大学计算机教育国外著名教材、教参系列（影印版）

The 80x86 IBM PC and Compatible Computers (Volumes I&II)

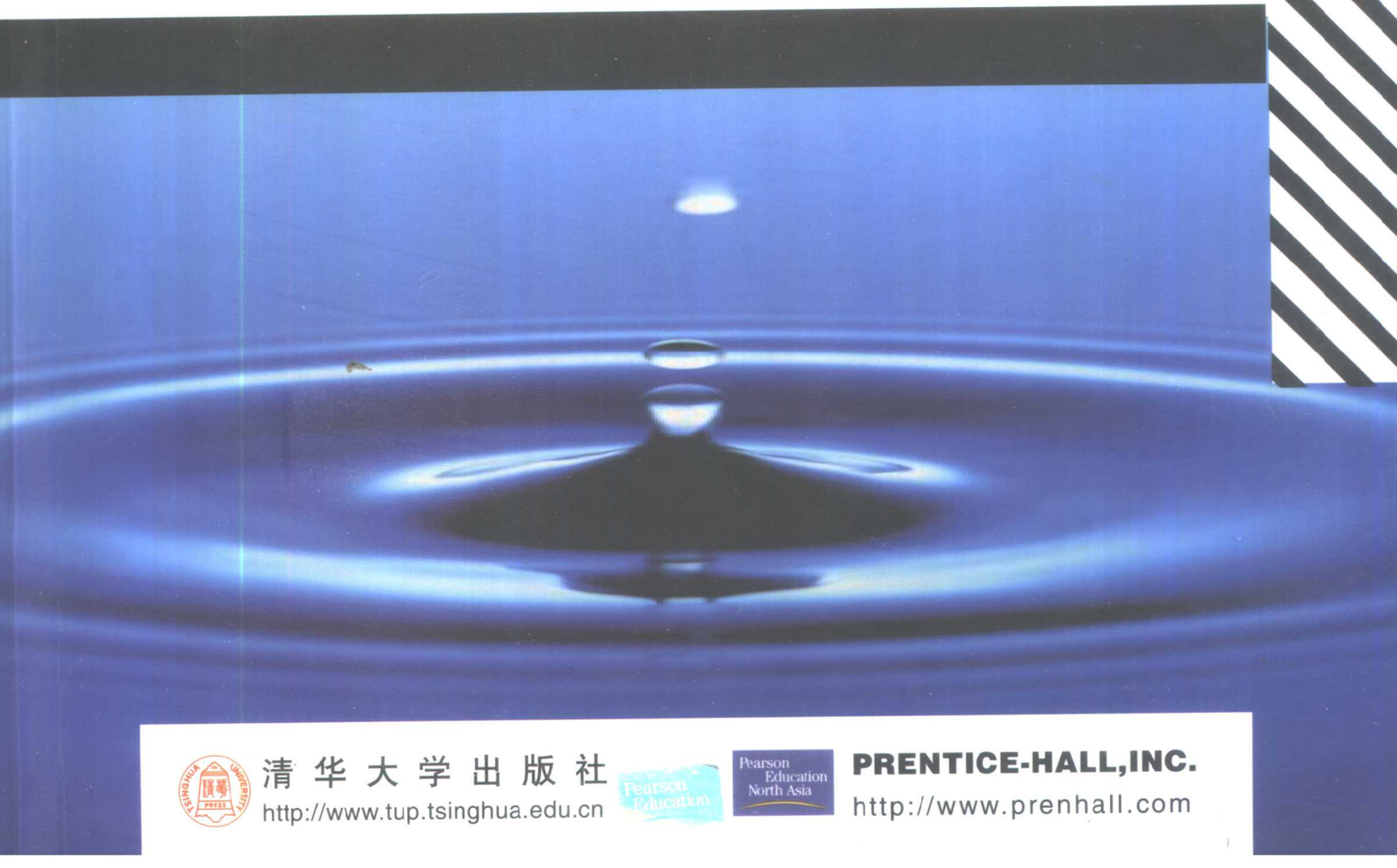
Assembly Language, Design and Interfacing 3rd ed.

