

Advanced PIC Microcontroller Projects in C

From USB to RTOS with the PIC18F Series

Dogan Ibrahim



AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Newnes is an imprint of Elsevier



Newnes is an imprint of Elsevier
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA
Linacre House, Jordan Hill, Oxford OX2 8DP, UK

Copyright © 2008, Elsevier Ltd. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone: (+44) 1865 843830, fax: (+44) 1865 853333, E-mail: permissions@elsevier.com. You may also complete your request online via the Elsevier homepage (<http://elsevier.com>), by selecting "Support & Contact" then "Copyright and Permission" and then "Obtaining Permissions."

- ∞ Recognizing the importance of preserving what has been written, Elsevier prints its books on acid-free paper whenever possible.

Library of Congress Cataloging-in-Publication Data

Ibrahim, Dogan.

Advanced PIC microcontroller projects in C: from USB to RTOS with the PIC18F series/Dogan Ibrahim
p. cm.

Includes bibliographical references and index.

ISBN-13: 978-0-7506-8611-2 (pbk. : alk. paper) 1. Programmable controllers. 2. C (Computer program language) I. Title.

TJ223.P76I268 2008

629.8'95—dc22

2007050550

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library.

ISBN: 978-0-7506-8611-2

For information on all Newnes publications
visit our Web site at www.books.elsevier.com

Printed in the United States of America

08 09 10 11 12 13 9 8 7 6 5 4 3 2 1

Working together to grow
libraries in developing countries

www.elsevier.com | www.bookaid.org | www.sabre.org

ELSEVIER BOOK AID Sabre Foundation
International

Contents

Preface	xiii
Acknowledgments	xv
Chapter 1: Microcomputer Systems	1
1.1 Introduction.....	1
1.2 Microcontroller Systems	1
1.2.1 RAM	5
1.2.2 ROM	5
1.2.3 PROM	5
1.2.4 EPROM.....	6
1.2.5 EEPROM	6
1.2.6 Flash EEPROM	6
1.3 Microcontroller Features	6
1.3.1 Supply Voltage	7
1.3.2 The Clock.....	7
1.3.3 Timers	7
1.3.4 Watchdog	8
1.3.5 Reset Input	8
1.3.6 Interrupts	8
1.3.7 Brown-out Detector	9
1.3.8 Analog-to-Digital Converter	9
1.3.9 Serial Input-Output	9
1.3.10 EEPROM Data Memory	10
1.3.11 LCD Drivers.....	10
1.3.12 Analog Comparator.....	10
1.3.13 Real-time Clock.....	11
1.3.14 Sleep Mode	11
1.3.15 Power-on Reset.....	11

1.3.16	Low-Power Operation	11
1.3.17	Current Sink/Source Capability	11
1.3.18	USB Interface	12
1.3.19	Motor Control Interface	12
1.3.20	CAN Interface	12
1.3.21	Ethernet Interface.....	12
1.3.22	ZigBee Interface	12
1.4	Microcontroller Architectures.....	12
1.4.1	RISC and CISC	13
1.5	Number Systems.....	13
1.5.1	Decimal Number System	14
1.5.2	Binary Number System	14
1.5.3	Octal Number System	15
1.5.4	Hexadecimal Number System	15
1.6	Converting Binary Numbers into Decimal.....	16
1.7	Converting Decimal Numbers into Binary.....	16
1.8	Converting Binary Numbers into Hexadecimal.....	18
1.9	Converting Hexadecimal Numbers into Binary	20
1.10	Converting Hexadecimal Numbers into Decimal	21
1.11	Converting Decimal Numbers into Hexadecimal	22
1.12	Converting Octal Numbers into Decimal.....	23
1.13	Converting Decimal Numbers into Octal.....	23
1.14	Converting Octal Numbers into Binary	24
1.15	Converting Binary Numbers into Octal	26
1.16	Negative Numbers	26
1.17	Adding Binary Numbers	27
1.18	Subtracting Binary Numbers	29
1.19	Multiplication of Binary Numbers.....	29
1.20	Division of Binary Numbers	31
1.21	Floating Point Numbers	31
1.22	Converting a Floating Point Number into Decimal	33
1.22.1	Normalizing Floating Point Numbers	34
1.22.2	Converting a Decimal Number into Floating Point	34
1.22.3	Multiplication and Division of Floating Point Numbers	36
1.22.4	Addition and Subtraction of Floating Point Numbers	37
1.23	BCD Numbers	38
1.24	Summary	40
1.25	Exercises	40
Chapter 2:	PIC18F Microcontroller Series	43
2.1	PIC18FXX2 Architecture.....	46
2.1.1	Program Memory Organization	50

