ANALYSIS OF FALSE ALARMS DURING SYSTEM DESIGN

William R. Simpson
John W. Sheppard

ARINC Research Corporation
2551 Riva Road
Annapolis, MD 21401

ABSTRACT

The problem of false alarms in electronic monitoring systems has grown over the last decade. This growth has been associated with increasing system complexity and advances in the state of the art. Studies have shown that some systems exhibit as many as 40% or more "false pulls," and associated with the false-alarm problem, a large volume of wasted or ineffective maintenance actions exist. What is a false alarm? Can the extent of the problem be anticipated by the designer? Can a designer take steps to eliminate or reduce the effects of false alarms? This paper explores the answers to these questions by describing field maintenance data, prediction models, and analytical techniques.

INTRODUCTION

What is a false alarm? The experts in field maintenance are far from reaching consensus as indicated by the literature. MIL-STD-2165 defines a false alarm as a fault indicated by built-in test (BIT) or other monitoring circuitry where no fault exists.[1] MIL-STD-1309C defines false alarms the same way—by limiting the definition to BIT.[2] The *RADC Testability Notebook* defines false alarm as an indicated fault where no fault exists.[3] The source of indication may be by BIT or other means. A survey conducted for Rome Air Development Center (RADC) to obtain intuitive definitions includes the two above plus several others, including a failure detection that cannot be repeated.[4]

This large variance in definition ignores several issues important to field maintenance. For example, if an indication is known to be a false alarm and it does not trigger a maintenance action, then is it a false alarm? For an easily recognized false alarm, we either filter out or ignore the indication so that there is no fault indicated. If we are unable to define false alarms from this standpoint, then we should look at their effects. In the field, we have two principal effects of false alarms:

- Increased maintenance because of diagnosis being performed on otherwise healthy systems

- Decreased mission effectiveness because we ignore indications that we think are false alarms (Some of these may be real failures.)

We would put the latter in the undetected failure category. The indications may have been detected and ignored, but the tendency toward false alarms has placed them in the latter category. The former is certainly in the false-alarm category, but suffers from not being visible because we may diagnose an unhealthy system, but not be able to reproduce the fault. This is typically classified in maintenance reporting schemes as one of the following:

- No Evidence of Fault (NOEF)

- No Fault Found (NFF)

- Cannot Duplicate (CND).

What then, is a false alarm? We tend to favor a definition that is based on a maintenance event. At least from a field perspective we know something happens. For example, a spurious signal filtered by the BIT software would not be taken as an indication. Conversely, an anomaly not detected by BIT but strong enough to generate a system operator complaint may result in a maintenance action. By this definition, a false alarm would be defined as:

> A fault indication that triggers a maintenance action where no fault exists.

CH3158-3/92/0000-0657 $1.00 © 1992 IEEE

If the maintenance action is diagnose and repair, and there is no fault, then we will obtain a NOEF, NFF or CND.

THE FALSE-ALARM SPECIFICATION

Even this definition of false alarm is not measurable in the field unless we conduct a detailed laboratory follow-up of our CND event to ascertain the cause of the CND. A study of more than 22,000 maintenance events on 38 systems classified 12 subcategories of CNDs.[5] Five of these were designated as false alarms. Some of the possible causes were human errors, test equipment failures, and BIT failures. In practice false alarms are simply not measurable unless we are willing to submit every maintenance event to a thorough postmortem examination. In fact, any specification for false alarms will either be ignored or not enforceable because no contractor will accept any definition without requiring the postmortem examination described above. It should be pointed out that on new systems, the only party qualified to conduct such a postmortem examinatiön is the manufacturer of the system.

In the past, specification of false alarms served as an indication of wishes rather than an enforceable event. A specification that can be enforced would be based on measurable events, and preferably ones that are reported, although special reporting schemes could be considered. Two such measures that deal with the problems of ineffective maintenance would be CND rate and false-removal rate. These are favorite topics of logistics engineers and maintenance analysts because real field data exist. The CND rate makes a fair estimate of false-alarm rate if we use the definition recommended in this paper. Of course this would not include the failure indications that are ignored, but it would include every other false alarm (together with a few other causes such as maintenance errors or BIT errors). This would make CND rate an upper bound to false alarm rate.

