RELIABILITY IN COMPUTER SYSTEM DESIGN

B.S. DHILLON

RELIABILITY IN COMPUTER SYSTEM DESIGN

by

B.S. Dhillon
Professor
Faculty of Engineering
University of Ottawa

Ablex Publishing Corporation, Norwood, N.J. 07648

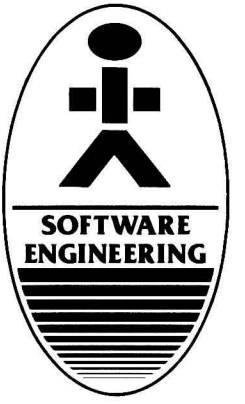
Library of Congress Cataloging-in-Publication Data

Dhillon, B. S.
Reliability in computer system design.

(Ablex series in software engineering)
Bibliography: p. ix
Includes index.
1. System design. 2. Electronic digital computers—Reliability. 3. Computer software—Reliability.
I. Title. II. Series.
QA76.9.S88D49 1987 004.2'1 87-1164
ISBN 0-89391-412-6

Ablex Publishing Corporation
355 Chestnut Street
Norwood, New Jersey 07648

RELIABILITY IN COMPUTER SYSTEM DESIGN



THE ABLEX SERIES IN SOFTWARE ENGINEERING

SERIES EDITOR

MARVIN V. ZELKOWITZ

Reliability in Computer System Design
B.S. Dhillon

in preparation

Requirements for a Software Engineering Environment
Edited by Marvin V. Zelkowitz

BIOGRAPHY OF THE AUTHOR

Dr. Dhillon is a full professor in the Faculty of Science and Engineering, University of Ottawa.

He attended the University of Wales where he received a B.Sc. in Electrical and Electronic Engineering and a M.Sc. in Industrial and Systems Engineering. His Ph.D. in Reliability Engineering was received from the University of Windsor. He wrote his doctoral thesis on reliability evaluation of networks composed of three state devices. He is Advisory Editor of "Microelectronics and Reliability: An International Journal", Associate Editor of "International Journal of Energy Systems", and Editor-at-Large for Engineering Books (Marcel Dekker, Inc.). Dr. Dhillon served as an associate editor of the 10th–13th Annual Modeling and Simulation Proceedings, Pittsburgh, Pennsylvania, USA. He has published over 170 articles on Reliability Engineering as well as nine books on various aspects of Engineering Reliability and related areas. Three of his books are translated into Russian and Chinese.

Serving as a referee to many national and international journals, book publishers and other bodies, he has presented keynote and invited lectures at various national and international conferences. Dr. Dhillon has several years of experience in electronics and nuclear power industries.

He is recipient of the American Society for Quality Control's Austin J. Bonis Reliability Award and the Society of Reliability Engineers' Merit Award, as well as several appreciation certificates from various American professional societies. A registered Professional Engineer in Ontario, Professor Dhillon is listed in the American Men and Women of Science, Dictionary of International Biography, Men of Achievement, Who's Who in Technology, Personalities of America, etc.

PREFACE

Computers are increasingly being used at an alarming rate for various purposes. Some people have predicted that those days are not very far off when the computer business will be the largest single component of the US economy. As computer failures or mistakes can effect our daily lives, their reliability is important to all of us. In addition, the reliability of computers used in critical areas such as defense, aerospace and nuclear power generation is of utmost importance because their failures could be very costly and catastrophic.

In computers, the reliability problem is associated with both hardware and software. Therefore, both these items have to be reliable for their successful operations. At the moment, to the author's knowledge, there is no published book that effectively covers the reliability of computer hardware and software within the framework of a single volume. In order to design and produce reliable computers, the knowledge in reliability of computer hardware and software is important to concerned engineers. An engineer needing information in both these two areas generally faces inconvenience because the information on these two topics is available in some texts and in various technical articles but not in a single volume. This book is an attempt to fulfil this vital need. The computer hardware and software reliability (and related) topics are treated in such a manner that the reader require no previous knowledge to understand the contents. The emphasis is on the structure of the concepts rather than the minute details. Sources for most of the material presented are given in the References. The book contains over 850 references on computer system reliability and several examples along with their solutions. The references will be useful to the reader who wishes to delve more deeply into a specific area of computer system reliability.

This book should be useful to anyone concerned with design and production of computers. In particular computer design engineers, software specialists, system engineers; reliability, quality control and safety engineers; electronics engineers, project engineers and managers, engineering administrators, and senior undergraduate and graduate students of computer science, computer engineering, reliability and quality control should benefit the most.

The book is composed of twelve chapters plus appendix. The appendix contains 717 references on computer system reliability. The references are separated into two major categories; i.e. computer hardware reliability and software reliability.

I wish to thank many leading professionals and friends whose invisible inputs have shaped my thinking on many areas of this text. The author is deeply greatful to Dr. S.N. Rayapati for the preparation of diagrams for this book. I am indebted to my relatives and friends in and outside Ottawa for their interest and encouragement throughout. Finally, I thank my wife, Rosy, for typing the entire manuscript and for her help in proofreading. During the preparation of the manuscript the disturbances of my little daughter, Jasmine, have also helped because they led to many rest and coffee breaks!

B.S. Dhillon
Ottawa, Ontario

TABLE OF CONTENTS

1
INTRODUCTION

1.1 NEED FOR RELIABILITY

Nowadays computers have become very complex and sophisticated, and their applications have increased at an alarming rate. For example, according to the *Wall Street Journal* of January 23, 1979, computer costs accounted for 1% (i.e. $25.65 billion) of the gross national product (GNP) of the United States. Furthermore, according to some predictions, the computer business in the United States will overtake the automobile business in the 1980s. This means that the computer business will be the largest single component of the U.S. economy.