Muhammad Ali Mazidi
Janice Gillispie Mazidi

80x86 IBM PC及兼容计算机(卷Ⅰ和卷Ⅱ)

汇编语言，设计与接口技术

（第3版）



清华大学出版社
http://www.tup.tsinghua.edu.cn
Pearson Education
Pearson Education North Asia
PRENTICE-HALL,INC.
http://www.prenhall.com

THE 80x86 IBM PC
AND COMPATIBLE COMPUTERS
VOLUMES Ⅰ & Ⅱ

Assembly Language, Design and Interfacing

Third Edition

80x86 IBM PC 及兼容计算机

卷Ⅰ和卷Ⅱ

汇编语言，设计与接口技术

第 3 版

Muhammad Ali Mazidi

Janice Gillispie Mazidi

清华大学出版社　　**Prentice-Hall, Inc.**

（京）新登字 158 号

The 80x86 IBM PC and Compatible Computers Volumes Ⅰ & Ⅱ
Assembly Language, Design, and Interfacing 3rd ed.
Muhammad Ali Mazidi, Janice Gillispie Mazidi

北京市版权局著作权合同登记号：图字：01-2002-0657

书　名：80x86 IBM PC 及兼容计算机卷 I 和卷 II：汇编语言，设计与接口技术（第 3 版）
作　者：M. A. Mazidi, J. G. Mazidi
出版者：清华大学出版社（北京清华大学学研大厦，邮编 100084）
http:// www.tup.tsinghua.edu.cn
印刷者：北京人民文学印刷厂
发行者：新华书店总店北京发行所
开　本：880×1230 1/16　印张：63.75
版　次：2002 年 6 月第 1 版　2002 年 6 月第 1 次印刷
书　号：ISBN 7-302-04999-8/TP・2819
印　数：0001～3000
定　价：85.00 元

出 版 说 明

进入21世纪，世界各国的经济、科技以及综合国力的竞争将更加激烈。竞争的中心无疑是对人才的争夺。谁拥有大量高素质的人才，谁就能在竞争中取得优势。高等教育，作为培养高素质人才的事业，必然受到高度重视。目前我国高等教育的教材更新较慢，为了加快教材的更新频率，教育部正在大力促进我国高校采用国外原版教材。

清华大学出版社从1996年开始，与国外著名出版公司合作，影印出版了“大学计算机教育丛书（影印版）”等一系列引进图书，受到了国内读者的欢迎和支持。跨入21世纪，我们本着为我国高等教育教材建设服务的初衷，在已有的基础上，进一步扩大选题内容，改变图书开本尺寸，一如既往地请有关专家挑选适用于我国高校本科及研究生计算机教育的国外经典教材或著名教材以及教学参考书，组成本套“大学计算机教育国外著名教材、教参系列（影印版）”，以飨读者。深切期盼读者及时将使用本系列教材、教参的效果和意见反馈给我们。更希望国内专家、教授积极向我们推荐国外计算机教育的优秀教材，以利我们把“大学计算机教育国外著名教材、教参系列（影印版）”做得更好，更适合高校师生的需要。

计算机引进版图书编辑室

2002.3

PREFACE TO THE SERIES

"I think that Intel has some of its greatest times ahead of it. That's because they are driving microprocessor design technology and enjoy the largest installed base of software in the world. If you're going to learn only one instruction set, it's going to be the Intel X86." *

Philippe Kahn
Founder, Borland International, Inc.

It is currently estimated that there are over 100 million 80x86-based (8088, 8086, 80286, 80386, 80386SX, 80486, 80486SX, Pentium) IBM and compatible computers in the world and this number is growing by 40 to 50 million units a year. The alliance of Intel, IBM, and Microsoft brought about a revolution in the computer industry by creating a unified system that became the standard for desktop computers. Intel provided the 80x86 microprocessors and Microsoft developed the DOS operating system, but it was IBM who set the revolution in motion by making the architecture of the PC open for cloning. In the absence of such a role by IBM, we would have desktop computers with four or five different architectures and operating systems, all incompatible with each other. This would have been more like the tower of Babel than the friendly world of IBM PCs and compatibles that we have known and enjoyed since 1981 when the first IBM PC was announced. The fact that the newer-generation 80x86 CPUs are achieving the power of minicomputers will assure the survival of the 80x86 well beyond the year 2000. These facts explain why many companies such as Sun Micro and Next have made available an 80x86 version of their operating systems.

Why this series?

It is our belief that many computer hardware and software concepts are much easier to learn if one has access to a system whereby these concepts can be experimented with hands-on. Undoubtedly, the 80x86-based PC is the most affordable tool to achieve this objective. The steadily decreasing price of PCs has made these tools available to schools, students, individuals, and small businesses.

Although there are many fine books that deal with various hardware or software aspects of the PC, this series is designed to provide a systematic and comprehensive introduction to both the software and hardware of the PC. We have embarked on the task of creating this series of books which will provide a guide to those wanting to become proficient in the PC. The range of topics selected and their degree of coverage have been designed based on over ten years of classroom experience introducing these concepts to students. Emphasis has been placed on providing information in such a way as to enable the student to gain hands-on experience quickly in order to master the concepts as they are presented.

