

# **Microcomputer Interfacing**

**G. Jack Lipovski**



**Lexington Books**

# **Microcomputer Interfacing**

**Principles and Practices**

**G. Jack Lipovski**  
University of Texas

**LexingtonBooks**  
D.C. Heath and Company  
Lexington, Massachusetts  
Toronto

**Library of Congress Cataloging in Publication Data**

Lipovski, G. Jack  
Microcomputer interfacing.

1. Microcomputers. 2. Computer interfaces. I. Title.  
TK7888.3.L49            001.64'404            79-9683  
ISBN 0-669-03619-6

*Copyright © 1980 by D.C. Heath and Company*

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage or retrieval system, without permission in writing from the publisher.

Fourth printing, August 1982

Published simultaneously in Canada

Printed in the United States of America

International Standard Book Number: 0-669-03619-6

Library of Congress Catalog Card Number: 79-9683

# **Microcomputer Interfacing**

# Lexington Books Series in Computer Science

*Kenneth J. Thurber*, General Editor

## **Basic Computer Logic**

*John R. Scott*

## **Comparative Studies in Software Acquisition**

*Steven Glaseman*

## **Computing in the Humanities**

*Peter C. Patton and Renee A. Holoien*

## **Data Base Computers**

*Olin H. Bray and Harvey A. Freeman*

## **Database Management System Anatomy**

*James A. Larson*

## **Data Structures and Computer Architecture**

*Kenneth J. Thurber and Peter C. Patton*

## **Distributed Database Management Systems**

*Olin H. Bray*

## **Distributed-Processor Communications Architecture**

*Kenneth J. Thurber and G.M. Masson*

## **Interconnection Networks for Large-Scale Parallel Processing**

*Howard J. Siegel*

## **Local Networks**

*W.R. Franta and Imrich Chlamtac*

## **Microcomputer Interfacing**

*G. Jack Lipovski*

## **Requirements-Oriented Computer System Design**

*Kenneth J. Thurber*

## **Software Blueprint and Examples**

*Yaohan Chu*

## **User-Designed Computing**

*Louis Schlueter, Jr.*

*Dedicated to my wonderful wife*

*Belle*

## Preface

The microcomputer has changed so many aspects of engineering that the cliché “microcomputer revolution” appears in numerous magazine articles and learned papers in the field. It is a great tool. Microcomputer-systems designers, however, have had to develop their own foundations, based on trade magazine articles and applications notes, because no available textbooks or reference books have specifically developed those foundations. This book aims to teach the principles and practices of microcomputer-systems design in general and interfacing in particular.

This book was developed from a set of notes for a senior-level course in microcomputer design. The course focuses on the combined hardware-software design of microcomputer systems. It emphasizes the principles—the theory—of design because that is as necessary for a solid foundation for system design as is theory in any engineering discipline. Even so, it also emphasizes the practices—the details of how to get a system to work—because microcomputer-system design requires hands-on experience. The difference between a student who merely reads about microcomputers and a student who has worked with one is clear evidence that theory must be taught with practice. Practical experience is desirable in almost any engineering course. This is not always possible. But microcomputer systems are inexpensive enough that the school or the student can afford this hands-on opportunity, and the joy of seeing the principles work is so attractive that the student often cannot get enough of the material. So the course and this book emphasize both the principles and the practices of microcomputer design.

These principles and practices must cover both hardware and software. A purely hardware approach might be more attractive to a seasoned digital-system designer or an electrical engineering student, but that approach leads to ignorant choices that either do not take full advantage of the tremendous power of software or that force unnecessary constraints and therefore higher costs on its development. On the other hand, a purely software approach misses the opportunity to understand how and why things happen and how to take advantage of these things.

A combined hardware-software approach does require more background. The course on which this book is based is the second course of a two-course sequence. The first course teaches basic assembler-language programming. It is really just a standard computer-science course on assembler-language programming, but it is taught with an assembler language of a microcomputer. The second course builds upon that background. It also presumes some background in logic design, such as would be obtained in a standard introductory course on that topic. This

book, however, has two chapters that survey the two prerequisite courses. These two chapters can be skimmed as a review of the required background material, or they can be carefully studied as a condensed tutorial if the reader has not had the earlier courses or their equivalent material. The two chapters are comparatively compressed and terse, because they are intended as review or intensive tutorial material preliminary to the main subject of the book, which follows in the later chapters.

