a - 0$ or $s - a - 1$*

Examples of phase II diagnosis for a 4 × 4 switch with $L = 2$ are illustrated in Figs. 7 and 8 for states $S_0$, and $S_6$, respectively. In these Figures, $SE(1, 6)$ is assumed to be faulty. The results in these figures can be obtained from Table 3 if one sets $a = 1$ and $b = 6$. For example, if the faulty state is $S_{11}$, the value of $RDB_3$ is changed from fault free value 8 to 9. In Fig. 7, the fault can be detected and one knows which diagonal the faulty SE is in. In Fig. 8, the row number can be easily located from the **0** or **1**. The column number can also be located, because the number of outputs which receive vector different from the fault-free ones is equal to 1 excluding the vector of row 1 ($LK - b - 1 = 1$ and $LK = 8$, thus $b = 6$).

From the above results, one only knows which row or column the faulty SE is in if $(s_1, s_2) \in \{(S_1, S_{10}), (S_4, S_{10})\}$. If $s_1 = S_0$ and $s_2 \in \{S_0, S_2, S_3, S_8, S_{11}, S_{12}, S_{14}, S_{15}\}$, the test results only provide information about a set of diagonal SEs.

If $(s_1, s_2) = (S_1, S_{10})$, one knows which column the faulty SE is in. To obtain the row number, a binary search algorithm is necessary. Let area 1 include the SEs from row 0 to row $[(K - 1)/2]$ and area 2 include the SEs from row $[(K - 1)/2] + 1$ to row $K - 1$ where $[x]$ represents the largest integer smaller than or equal to $x$. All SEs in area 1 are set in the cross state and all SEs in area 2 are set in the toggled state. To accomplish this, the addresses of test cells entering from rows in area 1 and area 2 are set to 0 and 1, respectively. Since the addresses of all test cells entering from columns are set to 1. All SEs in area 1 are set in the cross state and all SEs in area 2 are set in the toggled state. If the faulty SE is in area 1, one of the outputs receives a **0** or **1**, and area 1 is searched again. Otherwise, all outputs receive fault free vectors and area 2 is searched again. By iterating the binary search procedure, one can eventually locate the faulty SE.

Similarly, if $(s_1, s_2) = (S_4, S_{10})$, one knows which row the faulty SE is in. To obtain the column number, another binary search algorithm is necessary. Let area 1 include the SEs from column 0 to column $[(LK - 1)/2]$ and area 2 include the SEs from column $[(LK - 1)/2] + 1$ to column $LK - 1$. All SEs in area 1 are set in the cross state and all SEs in area 2 are set in the toggled state. To accomplish this, the addresses of test cells entering from columns in area 1 and area 2 are set to 0 and 1, respectively. Since the addresses of all test cells entering from rows are set to 1, SEs in area 1 and area 2 are set to the cross state and the toggled state, respectively. If the faulty SE is in area 1, an output will receive a **0** or **1**. If the faulty SE is in area 2, all outputs receive fault-free vectors. Again, by iterating the binary search procedure, one can eventually locate the faulty SE.

To locate the diagonal SEs, a third binary search algorithm is developed. First, the diagonal SEs are partitioned into two areas. Assume the faulty SE (i.e. $SE(a, b)$ is in a diagonal which covers column $L$ to column $U$). Let area 1 contain the SEs from column $L$ to $[(L + U)/2]$ and area 2 contain the SEs from column $[(L + U)/2] + 1$ to $U$. All SEs in area 1 are set in the cross state and all SEs in area 2 are set in the toggled state. To accomplish this, the addresses of test cells entering from column in area 1 and area 2 are set to 0 and 1, respectively. Since the addresses of all test cells entering from rows are set to 1, SEs in area 1 and area 2 are set to the cross state and the toggled state, respectively. If the faulty SE is in area 1, all outputs receive fault-free vectors. If the faulty SE is in area 2, at least an output receives a vector which is different from the fault-free vector. Based on the results, one knows which area the faulty SE is in. By iterating the procedure, one can determine the location of the faulty SE.