PREDICTING MAINTENANCE PROBLEMS

System-level diagnosis has always been an afterthought in system design. Initially (i.e, circa 1930) system-level failures announced themselves. Parts fell off, items quit working, or the failure symptom itself pointed to the subsystem that demanded repair. As systems grew in complexity, maintenance became less predictable. Maintenance technicians were not always sure when the system was not working right and some failures were more difficult to detect. Nevertheless, most failures were still fairly easy to locate. More recently system reliability has improved significantly. Parts of many systems have significantly lower failure rates, but for the more complex systems, overall failure rates continue to be significant. In addition, system and test design has resulted in "false pull" rates of 40% or more. Studies of the CH-54 and F-16 show that troubleshooting actions can consume 50% of the total labor-hours spent for repairs.[6-7] Data for the scheduled airlines show similar trends for complex electronics.[8]

Several options exist for reducing the problem of false alarms in the field, as discussed below. Unfortunately, these options are all expensive. The expense, however, is considerably less than fielding a system with the problems cited above. Further, we can predict which systems are likely to have a problem, thus providing a means of selecting those systems that will benefit most from efforts to reduce false alarms. For an earlier referenced study the authors were able to develop predictor equations for CND rate using design attributes with a 91% to 92% correlation factor.[5] Critical variables in the prediction of CND rate and the burden associated with CND are the following:

- Complexity measures related to the topological patterns of functional paths and similar to measures derived in sneak circuit analysis
- Failure rate prediction from MIL-HDBK-217 or other methodology[9]
- Measures of transient factors related to relays, capacitors, integrated circuits, and transistors

The original work included two systems for verification, and the actual values were within 10% of the predicted values and well within the 95% prediction interval.[5] Only one full-scale prediction has been undertaken thus far and it involved 12 line replaceable units on a state-of-the-art radar system.[9-10] To date, not enough field data have been gathered to check the accuracy of the prediction, although the qualitative ranking of systems with expected problems appears correct.

PREVENTING MAINTENANCE PROBLEMS

Predicting a problem is, of course, insufficient. Once we know that a problem will exist in a fielded system, we should take action to minimize the problem. False alarms result from imperfect testing. The better we understand a process or technology, the more accurate the testing becomes. False alarms usually become a problem when system complexity becomes great, or a design pushes the state of the art, or both. Four viable solutions to false-alarm problems are available:

- Improving test science—We can avoid false alarms by sampling more often, modeling in greater detail, and accounting for a greater number of variables. In the case of BIT, this creates an increased software requirement and may require the addition of sensors to the built-in test equipment (BITE).

- Increasing tolerances for the test—We can avoid false alarms by making the test less sensitive to anomalous behavior. Unfortunately, this may also result in reducing the ability of the test to detect real anomalies.

- Conducting repeat polling—In repeat polling, we try to avoid false alarms by executing a test multiple times. Each time the test is evaluated, the test algorithm uses the results to confirm any previous executions. Repeat polling is intended to allow transient characteristics to work their way through the system without triggering a failure indication. Repeat polling requirements are usually written as, "*n* or more indications within *m* time units." This may also lead to missed detections. A better approach would be to recognize transient characteristics using the first solution above.

- Cross-correlating test information—We can correlate an anomalous indication with other testing to either confirm or deny the original information. The information flow model can utilize this technique to assist in planning for false-alarm prosecution.

The first three of these are empirical. The last can be done analytically.

FALSE-ALARM TOLERANCE

False-alarm tolerance (FAT) is a measure of our ability to perform test-to-test cross-checking. If we examine Figure 1, we can see that $test_2$ can be used to verify an anomalous indication of $test_1$.