More about this volume

Volume 1 of this series provides an introduction to Assembly language programming on the PC, and Volume 2 covers the hardware design and interfacing of 80x86 systems. This combined volume includes Volumes 1 and 2 in their entirety.

* "The Empire Strikes Back," Upside, June 1992, p. 42.

PREFACE TO VOLUMES I AND II

Purpose

This combined volume is intended for use in college-level courses in which both Assembly language programming and 80x86 PC interfacing are discussed. It not only builds the foundation of Assembly language programming, but also provides a comprehensive treatment of 80x86 PC design and interfacing for students in engineering and computer science disciplines. This volume is intended for those who wish to gain an in-depth understanding of the internal working of the IBM PC, PS, and 80x86 compatible computers. It builds a foundation for the design and interfacing of microprocessor-based systems using the real-world example of the 80x86 IBM PC. In addition, it can also be used by practicing technicians, hardware engineers, computer scientists, and hobbyists who want to do PC interfacing and data acquisition.

Prerequisites

Readers should have a minimal familiarity with the IBM PC and the DOS operating system in addition to having had an introductory digital course. Knowledge of other programming languages would be helpful, but is not necessary.

Although a vast majority of current PCs use 386, 486, or Pentium microprocessors, their design is based on the IBM PC/AT, an 80286 microprocessor system introduced in 1984. A good portion of PC/AT features, hence its limitations, are based on the original IBM PC, an 8088 microprocessor system, introduced in 1981. In other words, one cannot expect to understand fully the architectural philosophy of the 80x86 PC and its expansion slot signals unless the 80286 PC/AT and its subset, the IBM PC/XT, are first understood. For this reason, we describe the 8088 and 80286 microprocessors in Chapters 9 and 10. In doing so, we describe the purpose and use of the supporting chips of the 8088, 80286 microprocessor such as the 8288, 8284, 82288, and 82284. Although these supporting chips provide the necessary timing for the 8088/86/286 processors, they are no longer used in later generation 386/486/Pentium microprocessors, since their functions are incorporated into the CPU.

Contents of Volume I

A systematic, step-by-step approach has been used in covering various aspects of Assembly language programming. Many examples and sample programs are given to clarify concepts and provide students an opportunity to learn by doing. Review questions are provided at the end of each section to reinforce the main points of the section. We feel that one of the functions of a textbook is to familiarize the student with terminology used in technical literature and in industry, so we have followed that guideline in this text.

Chapter 0 covers concepts in number systems (binary, decimal, and hex) and computer architecture. Most students will have learned these concepts in previous courses, but Chapter 0 provides a quick overview for those students who have not learned these concepts, or who may need to refresh their memory.

Chapter 1 provides a brief history of the evolution of 80x86 microprocessors and an overview of the internal workings of the 8086 as a basis of all 80x86 processors. Chapter 1 should be used in conjunction with Appendix A (a tutorial introduction to DEBUG) so that the student can experiment with concepts being learned on the PC. The order of topics in Appendix A has been designed to correspond to the order of topics presented in Chapter 1. This allows the student to begin programming with DEBUG without having to learn how to use an assembler.

Chapter 2 explains the use of assemblers to create programs. Although the programs in the book can be used with Microsoft's MASM assembler, any Intel-compatible assembler such as Borland's TASM will also do.

Chapter 3 introduces the bulk of the logic and arithmetic instructions for unsigned numbers, plus bitwise operations and bit manipulation in C.

Chapter 4 introduces DOS and BIOS interrupts. Programs in Assembly and C allow the student to get input from the keyboard and send output to the monitor. In addition, interrupt programming in C is described, as well as how to put Assembly language code in C programs.

Chapter 5 describes how to use macros to develop Assembly language programs in a more time-efficient and structured manner. We also cover INT 33H mouse function calls and mouse programming.

Chapter 6 covers arithmetic and logic instructions for signed numbers as well as string processing instructions.

Chapter 7 discusses modular programming and how to develop larger Assembly language programs by breaking them into smaller modules to be coded and tested separately. In addition, linking Assembly language modules with C programs is thoroughly explained.

Chapter 8 introduces some 32-bit concepts of 80386 and 80486 programming. Although this book emphasizes 16-bit programming, the 386/486 is introduced to help the student appreciate the power of 32-bit CPUs. Several programs are run across the 80x86 family to show the dramatic improvement in clock cycles with the newer CPUs.

Contents of Volume II

Chapter 9 describes the 8088/86 microprocessor and supporting chips in detail and shows how they are used in the original IBM PC/XT. In addition, the origin and function of the address, data, and control signals of the PC/XT expansion slot are described.

