

# 多故障的电子设备系统诊断设计方法研究

何 丽, 彭黎丽

(西南电子设备研究所, 成都 610036)

**摘要:** 针对复杂电子设备在使用过程中多故障诊断难的问题, 以相关性模型为基础, 提出了一种多故障诊断的设计方法: 首先, 根据相关性模型确定测试 D 矩阵, 并获得组成单元与测试次数的关系矩阵; 其次, 根据每次实际测试的结果构建测试结果矩阵, 并计算获得测试故障单元矩阵; 最后, 通过综合测试故障单元矩阵与测试次数关系矩阵的数据, 确定故障单元的概率, 根据故障概率定位出故障单元; 通过实例验算表明: 多故障诊断方法可准确定位故障单元, 大幅降低误修率。

**关键词:** 多故障; 相关性模型; 测试性 D 矩阵; 故障诊断; 测试性

## Design Method of Electronic System Diagnosis with Multiple Faults

HE Li, PENG Lili

(Southwest Institute of Electronic Equipment, Chengdu 610036, China)

**Abstract:** In order to solve the problems of Multiple fault diagnosis in complex electronic equipment, a design method of multiple fault diagnosis based on correlation model is proposed. Firstly, the testability D matrix is determined according to the correlation model, and the relevance matrix indicates the relationship between component unit and test times. Secondly, the test result matrix is constructed according to the actual test results, and the test fault unit matrix is calculated. Finally, through synthesizing the data of test failure unit matrix and test times relation matrix, the probability of failure unit is determined, and the failure unit is located according to the failure probability. The example shows that the multiple fault diagnosis method can accurately locate the fault unit and greatly reduce the invalid maintenance activities.

**Keywords:** multiple fault; relevance model; testability D matrix; diagnosis; testability

### 0 引言

现代战争的特点要求武器电子设备能快速反应, 快速投入使用, 一旦出现故障, 设备能及时准确地报告故障部位和故障危害性, 以便维修人员迅速排除故障, 提高设备的战备完好性。由此可见, 武器电子设备是否能及时准确地进行自身故障的检测和隔离对于设备的可用性有着至关重要的作用。因此, 在电子设备的研制阶段, 对其内部的故障诊断进行规划和设计成为了一项重要工作。

目前, 在电子设备的设计阶段通常按照 GJB2547A《装备测试性工作通用要求》规定的项目和方法开展相关的诊断设计, 主要通过测试性建模来优化电子设备的测试性设计, 确定电子设备的诊断流程。测试性模型为相关性模型, 是将电子设备的组成单元和测试点之间、以及两个组成单元之间或两个测试点之间的逻辑关系用模型表现出来<sup>[1]</sup>。利用测试性模型来进行测试性设计特别适用于大型复杂的电子设备, 可通过模型理清设备内部的复杂交联关系, 降低设计分析的难度<sup>[2]</sup>。

在利用相关性模型开展故障诊断设计分析时, 对测试做了如下假设:

1) 状态唯二假设。即被测电子设备仅有正常及故障两

种状态: 正常状态下设备无故障可以正常工作; 故障状态下设备不能正常工作;

2) 单故障假设。即在任何时刻当电子设备处于故障状态时, 认为电子设备只发生一项故障;

3) 测试对测试点的复用假设。即一项测试可以利用一个和多个测试点, 一个测试点也可被一个或多个测试利用;

4) 测试有效性等同假设。即某一组成单元发生了故障, 在信号流可达的各个测试点上, 测量有效性都是一样的。

由此可见, 在设计阶段, 设备的故障诊断设计是在单故障假设前提下进行的, 然而在实际工程中, 并非所有的故障都是单个发生的, 很多情况下是多故障随机组合或者耦合出现<sup>[3]</sup>。以单故障的假设前提完成的设计使得设备在后续使用过程中会出现检测隔离不准确形成伪故障, 导致故障定位错误等, 引发设备的无效维修活动, 降低设备的使用效率。

因此, 如何适应多故障模式下的准确检测和隔离成为了复杂电子设备测试性设计的新需求。本文针以相关性模型为基础, 提出了多故障发生情况下的诊断流程设计思路, 并通过实例验证了新方法的可行性和有效性。

收稿日期: 2021-05-07; 修回日期: 2021-06-07。

作者简介: 何 丽(1978-), 女, 四川成都人, 大学本科, 高级工程师, 主要从事电子装备测试性设计、健康管理及故障预测技术方向的研究。

引用格式: 何 丽, 彭黎丽. 多故障的电子设备系统诊断设计方法研究[J]. 计算机测量与控制, 2021, 29(8): 7-12.