The practices discussed in this book are made concrete through detailed discussion of the Motorola M6800 family of microcomputers and the M6809 microprocessor instruction set. These are used as a means to the end of teaching principles and practices in general, rather than for the promotion, sale, or use of Motorola products. Applications, notes, and catalogs are available from Motorola to that end. Specific and detailed discussion encourages and aides the reader in learning through hands-on experience, and it vitally contributes to an enthusiasm for and understanding of the principles. The M6800 family is used primarily because the M6809 is believed to be the most easily taught microcomputer. Its instruction set is as complete as that of any other machine, supporting enough addressing modes and index registers to teach the intelligent use of data structures, and it is symmetrical and comparatively free from quirks and anomalies that detract from the subject under discussion. Nevertheless, detailed comparisons between the M6809 and other well-designed microcomputers clearly show that others may be better than the M6809 for different applications. We do not mean to imply by the selection of the M6809 that it is the best microcomputer for all applications. We want to stress this point to the reader; however, a comparative study of different microcomputers and applications is beyond the scope of this book. On the other hand, the M6800 family, and the M6809 microcomputer in particular, are excellent vehicles for teaching the principles of microcomputer design.

The first two chapters quickly survey the background needed for the remainder of the book. Chapter 1 covers the instruction set of the M6809. It is intended as a survey for a reader who is acquainted with some other assembler language, either of another microcomputer or of a large machine. Chapter 2 covers basic computer organization but confines itself to those aspects of the topic that are particularly germane to microcomputers. For example, basic computer organization traditionally covers floating-point arithmetic, but this chapter does not; and this chapter dwells on tristate buses, although these are not often covered in discussions of computer organization. The rest of the book covers three basic themes: parallel input/output (I/O) ports in chapters 3 and 4, analog components in chapters 5 and 6, and communications devices in chapter 7. The simple parallel I/O port is displayed in chapter 3. Hardware and software aspects are studied



side by side, so that the reader need no longer be intimidated by the basic interface. Chapter 4 discusses interrupts and their alternatives. Hardware-software tradeoffs are analyzed, and different techniques are exemplified. Chapter 5 surveys the traditional (voltage) analog components that are commonly used in microcomputer I/O systems. Sufficient information is provided so that the reader can understand the uses, the limitations, and the advantages of analog components and can use this chapter as a springboard to other texts, magazine articles, or discussions with colleagues with a fuller understanding of analog design. Chapter 6 introduces the counter-timer as an interface to frequency-analog signals. Techniques to generate signals with a frequency, or to measure a signal with a frequency, that is analog to some quantity are discussed. Moreover, the hardware-software alternatives to using this most interesting integrated circuit are discussed and evaluated. Chapter 7 describes communications devices. The Universal Asynchronous Receiver Transmitter (UART) and related devices are thoroughly studied, and other communications techniques are described.

This book is intended to be complete in itself, with no additional books necessary. If the reader wishes to skip the problems at the end of each chapter, and to omit doing the laboratory experiments suggested in the text, this book is sufficient. However, we recommend doing the problems and some laboratory experiments, and to do so the reader will need a copy of *The Complete Motorola Microcomputer Data Library*, printed after 1 June 1980. (Earlier printings have an incomplete description of the 6809.) Most readers will either have that book or will want to acquire a copy for further information on the chips that can be interfaced to the M6809. However, our two appendixes on an Intersil UART and a Motorola addressable UART contain material that is not in the *Data Library* and will allow the reader to do all the problems. The reader may also want to refer to *The TTL Data Book for Design Engineers* published by Texas Instruments, Inc. Also, we are planning a laboratory manual (which we hope will be available by December 1980) and a solutions manual, which may be obtained by contacting the author.

Some remarks are necessary on the style of this book. Terms are formally introduced and used as carefully as possible, and an extensive index is provided to help locate explanations of these terms. This is necessary in order to make some sense out of a subject as broad and rich as microcomputer-system design. There is no room for muddy terminology or the use of undefined jargon. Even though the topic is under rapid development and the terminology used in trade journals and manufacturers' applications notes is inconsistent and often contradictory, the terminology used in a text must be clear and consistent. However, a book full of definitions is too hard to read. The first version of the course notes that led to this book tended to

be ponderous. Moreover, today's students are more attuned to television colloquialism and are tuned out to "third-person boring," which is often the style of learned textbooks. So we condescend to first-person conversational, and we enjoy it. The "we" in this book stands not only for the author but also for his colleagues, his teachers, and his students, who have taught him a great deal and who have collectively inspired and developed the principles and practices we discuss in the book. But we admit to exploring Webster's *Collegiate* occasionally for just the right word because we enjoy the challenge, and we even allow a pun or two where it does not interfere with the presentation of the material. We cannot deny it—microcomputer design is fun, and we have fun talking about it. Please forgive us if it shows in the style of the book.