In Chapter 10, the 80286 microprocessor and its supporting chips are examined in detail. In addition, we examine the origin of the signals of the PC/AT expansion slot, commonly known as the ISA bus.

Chapter 11 provides an introduction to various types of RAM and ROM memories, their interfacing to the microprocessor, the memory map of the 80x86 PC, the timing issue in interfacing memory to the CPU, and the checksum byte and parity bit techniques of ensuring data integrity in RAM and ROM.

Chapter 12 is dedicated to the interfacing of I/O ports, the use of IN and OUT instructions in the 80x86, and interfacing and programming of the 8255 programmable peripheral chip. We also cover the PC Interface Trainer and Bus Extender, which are used to interface PCs to devices for data acquisition such as LCDs, stepper motors, ADC, DAC, and sensors. In addition, programming I/O with C language is covered.

Chapter 13 discusses the use of the 8253/54 timer chip in the 80x86 PC, as well as how to generate music and time delays.

Chapter 14 is dedicated to the explanation of hardware and software interrupts, the use of the 8259 interrupt controller, the origin and assignment of IRQ signals on the expansion slots of the ISA bus, and exception interrupts in 80x86 microprocessors.

Chapter 15 is dedicated to direct memory access (DMA) concepts, the use of the 8237 DMA chip in the 80x86 PC, and DMA channels and associated signals on the ISA bus.

Chapter 16 covers the basics of video monitors and various video modes and adapters of the PC, in addition to the memory requirements of various video boards in graphics mode.

Chapter 17 discusses serial communication principles, the interfacing and programming of National Semiconductor's 8250/16450/16550 UART chip, Intel's 8251 USART chip, and verifying data integrity using the CRC method.

Chapter 18 covers the interfacing and programming of the keyboard in the 80x86 PC, in addition to printer port interfacing and programming. In addition, a discussion of various types of parallel ports such as EPP and ECP is included.

Chapter 19 discusses both floppy and hard disk storage organization and terminology. We also show how to write Assembly language programs to access files using INT 21H DOS function calls.

Chapter 20 examines the 80x87 math coprocessor, its programming and interfacing, and IEEE single and double precision floating point data types.

Chapter 21 explores the programming and hardware of the 386 microprocessor, contrasts and explains real and protected modes, and discusses the implementation of virtual memory.

Chapter 22 is dedicated to the interfacing of high-speed memories and describes various types of DRAM, including EDO, SDRAM, and Rambus, and examines cache memory and various cache organizations and terminology in detail.

In Chapter 23 we describe the main features of the 486, Pentium, and Pentium Pro and compare these microprocessors with the RISC processors. Chapter 23 also provides a discussion of MMX technology and how to write programs to detect which CPU a PC has.

Chapter 24 describes the MS DOS structure and the role of CONFIG.SYS and batch files in the 80x86 PC, the writing of TSR (terminate and stay resident) programs, and device drivers.

Chapter 25 explains 80x86 PC memory terminology, such as conventional memory, expanded memory, upper memory block, and high memory area, as well as MS DOS memory management.

Chapter 26 provides an overview of IC technology including the recent advances in IC fabrication, describes IC interfacing and system design issues, and covers error detection and correction.

Chapter 27 is dedicated to the discussion of the various types of PC buses, such as ISA, EISA, USB, their performance comparisons, the local bus, and features of the PCI local bus.

In Chapter 28 we show how to use C language to access DOS function calls, BIOS interrupts, memory, input/output ports, and CMOS RAM of the 80x86.

Appendices

The appendices have been designed to provide all reference material required for the topics covered in this combined volume so that no additional references should be necessary.

Appendix A provides a tutorial introduction to DEBUG.
Appendix B provides a listing of Intel's 8086 instruction set along with clock cycles for 80x86 microprocessors.
Appendix C describes assembler directives with examples of their use.
Appendix D lists some commonly used DOS 21H function calls and INT 33H mouse functions.
Appendix E lists the function calls for various BIOS interrupts.
Appendix F provides a table of ASCII codes.
Appendix G lists the I/O map of 80x86-based ISA computers.
Appendix H provides a description of the BIOS data area.
Appendix I contains data sheets for various IC chips.

Diskette

There is a diskette attached to this combined volume (Volumes I and II) that provides the source code for programs and examples in the textbook. The files on the diskette are in ASCII format.