## 1 多故障的诊断设计方法

本文提出的多故障的诊断设计方法是基于相关性模型, 扩展获得测试次数矩阵, 并根据每次实际测试的情况确定测试结果矩阵, 得到测试故障单元及其发生概率, 最终形成故障隔离清单<sup>[4-10]</sup>。

具体的设计方法及步骤如下:

第一步, 确定系统的相关性模型;

1) 按照划分的系统功能和组成单元, 画出系统的功能框图;

2) 在功能框图上表明功能信号流向和各组成单元的连接关系;

3) 按照系统的测试性设计标注系统的测试点位置和编号, 表明各组成单元与各测试点的相关性关系, 形成系统的相关性模型。

图 1 所示为一个系统相关性模型, 其中方框代表了系统的各个功能单元, 圆圈代表了系统内部设计的测试点, 箭头表明了功能信息传递的方向<sup>[11]</sup>。

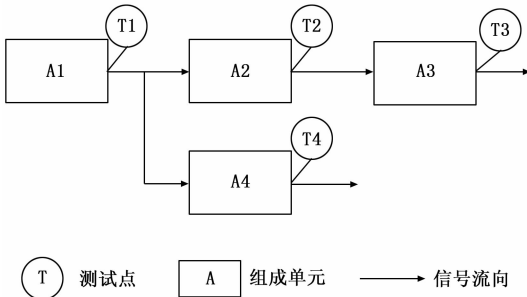


图 1 系统相关性模型示例

第二步, 确定组成单元与测试之间的相关性  $D$  矩阵;

根据多信号流图, 梳理出测试集、组成单元集, 并将组成单元和测试的依赖关系用矩阵表示出来, 即为相关性  $D$  矩阵<sup>[12-15]</sup>。

相关性  $D$  矩阵包括有:

1) 测试集  $T = \{T_1, T_2, T_3, \dots, T_i, \dots, T_m\}$ ;

2) 组成单元集  $A = \{A_1, A_2, A_3, \dots, A_j, \dots, A_n\}$ ;

3) 依赖矩阵  $D = [D_{ij}]_{m \times n}$ , 其中  $D_{ij}$  表示  $T_i$  与  $A_j$  的依赖关系, 当  $T_i$  能检测出  $A_j$  的时候,  $D_{ij}$  的值取为 1, 当  $T_i$  无法检测出  $A_j$  的时候,  $D_{ij}$  的值取为 0。

$$D_{m \times n} = \begin{bmatrix} D_{11} & \cdots & D_{1n} \\ \vdots & \ddots & \vdots \\ D_{m1} & \cdots & D_{mn} \end{bmatrix} \quad (1)$$

第三步, 根据多信号流模型和相关性  $D$  矩阵确定组成单元与系统内部故障诊断测试次数的关系矩阵:

测试次数矩阵为  $Tnum = [Tnum_j]_n$ , 其中  $Tnum_j$  表示在系统相关性模型中组成单元  $A_j$  可被测试合格的测试次数。  $Tnum_j$  的计算公式如下, 其中  $D_{ij}$  来自组成单元和测试的相关性  $D$  矩阵。

$$Tnum_j = \sum_{i=1}^m D_{ij} \quad (2)$$

第四步, 按照某次测试的实际测试结果, 构建某次测试的结果相关矩阵  $RD$ ;

测试结果相关矩阵  $RD$  包括有:

1) 测试结果集  $TR = \{TR_1, TR_2, TR_3, \dots, TR_i, \dots, TR_m\}$ ; 其中,  $TR_i$  为测试  $T_i$  在本次测试中的实际结果, 当测试  $T_i$  正常时,  $TR_i$  取值为 1, 当测试  $T_i$  异常时,  $TR_i$  取值为 0;

2) 组成单元集  $A = \{A_1, A_2, A_3, \dots, A_j, \dots, A_n\}$ ;

3) 测试结果矩阵  $RD = [RD_{ij}]_{m \times n}$ , 如下:

$$RD_{m \times n} = \begin{bmatrix} RD_{11} & \cdots & RD_{1n} \\ \vdots & \ddots & \vdots \\ RD_{m1} & \cdots & RD_{mn} \end{bmatrix} \quad (3)$$

其中:  $RD_{ij}$  表示测试结果  $TR_i$  与组成单元  $A_j$  的映射关系, 映射关系为:

$$RD_{ij} = TR_i \times D_{ij} \quad (4)$$

其中:  $D_{ij}$  为组成单元和测试的依赖矩阵的取值, 数据来自相关性  $D$  矩阵。由此可见, 当测试  $T_i$  与单元  $A_j$  无依赖关系时,  $RD_{ij}$  固定为 0; 当测试  $T_i$  与单元  $A_j$  有依赖关系, 且测试  $T_i$  正常, 即  $TR_i$  为 1 时,  $RD_{ij}$  才会为 1, 否则为 0。

第五步, 根据测试结果相关矩阵  $RD$ , 确定本次测试的故障单元矩阵:

故障单元矩阵为  $A\_RD = [A\_RD_j]_n$ , 其中,  $A\_RD_j$  是组成单元  $A_j$  的测试结果, 其计算公式如下  $RD_{ij}$  为测试结果相关矩阵中的值, 当  $m$  个测试的结果中, 只要有 1 个  $RD_{ij}$  为 1,  $A\_RD_j$  即为 1, 即表示对应的  $A_j$  测试正常,  $A\_RD_j$  为 0 的对应的  $A_j$  即为非正常状态。

$$A\_RD_j = V_{i=1}^m RD_{ij} \quad (5)$$

第六步, 根据测试故障单元矩阵和故障诊断测试次数矩阵来确定故障单元发生概率矩阵:

故障单元发生概率矩阵为  $P = [P_j]_n$ , 其中  $P_j$  为组成单元  $A_j$  本次测试的故障概率数值, 计算公式如下,  $P_j$  数值越大的表示本次测试对应的组成单元  $A_j$  故障的可能性越大,  $P_j$  为 0 的表示本次测试对应的组成单元  $A_j$  无故障。

$$P_j = \frac{A\_RD_j \times Tnum_j}{\sum_{i=1}^m A\_RD_j \times Tnum_i} \quad (6)$$

通过步骤 1~6 的分析和计算, 即可实现对复杂系统多故障情况下的故障定位, 其中, 步骤 1~3 涉及各矩阵的元素值是与系统测试性设计相关的, 当系统设计完毕后, 相关的矩阵元素值便可固化; 步骤 4~6 涉及各矩阵的元素值依据单次实际测试结果计算可得。

