PREFACE

This book examines the hardware and software features of the MCS-51 family of microcontrollers. The intended audience is college or university students of electronics or computer technology, electrical or computer engineering, or practicing technicians or engineers interested in learning about microcontrollers

The means to effectively fulfill that audience's informational needs were tested and refined in the development of this book. In its prototype form, *The 8051 Microcontroller* has been the basis of a fifth semester course for college students in computer engineering. As detailed in Chapter 10, students build an 8051 single-board computer as part of this course. That computer, in turn, has been used as the target system for a final, sixth semester "project" course in which students design, implement, and document a "product" controlled by the 8051 microcontroller and incorporating original software and hardware.

Since the 8051—like all microcontrollers—contains a high degree of functionality, the book emphasizes architecture and programming rather than electrical details. The software topics are delivered in the context of Intel's assembler (ASM51) and linker/locator (RL51).

It is my view that courses on microprocessors or microcontrollers are inherently more difficult to deliver than courses in, for example, digital systems, because a linear sequence of topics is hard to devise. The very first program that is demonstrated to students brings with it significant assumptions, such as a knowledge of the CPU's programming model and addressing modes, the distinction between an address and the content of an address, and so on. For this reason, a course based on this book should not attempt to follow strictly the sequence presented. Chapter 1 is a good starting point, however. It serves as a general introduction to microcontrollers, with particular emphasis on the distinctions between microcontrollers and microprocessors.

Chapter 2 introduces the hardware architecture of the 8051 microcontroller, and its counterparts that form the MCS-51 family. Concise examples are presented using short sequences of instructions. Instructors should be prepared at this point to introduce, in parallel, topics from Chapters 3 and 7 and Appendices A and C to support the requisite software knowledge in these examples. Appendix A is particularly valuable, since it contains in a single figure the entire 8051 instruction set.

Chapter 3 introduces the instruction set, beginning with definitions of the 8051's addressing modes. The instruction set has convenient categories of instructions (data transfer, branch, etc.) which facilitate a step-wise presentation. Numerous brief examples demonstrate each addressing mode and each type of instruction.

Chapters 4, 5, and 6 progress through the 8051's on-chip features, beginning with the timers, advancing to the serial port (which requires a timer as a baud rate generator), and concluding with interrupts. The examples in these chapters are longer and more complex than those presented earlier. Instructors are wise not to rush into these chapters: it is essential that students gain solid understanding of the 8051's hardware architecture and instruction set before advancing to these topics.

Many of the topics in Chapter 7 will be covered, by necessity, in progressing through the first six chapters. Nevertheless, this chapter is perhaps the most important for developing in students the potential to undertake large-scale projects. Advanced topics such as assemble-time expression evaluation, modular programming, linking and locating, and macro programming will be a significant challenge for many students. At this point the importance of hands-on experience cannot be over-emphasized. Students should be encouraged to experiment by entering the examples in the chapter into the computer and observing the output and error messages provided by ASM51, RL51, and the object-to-hex conversion utility (OH).

Some advanced topics relating to programming methods, style, and the development environment are presented in Chapters 8 and 9. These chapters address larger, more conceptual topics important in professional development environments.

Chapter 10 presents several design examples incorporating selected hardware with supporting software. The software is fully annotated and is the real focus in these examples. The second edition includes two additional interfaces; a digital-to-analog output interface using an MC1408 8-bit DAC, and an analog-to-digital input interface using an ADC0804 8-bit ADC. One of the designs in Chapter 10 is the SBC-51—the 8051 single-board computer. The SBC-51 can form the basis of a course on the 8051 microcontroller. A short monitor program is included (see Appendix G) which is sufficient to get "up and running." A development environment also requires a host computer which doubles as a dumb terminal for controlling the SBC-51 after programs have been downloaded for execution.

Many dozens of students have wire-wrapped prototype versions of the SBC-51 during the years that I have taught 8051-based courses to computer engineering students. Shortly after the release of the first edition of this text, URDA, Inc. (Pittsburgh, Pennsylvania) began manufacturing and marketing a PC-board version of the SBC-51. This has proven to be a cost-effective solution to implementing a complete lecture-plus-lab package for teaching the 8051 microcontroller to technology students. Contact URDA at 1-800-338-0517 for more information.

Finally, each chapter contains questions further exploring the concepts presented. This new edition includes 128 end-of-chapter questions—almost double the number in the first edition. A solutions manual is available to instructors from the publisher.

The book makes extensive use of, and builds on, Intel's literature on the MCS-51 devices. In particular, Appendix C contains the definitions of all 8051 instructions and Appendix E contains the 8051 data sheet. Intel's cooperation is gratefully acknowledged. I also thank the following persons who reviewed the manuscript and offered invaluable comments, criticism, and suggestions: Antony Alumkal, Austin Community College; Omer Farook, Purdue University—Calumet; David Jones, Lenoir Community College; Roy Seigel, DeVry Institute; and Chandra Sekhar, Purdue University—Calumet.

I. Scott MacKenzie

CONTENTS

APPENDIXES

1

INTRODUCTION TO MICROCONTROLLERS

1.1 INTRODUCTION

Although computers have only been with us for a few decades, their impact has been profound, rivaling that of the telephone, automobile, or television. Their presence is felt by us all, whether computer programmers or recipients of monthly bills printed by a large computer system and delivered by mail. Our notion of computers usually categorizes them as "data processors," performing numeric operations with inexhaustible competence.

We confront computers of a vastly different breed in a more subtle context performing tasks in a quiet, efficient, and even humble manner, their presence often unnoticed. As a central component in many industrial and consumer products, we find computers at the supermarket inside cash registers and scales; at home in ovens, washing machines, alarm clocks, and thermostats; at play in toys, VCRs, stereo equipment, and musical instruments; at the office in typewriters and photocopiers; and in industrial equipment such as drill presses and phototypesetters. In these settings computers are performing "control" functions by interfacing with the "real world" to turn devices on and off and to monitor conditions. **Microcontrollers** (as opposed to microcomputers or microprocessors) are often found in applications such as these.

It's hard to imagine the present world of electronic tools and toys without the microprocessor. Yet this single-chip wonder has barely reached its twentieth birthday. In 1971 Intel Corporation introduced the 8080, the first successful microprocessor. Shortly thereafter, Motorola, RCA, and then MOS Technology and Zilog introduced similar devices: the 6800, 1801, 6502, and Z80, respectively. Alone these integrated circuits (ICs) were rather helpless (and they remain so); but as part of a single-board computer (SBC) they became the central component in useful products for learning about and designing with microprocessors. These SBCs, of which the *D2* by Motorola, *KIM-1* by MOS Technology, and *SDK-85* by Intel are the most memorable, quickly found their way into design labs at colleges, universities, and electronics companies.

A device similar to the microprocessor is the microcontroller. In 1976 Intel introduced the 8748, the first device in the MCS-48™ family of microcontrollers. Within a single integrated circuit containing over 17,000 transistors, the 8748 delivered a CPU,