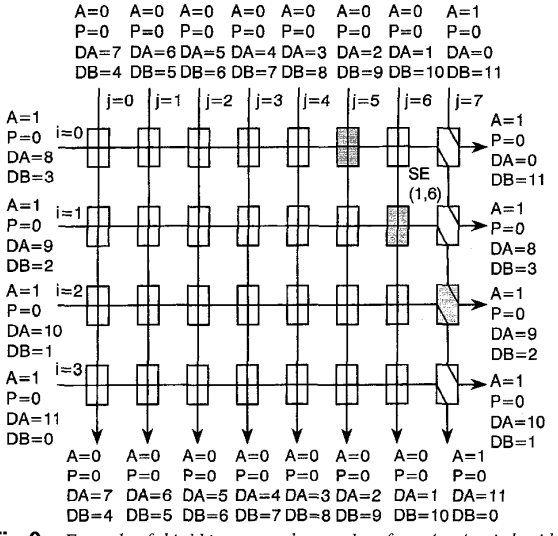


**Fig. 9** *Example of third binary search procedure for a 4 × 4 switch with L = 2: iteration 1*

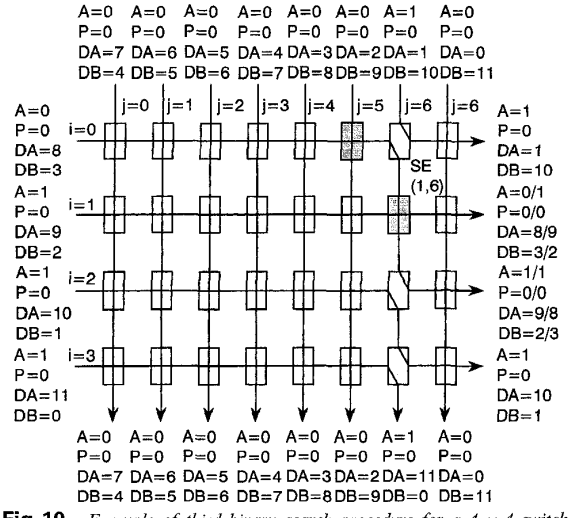


**Fig. 10** *Example of third binary search procedure for a 4 × 4 switch with L = 2: iteration 2*

An example for applying the third binary search algorithm to a 4 × 4 switch with $L = 2$ is illustrated in Figs. 9 and 10. We assume that the faulty SE is $SE(1, 6)$ and the faulty state is $s_2 = S_0$. The outputs of the second row and the third row are exchanged when all SEs are set in the toggled state. Therefore, the SE array will be searched from column 5 to column 7 using the third binary search algorithm. In the first iteration, shown in Fig. 9, area 1 contains the SEs from column 5 to column 6 and area 2 contains the SEs of column 7. The SEs in area 1 are set in the cross state and the SEs in area 2 are set in the toggled state. In this iteration, all outputs receive fault-free vectors. Therefore, one knows the faulty SE is in area 1. In the second iteration, shown in Fig. 10, the SEs in area 1

(column 5) are set in the cross state and the SEs in area 2 (column 6) are set in the toggled state. Since the outputs of the second row and the third row are exchanged, one knows the faulty SE is in area 2 (column 6).

Notice that, in the above fault diagnosis procedure, the time complexity of phase I and phase II is constant. For those (10 out of 255) faulty functional states that require the help of a binary search to locate the faulty SE, it takes $\lceil \log_2 K \rceil$ iterations, where $K$ is the size of switch and $\lceil x \rceil$ represents the smallest integer greater than or equal to $x$.

## 6 Conclusions

In this paper we have presented an efficient fault diagnosis procedure to detect, locate, and identify the fault type of single SE faults for the distributed knockout switch. The operations of SE is slightly modified to facilitate fault diagnosis. Most single faults can be detected, located, and identified in two phases. Binary search algorithms are required if the faulty SE cannot be located in two phases. Further research can be focused on diagnosing multiple faults and/or different fault models.

## 7 References


1 THANAWASTIEN, S., and NELSON, V.P.: 'Testing and fault-tolerance of multistage interconnection network', *IEEE Computer*, 1982, **15**, pp. 41–53

2 FENG, T.Y., and KAO, I.P.: 'On fault-diagnosis of some multistage interconnection networks'. 1982 international conference on *Parallel processing*, 1982, pp. 99–101

3 THANAWASTIEN, S., and NELSON, V.P.: 'Optimal fault detection test sequences for shuffle/exchange networks'. FTCS 13th international symposium Fault-tolerant computing, 1983, pp. 442–445

4 LEE, T.H., and CHOU, J.J.: 'Diagnosis of single faults in bitonic sorters', *IEEE/ACM Trans. Netw.*, 1994, **2**, (5), pp. 497–509

5 CHOE, B.S., and CHAO, H.J.: 'Fault tolerance of a large-scale multicast output buffered ATM switch'. IEEE INFOCOM'94, 1994, (San Francisco, CA),pp. 1456–1464

6 CHAO, H.J.: 'A recursive modular terabit/second ATM switch', *IEEE J. Select. Areas Commun.*, 1991, **9**, (8), pp. 1161–1172

7 CHENG, Y.J., LEE, T.H., and SHEN, W.Z.: 'Design and performance evaluation of a distributed knockout switch with input and output buffers', *IEE Proc. Commun.*, 1996, **143**, (3), pp. 149–154