Lab Manual

There is a lab manual for this combined volume (Volumes I and II). The section for Volume I covers Assembly language programming using DEBUG and assemblers such as Microsoft's MASM and Borland's TASM. It includes 24 labs covering data types, arithmetic operations, string handling, graphics programming, 32-bit programming, and macros. In addition, there are 10 advanced labs involving more complex programming techniques such as sorting, advanced calculations, data structures and manipulation. The section for Volume II begins by exploring system programming using DEBUG, assemblers, and 32-bit programming features available in MicroSoft's CodeView and Borland's Turbo Debugger. Then it describes how to wire-wrap a PC bus extender in order to access signals on the expansion slot. This bus extender is used to interface devices to the PC such as LCDs and LEDs, ADC and DAC converters, sensors, printers, and more.

Acknowledgments

This book is the result of the dedication, work, and love of many individuals. Our sincere and heartfelt appreciation goes out to all of them. First, we must thank the reviewers who provided valuable suggestions and encouragement: Mr. William H. Shannon of the University of Maryland, Mr. Howard W. Atwell of Fullerton College, Mr. David G. Delker of Kansas State University, Mr. Michael Chen of Duchess Community College, Mr. Yusuf Motiwala of Prairie View A&M University, and Mr. Donald T. Coston of ITT Technical Institute. We were truly amazed by the depth and breadth of their knowledge of microprocessor-based system design in general and 80x86 PC architecture in particular. We sincerely appreciate their comments and suggestions. Some of their suggestions are incorporated in the lab book due to lack of space in this volume.

Thanks also must go to the many students whose comments have helped shape this book, especially Daniel Woods, Sam Oparah, Herbert Sendeki, Greg Boyle, Philip Fitzer, Adnan Hindi, Kent Keeter, Mark Ford, Shannon Looper, Mitch Johnson, Carol Killelea, Michael Madden, Douglas McAlister, David Simmons, Dwight Brown, Clifton Snyder, Phillip Boatright, Wilfrid Lowe, Robert Schabel, and the Class of '94, who helped find errors in the prepublication draft.

A word must also be said of our colleagues, especially the late Mr. Allan Escher, whose encouragement set the making of this series into motion. For the last 25 years, his dedication and love of microprocessor education were a source of inspiration to many. A special thanks goes to Mr. James Vignali for his enthusiasm in discussing the internal intricacies of the 80x86 PC and his readiness to keep current with the ever-changing world of the PC.

Special thanks go to Tom Selgas, Larry Tittle, and the technical support staff of Cyrix Corporation for their valuable input.

Finally, we offer our appreciation for the dedicated professionals at Prentice Hall. Many thanks to Charles Stewart for his continued support of this series.

Acknowledgments for this edition

First, we would like to sincerely thank the following professors from some outstanding engineering schools whose enthusiasm for the book, suggestions, and kind words have been encouraging to us and made us think we are on the right track: Dr. Michael Chwialkowski (Electrical Engineering Dept., University of Texas at Arlington), Dr. Roger S. Walker (Computer Science Engineering Dept., University of Texas at Arlington), Dr. Behbood Zoghi (Electronics Engineering Technology, Texas A&M University).

We would also like to thank the following individuals who made suggestions for this new edition and sent us corrections for the errors in the first edition: John Berry, Clyde Knight, Robert Jones (all of DeVry Institute of Technolgy), Lynnette Garetz (Heald College), Peter Woof (Southern Sydney Institute, Lidcombe College of Tafe), M. Soleimanzadeh, Mark Lessley, Snehal Amin, Travis Erck, Gary Hudson.

ABOUT THE AUTHORS

Muhammad Ali Mazidi holds Master's degrees from both Southern Methodist University and the University of Texas at Dallas, and currently is completing his Ph.D. in the Electrical Engineering Department of Southern Methodist University. He is a co-founder and chief researcher of Microprocessor Education Group, a company dedicated to bringing knowledge of microprocessors to the widest possible audience. He also teaches microprocessor-based system design at DeVry Institute of Technology in Dallas, Texas.

Janice Gillispie Mazidi has a Master of Science degree in Computer Science from the University of North Texas. After several years experience as a software engineer in Dallas, she co-founded Microprocessor Education Group, where she is the chief technical writer and production manager, and is responsible for software development and testing.

The Mazidis have been married since 1985 and have two sons, Robert Nabil and Michael Jamal.

The authors can be contacted at the following address if you have any comments or suggestions, or if you find any errors.

Microprocessor Education Group
P.O. Box 381970
Duncanville, TX 75138

email: mmazidi@dal.devry.edu

CONTENTS AT A GLANCE

CHAPTERS

Assembly Language Programming on the IBM PC, PS, and Compatibles

Design and Interfacing of the IBM PC, PS, and Compatibles

APPENDICES

CONTENTS