## 2 算法验证

为验证多故障诊断设计方法的可行性, 在某信号采集电子设备中, 采用传统的诊断设计方法和多故障的诊断设计方法开展了对比验证。

### 2.1 验证系统

某信号采集电子设备组成原理如图 2 所示, 设备由放

大前端、开关、变频单元、频率综合器、接收处理、信号处理、主控计算机和自检源组成。

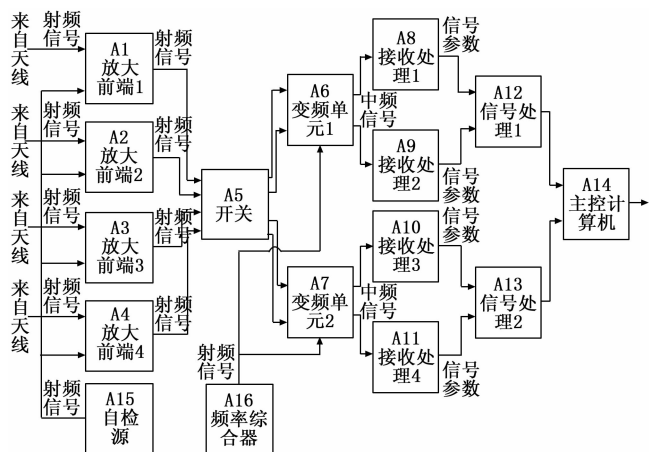


图 2 某信号采集电子设备原理框图

信号采集电子设备接收来自天线的射频信号, 经放大前端进行放大后, 通过开关切换至不同的变频单元下变为中频信号, 接收处理对中频信号进行分选采集获得信号参数, 并送至信号处理进行融合, 融合后的信号参数上报主控计算机再进行融合后显示输出。

在开展测试性设计时, 信号采集电子设备内部设计了自检源, 可为设备的自检提供标准的测试信号, 同时在信号采集电子设备内部设计了多个测试点, 以便进行故障的检测和隔离, 详细情况如表 1 所示。

表 1 某信号采集电子设备测试设计汇总表

测试	位置	测试设计情况
T1	A1	输出射频信号状态检测
T2	A2	输出射频信号状态检测
T3	A3	输出射频信号状态检测
T4	A4	输出射频信号状态检测
T5	A8	A1 支路射频信号参数检测
T6	A9	A2 支路射频信号参数检测
T7	A10	A3 支路射频信号参数检测
T8	A11	A4 支路射频信号参数检测
T9	A12	A1A2 支路融合参数检测
T10	A13	A3A4 支路融合参数检测
T11	A14	所有支路融合参数检测

## 2.2 验证系统相关性模型

根据信号采集电子设备的组成、信号流关系以及设计的测试, 梳理形成信号采集电子设备的相关性模型, 如图 3 所示。

在相关性模型中,  $A = \{A1, A2, \dots, A16\}$  为信号采集电子设备的组成单元,  $T = \{T1, T2, \dots, T11\}$  为信号采集电子设备内部设计的测试。

根据验证系统的相关性模型确定组成单元与测试之间的相关性  $D$  矩阵如图 4 所示, 其中  $D_{ij} = 1$  的表示  $T_i$  项测试可以检测确定  $A_j$  组成单元是否故障,  $D_{ij} = 0$  的表示  $T_i$  项

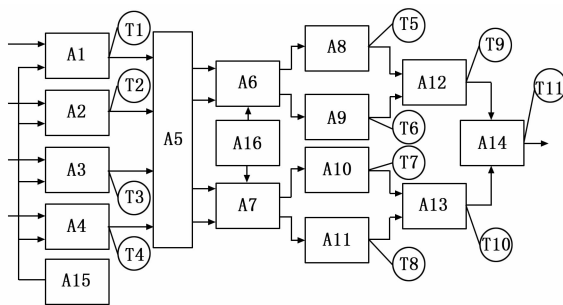


图 3 验证系统相关性模型

测试无法检测确定  $A_j$  组成单元是否故障<sup>[16-18]</sup>。

D	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16
T1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
T4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
T5	1	0	0	0	1	1	0	1	0	0	0	0	0	0	0	1
T6	0	1	0	0	1	1	0	0	1	0	0	0	0	0	0	1
T7	0	0	1	0	1	0	1	0	0	1	0	0	0	0	0	1
T8	0	0	0	1	1	0	1	0	0	0	1	0	0	0	0	1
T9	1	1	0	0	1	1	0	1	1	0	0	1	0	0	0	1
T10	0	0	1	1	1	0	1	0	0	1	1	0	1	0	0	1
T11	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

图 4 验证系统组成单元与测试相关性  $D$  矩阵

## 2.3 验证系统故障假设

由于本设计方法主要是针对设备发生多故障情况时的应用, 因此在验证中, 设定某时刻信号采集电子设备的 A6 和 A10 同时发生了故障, 导致电子设备内部出现多故障情况。