## Acknowledgments

The author would like to express his deepest gratitude to all the people who have contributed to this book. In addition to a number of students, fellow faculty members, and colleagues in industry who have helped in no small part, special thanks are due to Professor Charles Roth, who taught the course using these notes and who corrected many errors and contributed some of the better problems at the end of each chapter. Deep thanks are given Terry Ritter at Motorola, one of the designers of the M6809, for his extensive review of the manuscript, and for the many corrections and suggestions that make the book so much better. I would like to thank Lance Leventhal and Harold Stone for some fine comments in their reviews of the manuscript; these comments were invaluable. Gratitude is also offered to Professor Terry Wagner and students Ed Upchurch and Jeff Oliver, who proofread some of the chapters. Finally, I acknowledge the support of my secretary, Janelle Dolby, for correcting errors on the text edition, and my technician, Pat Horne, for preparing some of the rough-draft figures. If I omitted anyone, may I just say that throughout the book, the word *we* refers to me and to all of you who have helped in the formation of the ideas and in the generation of this book.

# Contents

	<b>List of Figures</b>	ix
	<b>List of Tables</b>	xiii
	<b>Preface</b>	xv
	<b>Acknowledgments</b>	xix
<b>Chapter 1</b>	<b>Microprocessing</b>	1
	1-1 An Introduction to the Microcomputer	2
	1-2 Instruction Sets of the 6809 and the 6800	17
	1-3 Some Programming Techniques	56
	1-4 Conclusions	93
	Problems	95
<b>Chapter 2</b>	<b>Bus Hardware and Signals</b>	101
	2-1 Digital Hardware	102
	2-2 Control Signals and Their Sequences in a Microprocessor	113
	2-3 Interfacing to the Bus	124
	2-4 Conclusions	139
	Problems	141
<b>Chapter 3</b>	<b>Parallel Input/Output</b>	149
	3-1 Input/Output Architecture and Simple Input/Output Devices	150
	3-2 Input/Output Software	158
	3-3 Programmable Parallel Input/Output Devices	170
	3-4 Conclusions	183
	Problems	185
<b>Chapter 4</b>	<b>Interrupts and Alternatives</b>	191
	4-1 Synchronization in Input/Output Devices	192
	4-2 Synchronization Alternatives	200
	4-3 The Interrupt Mechanisms in the M6809	217
	4-4 Direct Memory Access and Context Switching	224
	4-5 Conclusions	233
	Problems	235

<b>Chapter 5</b>	<b>Analog Interfacing</b>	<b>241</b>
	5-1 Input and Output Transducers	242
	5-2 Basic Analog Processing Components	248
	5-3 Signal Conditioning Using OP AMPs and Analog Switches	262
	5-4 A Keyboard and LED Display Module	270
	5-5 Converters	281
	5-6 Data-Acquisition Systems	291
	5-7 Conclusions	297
	Problems	299
 <b>Chapter 6</b>	 <b>Counters and Timers</b>	 <b>307</b>
	6-1 The Simple M6840 Module	308
	6-2 Signal Generation	311
	6-3 Frequency Analog Measurement	329
	6-4 A Comprehensive List of M6840 Device Configurations	338
	6-5 Conclusions	344
	Problems	347
 <b>Chapter 7</b>	 <b>Communication Systems</b>	 <b>355</b>
	7-1 Communications Principles	356
	7-2 Signal Transmission	360
	7-3 UART Link Protocol	369
	7-4 Other Protocols	388
	7-5 Conclusions	395
	Problems	397
 <b>Appendix A</b>	 <b>The ASCII Character Set</b>	 <b>403</b>
 <b>Appendix B</b>	 <b>The IM6402/6403 UART</b>	 <b>405</b>
 <b>Appendix C</b>	 <b>The MC14469 Addressable UART</b>	 <b>413</b>
	 <b>Index</b>	 <b>421</b>
	 <b>About the Author</b>	 <b>427</b>

