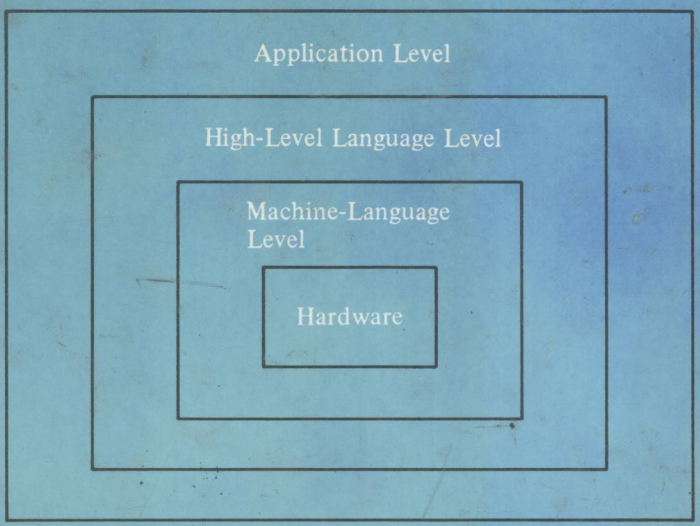
ELEMENTS OF MICRO-PROGRAMMING

DILIP K. BANERJI
JACQUES RAYMOND



ELEMENTS OF MICROPROGRAMMING

DILIP K. BANERJI
Jawaharlal Nehru University
New Delhi, India

JACQUES RAYMOND
University of Ottawa
Ottawa, Ontario, Canada

Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632

Library of Congress Cataloging in Publication Data

BANERJI, DILIP K.
Elements of microprogramming.

(Prentice-Hall software series)
Includes bibliographies and index.
1. Microprogramming. I. Raymond, Jacques.
II. Title. III. Series.
QA76.6.B347 001.64'2 81-10566
ISBN 0-13-267146-8 AACR2

Prentice-Hall Software Series
Brian W. Kernighan, advisor

Editorial/production supervision and interior design
by *Aliza Greenblatt*

Cover Design
by *Judith A. Matz*

Manufacturing buyer
Gordon Osbourne

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

To our students

PRENTICE-HALL INTERNATIONAL, INC., *London*
PRENTICE-HALL OF AUSTRALIA PTY. LIMITED, *Sydney*
PRENTICE-HALL OF CANADA, LTD., *Toronto*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*
PRENTICE-HALL OF SOUTHEAST ASIA PTE. LTD., *Singapore*
WHITEHALL BOOKS LIMITED, *Wellington, New Zealand*

Preface

This book has essentially evolved out of the course CSI 4114/ELG 5189, "Microprogramming and Machine Architecture," that the authors have offered at the University of Ottawa since 1971. Since the material for the course is derived from various sources, including some of our own work, our students repeatedly urged us to put it all together as a package, which eventually took the form of a manuscript.

The book is intended to provide a comprehensive coverage of the basic principles, practices, and applications of microprogramming. We have endeavored to stick to this objective as much as possible. Admittedly, however, as in any book, there may be a few shortcomings. For the positive aspects of the book the authors would like to share the credit with their students and colleagues who have consistently provided encouragement and support; for any shortcomings, the buck stops at our desks.

We have deliberately avoided case studies of specific microprogrammed/microprogrammable machines. With rapidly changing computer technology, machines tend to be outdated relatively quickly, thus rendering the books containing case studies out-of-date. By concentrating on basic concepts and applications of microprogramming, we hope that this book will

remain useful for a longer period. Throughout the book we have used only one microprogrammable machine, the Microdata 1600,* available at the University of Ottawa, to develop and illustrate microprogramming examples. Some readers may not agree with our choice of the machine, but this choice was motivated by the availability of the machine and by our desire to list tested microprograms.

Chapter 1 is purely introductory in nature; its main objective is to introduce the basic concepts of microprogramming. Chapter 2 traces the historical growth of microprogramming since the introduction of the concept in 1951 by Wilkes. The evolution of microprogramming up to the present stage as an integral part of computer architecture is discussed briefly. Chapter 3 reviews the basic concepts of computer organization. It is primarily intended as a refresher course in computer organization and can be skipped by readers familiar with computer hardware and organization. Chapter 4 discusses both the hardware resources required to support microprogramming and the basic schemes for implementing microprogram control and technologies used therein. Chapter 5 analyzes, in depth, the various aspects of microinstruction design, formats, and execution. It also discusses various design tradeoffs in microinstruction design.

