FAULT TOLERANT AND FAULT TESTABLE HARDWARE DESIGN

Parag K. Lala

FAULT TOLERANT AND FAULT TESTABLE HARDWARE DESIGN

Parag K. Lala

formerly of the
Department of Computer Science
University of York, UK

currently
Associate Professor
Dept. of Electrical and Computer Engineering
Syracuse University
New York, USA

Prentice/Hall PHI International

Englewood Cliffs, New Jersey London New Delhi Rio de Janeiro
Singapore Sydney Tokyo Toronto Wellington

Library of Congress Cataloging in Publication Data

Lala, Parag K., 1948–
Fault tolerant and fault testable hardware design.

Bibliography: p.
Includes index.
1. Electronic digital computers—Reliability. 2. Fault-tolerant computing. I. Title
TK7888.3.L27 1985 621.3819'583 84-8316
ISBN 0-13-308248-2 (case)

British Library Cataloguing in Publication Data

Lala, Parag, K.
Fault tolerant and fault testable hardware design.
1. Fault-tolerant computing
I. Title
001.64 QA76.9.F38

ISBN 0-13-308248-2

0-13-308248 2

Prentice-Hall International, Inc., *London*
Prentice-Hall of Australia Pty Ltd., *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall of India Private Ltd., *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Prentice-Hall of Southeast Asia Pte Ltd., *Singapore*
Prentice-Hall, Inc., *Englewood Cliffs, New Jersey*
Prentice-Hall do Brasil Ltda, *Rio de Janeiro*
Whitehall Books Ltd., *Wellington, New Zealand*

10 9 8 7 6 5 4 3 2 1

Typeset by Pintail Studios Ltd., Ringwood, Hampshire, England

Printed in the United States of America

PREFACE

Fault tolerance and testability have the common objective of improving the reliability of computer hardware. Fault tolerance is concerned with masking or recovering from the effects of faults once they have been detected, whereas testability involves a design approach aimed at easing the detection of faults. It is the relative cost which determines whether for a certain application one approach is more desirable than the other. Much work has been done recently in both areas. Such efforts have resulted in certain design philosophies and techniques. Unfortunately many engineers are not familiar with these; consequently there is an urgent need for university courses and textbooks to provide engineers with both general and specific directions on designing for fault tolerance and testability.

This book has been written mainly as a reference volume for postgraduate students in electrical engineering and computer science; it will also be suitable as a text for final-year options in undergraduate courses, provided that the readers have some background in switching theory and logic design. The book will prove equally useful for practicing engineers who require familiarity with recent research on reliable hardware design; a list of references on which I have drawn is provided at the end of each chapter.

Chapter 1 deals with the basics of reliability theory. Common terms used in reliability measure, such as mean-time-between-failures and availability, are defined and the importance of maintainability is emphasized.

Chapter 2 covers most of the important faults that can be found in digital circuits. The classical stuck-at-1, stuck-at-0 logic faults are discussed and then non-classical faults, such as bridging and stuck-open faults, are introduced. The difference between intermittent and transient faults is also considered and various models for intermittent faults are described.

Chapter 3 covers fault detection in combinational and sequential logic circuits. The basic testing techniques for detecting fault conditions in combinational circuits are explained with examples; the problem created by the presence of multiple faults is also considered. The testing of sequential circuits still remains a major problem, for which no generally accepted solution has been found. In this chapter the state-table verification technique for testing sequential circuits is discussed in detail. Next some non-conventional yet extremely effective techniques for testing logic circuits, such as random and transition count testing, are introduced. Finally the signature testing of logic circuits using linear shift registers is presented in detail with examples of signature formats and their generation.

Chapter 4 discusses in detail many classes of hardware fault tolerance techniques including a detailed examination of fault detection and recovery methods. The use of error-correcting codes in the design of fault tolerant sequential circuits and computer memory systems have also been dealt with. In addition software and time redundancy techniques are outlined. The concept of "fail soft" operation is introduced and several fault tolerant systems are described. Finally a scheme for fault tolerant design of VLSI chips is given.

Chapter 5 presents recent developments in the area of self-checking and fail-safe circuit design. The design of a specific circuit type, self-checking checkers, is examined for various error-detecting codes which are likely to be used in hardware design. Self-checking and fail-safe design of sequential circuits are also considered. The last section deals with the work done so far on the design of self-checking programmable logic arrays (PLA).

Chapter 6 focuses on the various design techniques which can be used to enhance the testability of combinational and sequential logic circuits. A number of design methods, proposed for improving the testability of VLSI chips, are described. The concepts of built-in-tests and autonomous self-tests are explained. In addition several techniques are considered to improve the testability of logic boards designed without due consideration to their testing.

Chapter 7 discusses the current research issues in reliable computing systems design.