根据图 2 可知, A6 和 A10 发生故障后, 将影响到 T5、T6、T7、T9、T10、T11 项的测试, 各测试点的测试情况如表 2 所示。

表 2 A6 和 A10 故障时刻设备测试情况

测试	测试结果情况
T1	正常
T2	正常
T3	正常
T4	正常
T5	异常
T6	异常
T7	异常
T8	正常
T9	异常
T10	异常
T11	异常

## 2.4 按照传统单故障诊断设计方法的设计

按照传统的单故障诊断设计方法, 在基于单故障发生的假设前提下, 依据验证系统组成单元与测试相关性  $D$  矩阵可梳理形成电子设备的图形化故障诊断树, 如图 5 所示。

图形化故障诊断树由方框、测试标志、连线、数字 0/1 组成。方框表示基于单元（或单元故障）的故障诊断中间结果和故障诊断最终结果；测试标志为“Tn”，表示所用的测试（或测试点）；连线用于表达测试执行次序或者测试跳转关系；数字 0/1 为测试结果标志，0 表示测试未通过，1 表示测试通过。

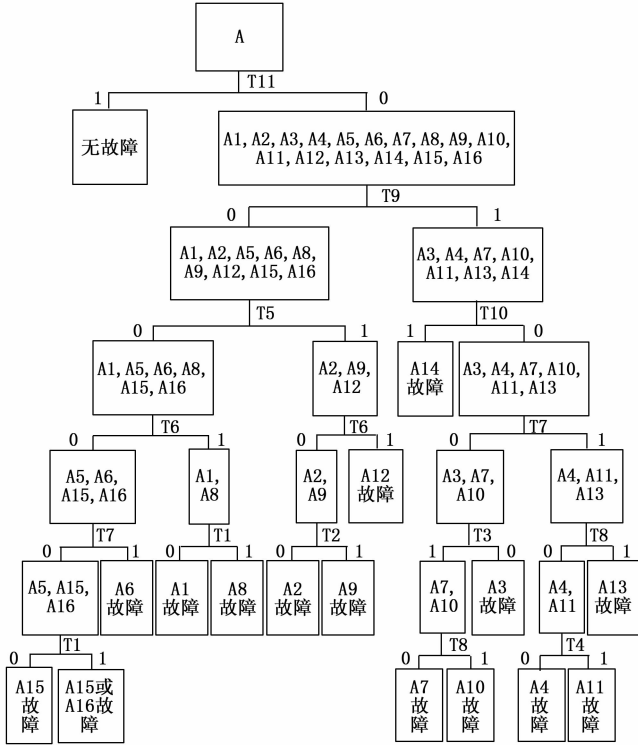


图 5 传统诊断方法的故障诊断树

按照故障单元假设情况，此时刻 A6 和 A10 同时发生故障，因此，测试点 T1~T11 的测试结果  $T = \{1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0\}$ ，测试结果带入诊断树，最终将故障隔离定位到 A5 或 A16 故障，与实际发生故障的情况不一致。如果根据测试结果对 A5 或者 A16 进行维修，故障无法消除，将持续报 A5 或 A16 故障，导致诸多无效维修活动。

### 2.5 按照多故障诊断设计方法的设计

采用多故障诊断设计方法进行设计时，在验证系统组成单元与测试相关性 D 矩阵的基础上，根据信号流程图确定组成单元与测试次数之间的相关矩阵，如图 6 所示。

$$T_{nm} = \begin{bmatrix} A1 & A2 & A3 & A4 & A5 & A6 & A7 & A8 & A9 & A10 & A11 & A12 & A13 & A14 & A15 & A16 \\ 4 & 4 & 4 & 4 & 7 & 4 & 4 & 3 & 3 & 3 & 3 & 2 & 2 & 1 & 11 & 7 \end{bmatrix}$$

图 6 组成单元与测试次数相关性 Tnm 矩阵

按照假设情况，此时刻 A6 和 A10 同时发生故障，因此，各测试点的测试结果为  $TR = \{1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0\}$ ，获得测试结果相关矩阵 RD 如图 7 所示。