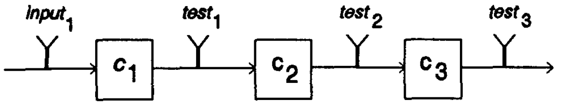

Figure 1. Serial System with Functional Tests

We can also use $test_3$ in this fashion. This enables us to measure our ability to cross-correlate, thus the false-alarm tolerance may be computed as:

$$\text{FAT} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} \phi_{ij}}{(N)\,(N-1)} \;;$$

$$\phi_{ij} = \begin{cases} 1; \text{ if } \boldsymbol{test_j} \text{ can confirm or deny } \boldsymbol{test_i} \text{ and } i \neq j \\ 0; \text{ otherwise} \end{cases}$$

where N = the number of tests

ϕ_{ij} = a confirmation factor

An ideal value of FAT would be 0.5000, which is based on a fully tested serial system. In truly complex systems, the FAT may be hard to compute by hand and automated analysis tools such as STAMP®[10] may be needed to compute the value. FAT will typically decrease as systems become larger and tests are removed. We have found that real systems with FAT values below 10% should be carefully analyzed. One way to maintain a high false-alarm tolerance is to retain redundant and excess tests. Of course, merely having these tests available is not sufficient; they must be used by the diagnostic strategy, which means that additional testing will be specified. The referenced tools provide a means for incorporating these extra tests in the diagnostic strategies.

SUMMARY

The subject of false alarms is complex. It is further complicated by the inability of false alarms to be measured by any reasonable means from field data. It should be avoided altogether in the specification process. Specifications on CND and false pull rates do not simply provide factors keyed to the maintenance process; these factors can actually be measured and used to enforce specifications.

The CND rate can be used as at least an upper bound on the false-alarm rate if definitions are chosen properly. This factor can be predicted during the design of a system with some rather simple techniques discussed and referenced in this paper. When the CND rate is excessive, a number of techniques exist, both empirical and analytic, to improve the test design and reduce field rates.

REFERENCES

1. Naval Electronic Systems Command (ELEX-8111). *Testability Program for Electronic Systems and Equipment*. MIL-STD-2165, Washington, DC: Naval Electronic Systems Command, 26 January 1985.

2. Naval Electronics Systems Command (ELEX-8111). *Definitions of Terms for Test, Measurement and Diagnostic Equipment*. MIL-STD-1309C. Washington, DC: Naval Electronics Systems Command, 18 November 1983.

3. Hughes Aircraft Company. *RADC Testability Notebook.* RADC-TR-82-189. Griffis AFB, NY: Rome Air Development Center, June 1982.

4. W. R. Simpson, J. H. Bailey, K. B. Barte, and E. Esker. *Prediction and Analysis of Testability Attributes: Organizational-Level Testability Prediction.* RADC-TR-85-268. Griffis AFB, NY: Rome Air Development Center, February 1986.

5. A. E. Gilreath, B. A. Kelley, and W. R. Simpson. *Predictors of Organizational-Level Testability Attributes.* 1511-02-2-4179. Annapolis MD: ARINC Research Corporation, 1 November 1986.

6. Thomas N. Cook, and J. Ariano. "Analysis of Fault Isolation Criteria/Techniques." *Proceedings Annual Reliability and Maintainability Symposium.* (held in San Francisco, CA) January 1980.

7. M. L. Labit, G. T. Harrison, and B. L. Rutterer. *Special Report on Operational Suitability (OS) Verification Study Focus on Maintainability.* 1751-01-2-2395. Annapolis, MD: ARINC Research Corporation, February 1981.

8. Aeronautical Radio, Inc. *Avionics Maintenance Conference Report—San Diego, 1987.* 87-087/MOF-34. Annapolis, MD, August 1987.

9. Rome Air Development Center. MIL-HDBK-217D. "Reliability Prediction of Electronics Equipment," Griffis AFB, Rome, NY: Rome Air Development Center, 1983.

10. W. R. Simpson, J. W. Sheppard, B. A. Kelley, and J. L. Graham. *Testability Prediction Report: Joint Stars Radar Subsystem.* 1518-21-01-4689. Annapolis, MD: ARINC Research Corporation 31 May 1988.

11. W. Simpson and J. Sheppard. "System Complexity and Integrated Diagnostics," *IEEE Design and Test of Computers*, Volume 8, No. 3, September 1991, pp. 16-30.