The appendix is included to provide an introduction to Markov models, which have been widely applied to the study of temporary faults. An annotated bibliography of conference proceedings and books for further reading, is also given.

I am grateful to my colleagues Professor I. C. Pyle and Mr D. G. Burnett-Hall, who helped me, directly or indirectly, in the preparation of certain sections of the manuscript. I also thank the reviewers of the manuscript for their constructive criticism, which has helped in making this a better book than it would have been otherwise. My special thanks go to Dr J. I. Missen of the City University, London, who first introduced me to the subject. I am also indebted to Angela Fairclough, who typed parts of the manuscript.

Lastly I acknowledge the support of my wife, Meena, with typing, editing and retyping to bring this text into reality; without her help and encouragement I should still be working on the manuscript.

PARAG K. LALA

CONTENTS

4 FAULT TOLERANT DESIGN OF DIGITAL SYSTEMS 69

1 BASIC CONCEPTS OF RELIABILITY

1.1 THE DEFINITION OF RELIABILITY

In recent years the complexity of digital systems has increased dramatically. Although semiconductor manufacturers try to ensure that their products are reliable, it is almost impossible not to have faults somewhere in a system at any given time. As a result, reliability has become a topic of major concern to both system designers and users [1.1, 1.2]. A fundamental problem in estimating reliability is whether a system will function in a prescribed manner in a given environment for a given period of time. This, of course, depends on many factors such as the design of the system, the parts and components used, and the environment. Performance of a given system, under given conditions, for a given period of time can be considered as a chance event—i.e. the outcome of the event is unknown until it has actually occurred. Hence it is natural to consider the reliability of a system as an unknown parameter which is defined to be the probability that the given system will perform its required function under specified conditions for a specified period of time.

The reliability of a system can be increased by employing the method of worst case design, using high-quality components and imposing strict quality control procedures during the assembly phase. However such measures can increase the cost of a system significantly. An alternative approach to reliable system design is to incorporate "redundancy" (i.e. additional resources) into a system with the aim of masking the effects of faults. This approach does not necessitate the use of high-quality components; instead standard components can be used in a redundant and reconfigurable architecture (see Chap. 4). In view of the decreasing cost of hardware components it is certainly less expensive to use the second approach to design reliable systems.

1.2 RELIABILITY AND THE FAILURE RATE

Let us consider the degradation of a sample of N identical components under "stress conditions" (temperature, humidity, etc.). Let $S(t)$ be the number of surviving components, i.e. the number of components still operating at time t after the beginning of the "ageing experiment", and $F(t)$ the number of components that have failed up to time t. Then the probability of survival of the components, also known as the *reliability* $R(t)$, is

$$R(t) = \frac{S(t)}{N}$$

The probability of failure of the components, also known as the *unreliability* $Q(t)$, is

$$Q(t) = \frac{F(t)}{N}$$

Since $S(t) + F(t) = N$, we must have

$$R(t) + Q(t) = 1$$

The failure rate, also known as the "hazard rate", $Z(t)$ is defined to be the number of failures per unit time compared with the number of surviving components:

$$Z(t) = \frac{1}{S(t)} \frac{\mathrm{d}F(t)}{\mathrm{d}t} \tag{1.1}$$

Study of electronic components show that under normal conditions the failure rate varies as indicated in Fig. 1.1. There is an initial period of high failure because in any large collection of components there are usually components with defects and these fail, i.e. they do not work as intended, after they are put into operation. For this reason, the first period is called the "burn-in" period of defective components. The middle phase is the "useful life" period when the failure rate is relatively constant; in this phase failures are random in time. The

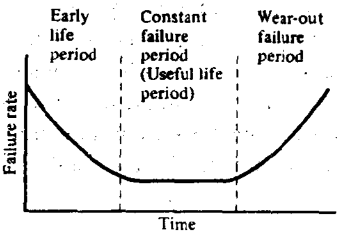

Fig. 1.1 Variation of failure rate with time.

final phase is the "wear out" period, when the failure rate begins to increase rapidly with time. The curve of Fig. 1.1 is often called the "bath-tub" curve because of its shape.

In the "useful life" period the failure rate is constant, and therefore

$$Z(t) = \lambda \text{ (say)} \tag{1.2}$$

With the previous nomenclature,

$$R(t) = \frac{S(t)}{N} = \frac{N - F(t)}{N} = 1 - \frac{F(t)}{N}$$

Therefore

$$\frac{\mathrm{d}R(t)}{\mathrm{d}t} = -\frac{1}{N} \cdot \frac{\mathrm{d}F(t)}{\mathrm{d}t}$$

or

$$\frac{\mathrm{d}F(t)}{\mathrm{d}t} = -N\frac{\mathrm{d}R(t)}{\mathrm{d}t} \tag{1.3}$$