RD	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16
TR1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
TR2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
TR3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
TR4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
TR5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR8	0	0	0	1	1	0	1	0	0	0	1	0	0	0	1	1
TR9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

图 7 组成单元与测试结果相关性 RD 矩阵

根据测试结果相关性矩阵计算得到故障单元矩阵如图 8 所示，并依据此矩阵进一步计算获得故障单元发生概率矩阵，如图 9 所示。

$$A_{RD} = \begin{bmatrix} A1 & A2 & A3 & A4 & A5 & A6 & A7 & A8 & A9 & A10 & A11 & A12 & A13 & A14 & A15 & A16 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

图 8 故障单元矩阵

$$P = \begin{bmatrix} A1 & A2 & A3 & A4 & A5 & A6 & A7 & A8 & A9 & A10 & A11 & A12 & A13 & A14 & A15 & A16 \\ 0 & 0 & 0 & 0 & 0 & 22.2\% & 0 & 16.7\% & 16.7\% & 16.7\% & 0 & 11.1\% & 11.1\% & 5.6\% & 0 & 0 \end{bmatrix}$$

图 9 故障概率矩阵

根据测试结果的故障概率矩阵可知，A6 发生故障的概率最大，为 22.2%，因此应优先启动对 A6 的维修更换。

当 A6 维修更换完毕后，对设备重新启动测试，此时 A6 的故障已消除，但 A10 故障仍存在，则测试结果将为  $T = \{1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0\}$ ，按照上述方法进行分析，依次获得测试结果相关性矩阵、故障单元矩阵和故障概率矩阵，分别如图 10~12 所示。

RD	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16
TR1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
TR2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
TR3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
TR4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
TR5	1	0	0	0	1	1	0	1	0	0	0	0	0	0	1	1
TR6	0	1	0	0	1	1	0	0	1	0	0	0	0	0	1	1
TR7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR8	0	0	0	1	1	0	1	0	0	0	1	0	0	0	1	1
TR9	1	1	0	0	1	1	0	1	1	0	0	1	0	0	1	1
TR10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TR11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

图 10 组成单元与测试结果相关性 RD 矩阵

$$A_{RD} = \begin{bmatrix} A1 & A2 & A3 & A4 & A5 & A6 & A7 & A8 & A9 & A10 & A11 & A12 & A13 & A14 & A15 & A16 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

图 11 故障单元矩阵

$$P = \begin{bmatrix} A1 & A2 & A3 & A4 & A5 & A6 & A7 & A8 & A9 & A10 & A11 & A12 & A13 & A14 & A15 & A16 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 50\% & 0 & 0 & 33.3\% & 16.7\% & 0 & 0 \end{bmatrix}$$

图 12 故障概率矩阵

由故障概率矩阵可知组成单元 A10 发生故障的概率最大, 为 50%, 因此应优先启动对 A10 的维修。

至此, 通过两次检测和维修活动, 将所有故障准确检测隔离并维修完成。

### 2.6 验证结果对比分析

在本实例中, 当电子设备同时发生 A6 和 A10 故障时, 采用传统的诊断设计方法和采用多故障诊断设计方法进行的设计, 其诊断和维修的对比如表 3 所示。

表 3 A6 和 A10 故障时刻设备测试情况

测试维修	传统的诊断设计	多故障诊断设计
第一次诊断	报 A5 或 A16 故障	报 A6(22.2%)故障
第一次维修	更换 A5	更换 A6
第二次诊断	报 A5 或 A16 故障	报 A10(50%)故障
第二次维修	更换 A16	更换 A10
第三次诊断	报 A5 或 A16 故障	无故障

由对比结果可知, 采用传统的诊断设计方法获得的诊断树对故障进行了错误的定位, 导致开展了多次无效的维修活动。而采用多故障诊断设计方法进行的设计, 通过故障发生概率大小的方式将故障单元准确区分出来, 通过较少的维修活动即完成了设备的修复。