Chapters 6 through 8 are concerned with microprogramming practice. In Chapter 6 we discuss the various aspects of developing microprograms. Starting with absolute coding, we discuss the need for a micro-assembly language, and then the need and availability of high-level languages for microprogramming. In Chapter 7 we are concerned with the microprogram implementation of some common machine instructions and functions. The example microprograms have been written for and tested on the Microdata 1600 machine. Chapter 8 discusses the effect of various factors on the performance of a microprogrammed system. In addition, it presents techniques for optimizing the size of a microprogram.

Chapters 9 through 13 essentially cover the various applications of microprogramming. In Chapter 9 we discuss the concept of emulation and show how this is achieved through microprogramming. A novel feature of this chapter is the section on I/O emulation which is generally not available in the existing texts. The emulation of an existing machine on the Microdata 1600 is described in some detail. Chapter 10 is concerned with microprogramming support for operating systems as well as high-level language processing. General principles of how this is achieved are discussed, along with a few specific examples of such applications. In Chapter 11, we discuss microprogrammable (bit-slice) microprocessors. The discussion is centered around the Intel 3000† microprocessor family which can be microprogrammed to suit user requirements. Chapter 12 describes the use of microprogramming to support block structured programming. This is

*Reprinted with permission of Microdata Corporation, Irvine, California.
†Reprinted by Permission of Intel Corporation, Copyright 1976.

illustrated by an example implemented on the Microdata 1600* machine. Finally, Chapter 13 describes some of the other applications of microprogramming: virtual memory implementation, microdiagnostics, implementation of specialized functions for computer graphics, and implementation of a communications controller. These are only a few representative examples of the other applications.

Three appendices provide useful background information to the reader. Appendix A provides a summary of the features of an emulated (target) machine discussed in Chapter 9, whereas Appendix B describes the features of the microprogrammable host machine. Appendix C provides a complete microprogram listing of the emulator discussed in Chapter 9. The References and Further Readings at the end of chapters provide a useful guide to relevant literature in the field.

The book is intended for use at Computer Science/Electrical Engineering senior and graduate levels. Chapters 1 through 8 are suitable for seniors, whereas the inclusion of Chapters 9 through 13 would make it suitable for graduate level. Some background in computer organization and some exposure to assembly language programming would be helpful in reading the book. The book can also be used as a handy reference for practicing engineers and computer professionals. It can be used either as a primary text for a full course on microprogramming or as a supplementary text in a course on computer architecture/organization. The introductory chapters can be used to supplement a basic computer organization course, while other chapters can supplement a more advanced course on architecture/organization. At the University of Ottawa it is to be used as a primary text for a full course on microprogramming.

No book can be completed without the authors receiving active support from various sources. In this context we would like to thank Data General Corporation, Microdata Corporation, and Intel Corporation for giving us permission to use some of their technical literature in the book. Many of our colleagues and students both at the University of Ottawa & JNU have helped in the preparation of the manuscript. In particular, the assistance of Peter Hickey at Ottawa in testing many of the microprograms is acknowledged. At JNU, Prasenjit Biswas, Abir Bhattacharya, and Arun Majumdar have helped a good deal. Without the assistance of these individuals, it would have been difficult to complete the book. Throughout the course of writing the book, our families have been very patient and have consistently rendered moral support. Finally, our special thanks are due to Claudette Henderson for her excellent typing.

DILIP K. BANERJI
Jawaharlal Nehru University
New Delhi, India

JACQUES RAYMOND
University of Ottawa
Ottawa, Ontario, Canada

*Reprinted with permission of Microdata Corporation, Irvine, California.

Contents

PART TWO MICROPROGRAMMING HARDWARE AND MICROINSTRUCTIONS

PART THREE MICROPROGRAMMING PRACTICE

PART FIVE EXAMPLES

PART ONE

Review

CHAPTER ONE

Basics of Microprogramming

1.1 INTRODUCTION

Since the introduction of the concept in 1951 by Wilkes [1], microprogramming has attained a great deal of acceptance and significance, not only as an alternative method of control unit design in a computer*, but also as a powerful tool in the hands of designers and users alike. The main reason for using this technique in computer control implementation is the tremendous flexibility attainable at a relatively small cost, which has made user-restructuring of computer architecture a reality. Also as Husson [2] has observed, "microprogramming has made it economically feasible to have the same comprehensive instruction set built into a whole line of new computers, even the smallest ones. Thus we now have computers that are architecturally compatible, yet their internal hardware, organization, and structure are drastically different." Besides offering the so-called upwards and downwards

*Although we talk of the control of a computer, the concept applies to any digital controller.