Substituting equations (1.2) and (1.3) in equation (1.1)

$$\lambda = -\frac{N}{S(t)} \cdot \frac{\mathrm{d}R(t)}{\mathrm{d}t}$$

$$= -\frac{1}{R(t)} \cdot \frac{\mathrm{d}R(t)}{\mathrm{d}t} \quad \text{since} \quad R(t) = \frac{S(t)}{N}$$

or

$$\lambda \cdot \mathrm{d}t = -\frac{\mathrm{d}R(t)}{R(t)}$$

The above expression may be integrated giving

$$\lambda \int_0^t \mathrm{d}t = -\int_1^{R(t)} \frac{\mathrm{d}R(t)}{\mathrm{d}t}$$

The limits of the integration are chosen in the following manner: $R(t)$ is 1 at $t = 0$, and at time t by definition the reliability is $R(t)$. Integrating, then

$$\lambda |t|_0^t = |\log_e R(t)|_1^{R(t)}$$

$$\lambda t = -|\log_e R(t) - \log_e 1|$$

$$-\lambda t = \log_e R(t)$$

Therefore

$$R(t) = \exp(-\lambda t) \tag{1.4}$$

The above relationship is generally known as the *exponential failure law*; λ is usually expressed as percentage failures per 1000 hours or as failures per hour. When the product λt is small,

$$R(t) = 1 - \lambda t \tag{1.5}$$

System failures, like component failures, can also be categorized into three periods of operation. The early system failures such as wiring errors, dry joints, faulty interconnections, etc., are normally eliminated by the manufacturer's test procedures. System failures occurring during the useful life period are entirely due to component failures.

If a system contains k types of component, each with failure rate λ_k, then the system failure rate, λ_{ov}, is

$$\lambda_{ov} = \sum_{1}^{k} N_k \lambda_k$$

where there are N_k of each type of component.

1.3 RELATION BETWEEN RELIABILITY AND MEAN-TIME-BETWEEN-FAILURES

Reliability $R(t)$ gives different values for different operating times. Since the probability that a system will perform successfully depends upon the conditions under which it is operating and the time of operation, the reliability figure is not the ideal for practical use. More useful to the user is the average time a system will run between failures; this time is known as the *mean-time-between-failures* (MTBF). The MTBF of a system is usually expressed in hours and is given by $\int_0^\infty R(t)\,dt$, i.e. it is the area underneath the reliability curve $R(t)$ plotted versus t; this result is true for any failure distribution. For the exponential failure law,

$$\text{MTBF} = \int_0^\infty \exp(-\lambda t)\,dt$$

$$= -\frac{1}{\lambda}\left|\exp(-\lambda t)\right|_0^t = \frac{1}{\lambda} \tag{1.6}$$

In other words, the MTBF of a system is the reciprocal of the failure rate. If λ is the number of failures per hour, the MTBF is expressed in hours. If, for example, we have 4000 components with a failure rate of 0.02% per 1000 hours, the average number of failures per hour is:

$$\frac{0.02}{100} \times \frac{1}{1000} \times 4000 = 8 \times 10^{-4} \text{ failures/hour}$$

The MTBF of the system is therefore equal to $1/(8 \times 10^{-4})$ or $1/8 \times 10^4 =$ 1250 hours. Substituting equation (1.6) in the reliability expression equation (1.4) gives

$$\begin{aligned} R(t) &= \exp(-\lambda t) \\ &= \exp(-t/\text{MTBF}) \end{aligned} \tag{1.7}$$

A graph of reliability against time is shown in Fig. 1.2. As time increases the reliability decreases and when $t = \text{MTBF}$, the reliability is only 36.8%. Thus a system with an MTBF of say 100 hours has only a 36.8% chance of running 100 hours without failure.

By combining equations (1.5) and (1.6), we have

$$\begin{aligned} R(t) &= 1 - \lambda t \\ &= 1 - \frac{t}{\text{MTBF}} \end{aligned}$$

Therefore

$$\text{MTBF} = \frac{t}{1 - R(t)}$$

Example A first-generation computer contains 10 000 thermionic valves each with $\lambda = 0.5\%/(1000 \text{ hours})$. What is the period of 99% reliability?

$$\begin{aligned} \text{MTBF} &= \frac{t}{1 - 0.99} \\ t &= \text{MTBF} \times 0.01 \\ &= \frac{0.01}{\lambda_{ov}} \end{aligned} \tag{1.8}$$

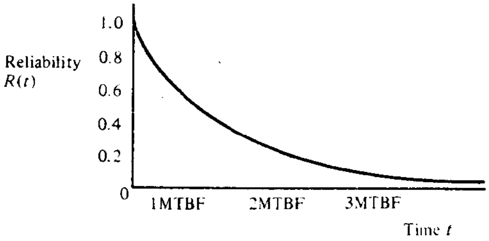

Fig. 1.2 Reliability curve.

$$N = \text{No. of valves} = 10\,000$$

$$\lambda = \text{failure rate of valves} = 0.5\%/(1000 \text{ hours})$$

$$= 0.005/(1000 \text{ hours})$$

$$= 5 \times 10^{-6}/\text{hour}$$

Therefore

$$\lambda_{ov} = N\lambda = 10^4 \times 5 \times 10^{-6} = 5 \times 10^{-2}/\text{hour}$$

From equation (1.8),

$$t = \frac{0.01}{5 \times 10^{-2}} = \frac{10^{-2}}{5 \times 10^{-2}} = 0.2 \text{ hour} = 12 \text{ minutes}$$

This figure was often typical!