### 2.7 算法性能综合分析

为进一步分析多故障诊断设计方法在不同多故障情况下的诊断准确性, 针对验证系统开展了组成单元所有组合故障情况下的算法性能对比分析, 具体情况如表 4 所示。

表 4 所有故障情况下的检测分析

故障单元数量	故障样本数	传统的诊断设计方法		多故障诊断设计方法	
		隔离正确样本数量	隔离准确率/%	隔离正确样本数量	隔离准确率/%
1	16	16	100	16	100
2	120	109	90.83	118	98.33
3	560	464	82.86	546	97.50
4	1 820	1 391	76.43	1 754	96.37
5	4 368	3 148	72.07	4 159	95.22
6	8 008	5 600	69.93	7 588	94.76
7	11 441	7 962	69.59	10 884	95.13
8	12 871	9 068	70.45	12 343	95.90
9	11 439	8 245	72.08	11 045	96.56
10	8 007	5 943	74.22	7 761	96.93
11	4 368	3 355	76.81	4 244	97.16
12	1 820	1 455	79.95	1 775	97.53
13	560	469	83.75	550	98.21
14	120	106	88.33	119	99.17
15	16	15	93.75	16	100
16	1	1	100	1	100
合计	65 535	47 347	72.25	62 919	96.01

由对比分析表可知, 多故障诊断设计方法的算法更适应系统的各种故障情况, 特别是在多故障情况下, 对故障单元的定位可达 96% 以上, 准确度远远高于传统诊断设计方法的算法。

### 3 应用验证

针对提出的多故障诊断设计方法, 在某收发阵列处理设备中开展了应用验证。

验证架构如图 13 所示, 某收发阵列处理设备采集内部各测试点的状态, 通过设备内部的控制模块上报上位机, 上位机根据实时的测试结果, 利用多故障诊断算法进行故障检测和隔离。

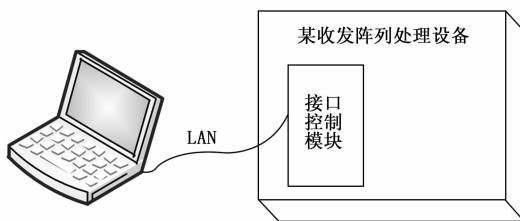


图 13 应用验证架构

上位机上的多故障诊断算法采用 C 语言实现, 如下所示, 诊断算法中需要的测试性相关性矩阵通过测试性建模获得。

```

Void FaultAnalyse(
int CellNum,
int TestNum,
int TestResult[],
int TestDMatrix[],
int TNumMatrix[],
double * AnalyseResult[]
)
{
int i=0;
int CellResult=0;
int TempData=0;
int TestNumResult[DefineNum]={0};
int SumData=0;

CellResult=0;
for(i=0;i<TestNum;i++)
{
if(TestResult[i]==1)
CellResult=CellResult|TestDMatrix[i];
}

for(i=0;i<CellNum;i++)
{
TempData=CellResult&0x0001;
if(TempData==0x00)
    
```

```

TestNumResult[CellNum-i-1] = TNumMatrix[CellNum-i
-1];
else
TestNumResult[CellNum-i-1]=0;
CellResult=CellResult>>1;
}
SumData=0;
for(i=0;i<CellNum;i++)
SumData=SumData+TestNumResult[i];
for(i=0;i<CellNum;i++)
* AnalyseResult[i]=(double)TestNumResult[i]/SumData;
}
    
```

收发阵列处理设备内部由 14 个模块组成，内部共计 11 个测试项目，分析形成  $11 \times 14$  的  $D$  矩阵。在多故障注入验证中，设置了多个模块同时故障情况<sup>[19-20]</sup>，图 14 为人为设置收发模块 2、收发模块 6、收发模块 8 和放大模块故障时的软件故障分析定位情况。通过多故障算法软件分析定位出发模块 2、收发模块 6、收发模块 8 发生故障概率最高 (19%)，与实际故障设置情况符合。



图 14 多故障诊断算法分析报故情况

经分析，收发阵列处理设备 14 个模块，在不同组合故障情况下，11 个测试项目组成的测试结果共计 258 种，样本 16 383 个，可正确检测隔离的为 16 348 个，覆盖达到 99%。