## List of Figures

1-1	Analogy to the von Neumann Computer	4
1-2	A Printed Circuit Card and a Dual-In-Line Package	15
1-3	Registers in the M6809	31
1-4	A Flow Chart	44
1-5	The 6800 Machine State	49
1-6	A Huffman Coding Tree	65
1-7	Some Formula Trees	86
1-8	The Operand Stack and the Stack Pointed To by S	89
2-1	A Block Diagram Showing the Effect of an Instruction	105
2-2	Some Common Gates	108
2-3	Logic Diagrams for a Popular Driver and Register	109
2-4	A Bus with Control Signals	116
2-5	The M6809 Microprocessor	121
2-6	Some Timing Relationships	127
2-7	Timing Relationships for the M6809	128
2-8	Timing Relationships of the National Semiconductor 2102-1	130
2-9	A Memory System	133
2-10	A Bus for Problems 7-9	142
3-1	Logic Diagrams for Basic I/O Devices	153
3-2	Logic Diagram of a Readable Output Register	156
3-3	A Traffic Light	162
3-4	Mealy Sequential Machine	166
3-5	A Linked-List Structure	169
3-6	Simplified Block Diagram of the M6821	173
3-7	Pin Connections for the M6809 and the M6821	177

3-8	An Integrated Circuit Tester	179
4-1	State Diagram for I/O Devices	193
4-2	An Address Trigger	196
4-3	M6821 Device A Interrupt/Output Mechanisms	198
4-4	Connections to a Paper-Tape Reader	205
4-5	Interrupt Request Path	206
4-6	Multiple Interrupt Sources	213
4-7	Vectored Interrupts	218
4-8	Simplified Block Diagram of the M6844	227
4-9	Interconnections for DMA	230
4-10	Connections for Context Switching	232
5-1	Position Transducers	245
5-2	Power-Output Devices	249
5-3	Triac Control Techniques	250
5-4	Operational Amplifier Circuits	253
5-5	General Analog Switches	254
5-6	A Practical Voltage Follower	259
5-7	Specific Analog Switches	260
5-8	The 555 Timer	262
5-9	Some Filters	263
5-10	Selecting Inputs for a Stereo Preamplifier	265
5-11	Computer Control of Amplification	266
5-12	Nonlinear Signal Conditioning	268
5-13	Contact Bounce and Its Prevention	271
5-14	Keyboard Structures	275
5-15	Seven-Segment Displays	279
5-16	D-to-A Converters	283
5-17	A-to-D Conversion Techniques	286
5-18	Frequency-to-Voltage Conversion	290

<b>5-19</b>	Voltage-to-Frequency Conversion	291
<b>5-20</b>	A Data-Acquisition System	292
<b>5-21</b>	A Spline Technique	294
<b>5-22</b>	Feedback Control	295
<b>5-23</b>	Some OP AMP Circuits	301
<b>6-1</b>	The M6840	309
<b>6-2</b>	The Basic Counter Device	311
<b>6-3</b>	A Gated Square-Wave Generator	314
<b>6-4</b>	A Touch-Tone Generator	317
<b>6-5</b>	A Single-Shot Pulse Generator	320
<b>6-6</b>	A Real-Time Clock	323
<b>6-7</b>	An Event Counter, or a Frequency Counter	332
<b>6-8</b>	A Period or Pulse-Width Timer	334
<b>6-9</b>	State Diagrams for Period and Pulse-Width Measurements	343
<b>7-1</b>	Peer-to-Peer Communication in Different Levels	357
<b>7-2</b>	Drivers and Receivers	362
<b>7-3</b>	Originating a Call on a Modem	366
<b>7-4</b>	A GPC Linkage for Optical Communication	368
<b>7-5</b>	Frame Format for UART Signals	370
<b>7-6</b>	Block Diagram of a UART (IM6403)	375
<b>7-7</b>	Transmitter Signals	376
<b>7-8</b>	The M6850	379
<b>7-9</b>	Output of a Test Program	382
<b>7-10</b>	The M14469	384
<b>7-11</b>	Synchronous Formats	384
<b>7-12</b>	IEEE-488 Bus Handshaking Cycle	394



# List of Tables

1-1	Classification of Computers	13
1-2	Addressing Modes for the 6809	29
1-3	6809 Move Instructions	33
1-4	6809 Arithmetic Instructions	37
1-5	6809 Logical Instructions	38
1-6	6809 Edit Instructions	39
1-7	6809 Control Instructions	47
1-8	Assembler Directives for the 6800 or 6809	58
2-1	6809 Memory-Cycle Sequences (for use with a logic analyser)	125
2-2	Address Assignments for a Microcomputer	135
2-3	Timing Characteristics of Several 2114s	145
3-1	A Traffic-Light Sequence	164
3-2	Truth Table for the 74LS00	181
5-1	Characteristics for the CA3140	256
6-1	Touch-Tone Codes	315
6-2	Signal-Generation Modes	338
6-3	Pulse- and Period-Measurement Modes	341
7-1	RS232 Pin Connections for D25P and D25S Connectors	363
7-2	M6850 Control and Status Bits	380