Computers are used in critical areas such as aerospace, nuclear power generation, and defense. For such applications their reliability is of utmost importance because a computer failure in these areas could be very costly and catastrophic. Other factors such as increasing repair costs, harsher operating environments, use by novices, and the existence of bigger systems are also responsible for the increasing emphasis on reliability of computer systems. To improve reliability and assist field service personnel in fault isolation, computer hardware manufacturers such as International Business Machines (IBM), Amdahl, and Univac make use of redundancy. However, in computers the reliability problem is not only confined to the hardware aspect, but also extends to software. Both hardware and software have to be reliable for successful operation of a computer. Therefore there is a definite need to place emphasis on the reliability of both the computer hardware and the software.

1.2 HISTORY OF COMPUTER SYSTEM RELIABILITY

Relays and electronic tubes were extensively used in the earlier-day digital computers. In those computers, because of the poor reliability of such devices, a considerable amount of effort was directed to the areas of computer checking and self-repair [1]. In computers composed of relays, intermittent faults dominated the scene relative to permanent faults. To overcome this problem extensive use of dynamic checking was made.

The invention of the transistor may be regarded as an important milestone in the history of computer system reliability because transistors possessed a higher inherent reliability than did electronic tubes or relays. In the 1950s transistors found their way into the computer market.

Works of C.E. Shannon [2], W.R. Hamming [3], J. Von Neumann [4], and E.F. Moore and C.E. Shannon [5] have played an important role in computer system reliability. For example, it was J. Von Neumann who first proposed the replication scheme called triple modular redundancy (TMR) to improve the reliability of a system in 1956. Three years later, R. Eldred [6] considered efficient methods of test generation for combinational circuits. Efficient diagnostic tests for digital circuits were developed by S. Seshu and D.N. Freeman [7], S. Seshu [8], and J.M. Galey, R.E. Norby and J.P. Roth [9]. In 1962 a symposium on Redundancy Techniques for Computing Systems was held in Washington, D.C. The proceedings of that symposium were published by Spartan Books [10] under the editorship of R.H. Wilcox and W.C. Mann. In 1965 W.H. Pierce published a book [11] entitled "Failure Tolerant Design." This was probably the first book concerned with computer system reliability.

Over the past number of years many persons have contributed to the field of computer system reliability. Articles published by W.C. Carter and W.G. Bouricius [1], A. Avizienis [12,13], C.V. Ramamoorthy [14], M.Y. Hsiao, W.C. Carter, J.W. Thomas and W.R. Stringfellow [15], R.A. Short [16] and J. Goldberg [17] provide a very good overview of the subject. A comprehensive list of selected references is given in reference [18].

1.2.1 Software Reliability

Although the reliability of computer software is as important as that of computer hardware, effort in this direction did not surface until the 1960s. It was at Bell Laboratories [19] where the serious effort on software reliability probably started first in 1964. A histogram of problems per month reported for the first switching system software is an evidence of this effort [20]. In this case the software was composed of 100,000 words. Three years later in 1967, R.W. Floyd [21] considered methods for formal validation of software programs. In the same year G.R. Hudson [22] developed Markov birth–death models. Two conferences on software engineering, sponsored by the North Atlantic Treaty Organization (NATO), were held in West Germany and Italy in 1968 and 1969, respectively. Issues on reliable software were addressed in both these meetings [23]. In 1966, R. Barlow and E.M. Scheuer [24] published a mathematical model concerned with reliability growth (hardware system) during a development testing program. This model could also be applied for software debugging. Works of R.L. London [25] and J.L. Sauter [26] concerned with software reliability were published in 1969. Contributions of Z. Jelinski and P.B. Moranda [27], M.L. Shooman [28], G.J. Schick and R.W. Wolverton [29], N.F. Schneidewind [30], and J.D. Musa [31] are briefly discussed in Chapter 11. Many other researchers and authors have contributed to software reliability. A list of selective references is given in reference [32].

References on computer hardware reliability and software reliability listed in the appendix of this book attest the effort of other research workers and authors.

1.3 TERMS AND DEFINITIONS

This section presents selected terms and definitions used in computer system reliability [33–36].

Reliability: the probability that an item will carry out its required mission satisfactorily for a specified period of time when used according to designed conditions.

Availability: the probability that an item is functioning satisfactorily at any instant of time when used according to designed conditions. (In this situation the elements of the total time considered are the operating time, the logistic time, the active repair time, and the administrative time.)

Redundancy: the existence of two or more means for carrying out a specified function.

Failure: the termination of the ability of a unit to carry out its assigned mission.

Mean time to repair: the total corrective maintenance time over the total number of corrective maintenance actions performed during a defined time period.

Fault: an attribute which adversely affects an item's reliability.

Effectiveness: the capability of the item to carry out its assigned mission.

Debugging: the process of rectifying and isolating errors.

Software testing: the process of executing software to find out whether the results it generates are valid [35].

Fault-tolerant computing: the ability to execute given algorithms successfully regardless of computer software errors and hardware failures [12].

Software: computer program items in the form of magnetic tapes, disks or card decks as well as all kinds of descriptive documentation, including flow charts, listings, etc.

Software error: a clerical, conceptual or syntactic discrepancy which causes one or more faults in the software.

Software failure: failure which occurs in a situation when a computer program fault is elicited by some kind of input data, leading to the computer program incorrectly computing the specified function [36].

Software Fault: a discrepancy in the computer software which makes worse its capability to perform as intended.

Software reliability: the probability of a specified software functioning for a given period, without an error, when used within the framework of designed conditions on the specified machine.