#### 4 结束语

本文针对电子设备在设计过程中采用单故障假设进行的诊断设计无法适应使用过程中发生多故障时的故障隔离定位需求，提出了基于相关性  $D$  矩阵的多故障诊断设计方法。通过实例验证，本文提出的设计方法在测试结论中引入了故障概率信息，这极大地提高了对故障单元定位的准确度，方法简单易懂，具有易操作性，特别适用于复杂的电子设备。

#### 参考文献:

- [1] 刘刚, 黎放, 胡斌. 基于相关性模型的舰船装备测试性分析与建模 [J]. 海军工程大学学报, 2012, 24 (4): 46-51.
- [2] 刘刚, 吕建伟, 胡斌. 复杂装备测试性建模问题研究 [J]. 舰船电子工程, 2013, 33 (5): 137-139.
- [3] 田恒, 段富海, 樊亮, 等. 基于测试性 D 矩阵的多故障诊断与维修策略 [J]. 北京航空航天大学学报, 2018, 44 (4): 802-809.
- [4] 王成刚, 王学伟, 周晓东. 测试性建模与分析中的故障概率获取方法研究 [J]. 测试技术学报, 2010, 24 (1): 9-14.
- [5] 徐俊洁, 陈荣. 基于故障关联的多故障概率诊断方法 [J]. 计算机科学, 2017, 44 (4): 124-130.
- [6] 王红霞, 潘红兵, 叶晓慧. 多故障的测试序列问题研究 [J]. 兵工学报, 2011, 32 (12): 1518-1523.
- [7] 王子玲, 许爱强, 牛双诚. 基于多故障假设的诊断策略研究与应用 [J]. 工程设计学报, 2009, 16 (4): 281-285.
- [8] 刘晓芹, 黄考利, 连光耀, 等. 针对动态系统的多故障诊断模型与优化算法 [J]. 计算机测量与控制, 2010, 18 (2): 249-266.
- [9] 张毅, 杨秀霞. 飞行器电子系统的多故障诊断算法研究 [J]. 系统工程与电子技术, 2006, 28 (5): 788-790.
- [10] 欧阳一鸣, 鲁传武, 梁华国. 基于多故障模型的并发测试生成方法 [J]. 计算机工程与应用, 2007, 43 (29): 113-115.
- [11] 石君友. 测试性设计分析与验证 [M]. 北京: 国防工业出版社, 2015.
- [12] 杨智勇, 许爱强, 牛双诚. 基于多信号模型的系统测试性建模与分析 [J]. 工程设计学报, 2007, 14 (5): 364-394.
- [13] 马瑞萍, 董海迪, 马长李. 基于故障-测试相关性矩阵的测试性分析 [J]. 兵工自动化, 2016, 35 (5): 5-7.
- [14] 石君友, 龚晶晶, 徐庆波. 考虑多故障的测试性建模改进方法 [J]. 北京航空航天大学学报, 2010, 36 (3): 270-298.
- [15] 秦玉峰, 史贤俊, 王康. 考虑不确定因素的系统测试性建模与分析方法研究 [J]. 舰船电子工程, 2019, 39 (2): 150-170.
- [16] 闫鹏程, 连光耀, 刘晓芹, 等. 基于多故障模糊组的序贯多故障诊断方法 [J]. 计算机测量与控制, 2012, 20 (1): 34-37.
- [17] 蒋俊荣, 黄考利, 吕晓明, 等. 基于混合诊断模型的测试性建模技术研究 [J]. 计算机测量与控制, 2010, 18 (12): 2690-2698.
- [18] 张士刚, 刘颖, 宋立军, 等. 基于模型的系统级故障诊断与预测推理技术 [J]. 国防科技, 2016, 37 (2): 9-13.
- [19] 尹园威, 尚朝轩, 马彦恒, 等. 基于故障注入的雷达装备测试性验证试验方法 [J]. 计算机测量与控制, 2014, 22 (7): 2128-2134.
- [20] 颜世刚, 齐亚峰. 测试性验证试验的故障注入方法优化研究 [J]. 计算机测量与控制, 2019, 27 (4): 97-101.