1.4 MAINTAINABILITY

When a system fails, repair action is normally carried out to restore the system to operational effectiveness. The probability that a failed system will be restored to working order within a specified time is called the *maintainability* of the system. In other words maintainability is the probability of isolating and repairing a "fault" (see Chap. 2) in a system within a given time. There is therefore a relationship between maintainability and repair rate μ and hence with mean-time-to-repair (MTTR). MTTR and μ are always related |1.3|:

$$\mu = \frac{1}{\text{MTTR}}$$

MTTR and μ are related to maintainability $M(t)$ as follows:

$$M(t) = 1 - \exp(-\mu t) = 1 - \exp\left(-\frac{t}{\text{MTTR}}\right)$$

where t is the permissible time constraint for the maintenance action.

In order to design and manufacture a maintainable system, it is necessary to predict the MTTR for various fault conditions that could occur in the system. Such predictions are generally based on the past experiences of designers and the expertise available to handle repair work.

The system repair time consists of two separate intervals—passive repair time and active repair time |1.3|. The passive repair time is mainly determined by the time taken by service engineers to travel to the customer site. In many cases the cost of travel time exceeds the cost of the actual repair. The active repair time is directly affected by the system design and may be subdivided as follows:

1. The time between the occurrence of a failure and the system user becoming aware that it has occurred.
2. The time needed to detect a fault and isolate the replaceable component(s) responsible.
3. The time needed to replace the faulty component(s).
4. The time needed to verify that the fault has been removed and the system is fully operational.

The active repair time can be improved significantly by designing the system so that faults may be detected and quickly isolated. As more complex systems are designed, it becomes more difficult to isolate the faults. However if adequate self-test features are incorporated into the replaceable components of a system, it becomes easier to detect and isolate faults, which facilitates repair |1.4|.

1.5 AVAILABILITY

The availability of a system is the probability that the system will be "up", i.e. functioning according to expectations at any time during its scheduled working period |1.3|.

$$\text{Availability} = \frac{\text{System up-time}}{\text{System up-time} + \text{System down-time}}$$

$$= \frac{\text{System up-time}}{\text{System up-time} + (\text{No. of failures} \times \text{MTTR})}$$

$$= \frac{\text{System up-time}}{\text{System up-time} + (\text{System up-time} \times \lambda \times \text{MTTR})}$$

$$= \frac{1}{1 + (\lambda \times \text{MTTR})}$$

$$= \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}} \quad \text{since} \quad \lambda = \frac{1}{\text{MTBF}}$$

If the MTTR can be reduced, availability will increase and the system will be more economical. A system where faults are rapidly diagnosed is more desirable than a system which has a lower failure rate but where the cause of a failure takes a long time to locate, and consequently a lengthy system downtime is needed for repair.

1.6 SERIES AND PARALLEL SYSTEMS

The reliability of a system can be derived in terms of the reliabilities or the failure rates of the subsystems used to build it. Two limiting cases of system design are frequently met in practice:

1. Systems in which each subsystem must function if the system as a whole is to function.
2. Systems in which the correct operation of just one subsystem is sufficient for the system to function satisfactorily. In other words the system consists of redundant subsystems and will fail only if all subsystems fail.

Case 1 Let us consider a system in which a failure of any subsystem would cause a system failure. This can be modelled as a series system as shown in Fig. 1.3. If the subsystem failures are independent and R_i is the reliability of subsystem i, then the overall system reliability is

$$R_{ov} = \prod_{i=1}^{N} R_i$$

In the constant failure rate case where $R_i = \exp(-\lambda_i t)$

$$R_{ov} = \prod_{i=1}^{N} \exp(-\lambda_i t)$$

$$= \exp\left(\sum_{i=1}^{N} \lambda_i t\right)$$

Therefore the failure rate of the system is just the sum of the failure rates of the subsystems.

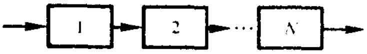

Fig. 1.3 Series system.

If the N subsystems have identical failure rates $\lambda_i = \lambda$, then $R_i = R$. Hence the overall system reliability is

$$R_{ov} = \exp(-N\lambda t)$$

$$= R^N$$

and

$$\text{MTBF} = \frac{1}{N\lambda}$$