2.1.2	Data Memory Organization	51
2.1.3	The Configuration Registers	52
2.1.4	The Power Supply	57
2.1.5	The Reset	57
2.1.6	The Clock Sources	60
2.1.7	Watchdog Timer	67
2.1.8	Parallel I/O Ports	68
2.1.9	Timers	74
2.1.10	Capture/Compare/PWM Modules (CCP)	84
2.1.11	Analog-to-Digital Converter (A/D) Module	93
2.1.12	Interrupts	101
2.2	Summary	115
2.3	Exercises	115

Chapter 3: C Programming Language..... 119

3.1	Structure of a mikroC Program	120
3.1.1	Comments	121
3.1.2	Beginning and Ending of a Program	121
3.1.3	Terminating Program Statements	121
3.1.4	White Spaces	122
3.1.5	Case Sensitivity	122
3.1.6	Variable Names	123
3.1.7	Variable Types	123
3.1.8	Constants	126
3.1.9	Escape Sequences	128
3.1.10	Static Variables	129
3.1.11	External Variables	129
3.1.12	Volatile Variables	130
3.1.13	Enumerated Variables	130
3.1.14	Arrays	131
3.1.15	Pointers	133
3.1.16	Structures	135
3.1.17	Unions	138
3.1.18	Operators in C	139
3.1.19	Modifying the Flow of Control	148
3.1.20	Mixing mikroC with Assembly Language Statements	159
3.2	PIC Microcontroller Input-Output Port Programming	160
3.3	Programming Examples	161
3.4	Summary	165
3.5	Exercises	165

Chapter 4: Functions and Libraries in mikroC.....	169
4.1 mikroC Functions.....	169
4.1.1 Function Prototypes	173
4.1.2 Passing Arrays to Functions.....	177
4.1.3 Passing Variables by Reference to Functions.....	180
4.1.4 Variable Number of Arguments	181
4.1.5 Function Reentrancy	184
4.1.6 Static Function Variables	184
4.2 mikroC Built-in Functions	184
4.3 mikroC Library Functions.....	188
4.3.1 EEPROM Library	189
4.3.2 LCD Library.....	192
4.3.3 Software UART Library	199
4.3.4 Hardware USART Library	204
4.3.5 Sound Library.....	206
4.3.6 ANSI C Library.....	208
4.3.7 Miscellaneous Library.....	212
4.4 Summary.....	218
4.5 Exercises.....	219
 Chapter 5: PIC18 Development Tools	 221
5.1 Software Development Tools.....	222
5.1.1 Text Editors.....	222
5.1.2 Assemblers and Compilers.....	222
5.1.3 Simulators	223
5.1.4 High-Level Language Simulators	224
5.1.5 Integrated Development Environments (IDEs).....	224
5.2 Hardware Development Tools.....	224
5.2.1 Development Boards.....	225
5.2.2 Device Programmers.....	239
5.2.3 In-Circuit Debuggers	242
5.2.4 In-Circuit Emulators	245
5.2.5 Breadboards.....	248
5.3 mikroC Integrated Development Environment (IDE)	251
5.3.1 mikroC IDE Screen	251
5.3.2 Creating and Compiling a New File.....	258
5.3.3 Using the Simulator.....	265
5.3.4 Using the mikroICD In-Circuit Debugger.....	272
5.3.5 Using a Development Board	277
5.4 Summary.....	285
5.5 Exercises.....	285

Chapter 6: Simple PIC18 Projects	287
6.1 Program Description Language (PDL)	288
6.1.1 START-END	288
6.1.2 Sequencing.....	288
6.1.3 IF-THEN-ELSE-ENDIF	288
6.1.4 DO-ENDDO	289
6.1.5 REPEAT-UNTIL.....	290
Project 6.1—Chasing LEDs	290
Project 6.2—LED Dice	295
Project 6.3—Two-Dice Project.....	301
Project 6.4—Two-Dice Project Using Fewer I/O Pins	303
Project 6.5—7-Segment LED Counter.....	313
Project 6.6—Two-Digit Multiplexed 7-Segment LED.....	319
Project 6.7—Two-Digit Multiplexed 7-Segment LED Counter with Timer Interrupt.....	326
Project 6.8—Voltmeter with LCD Display	334
Project 6.9—Calculator with Keypad and LCD	341
Project 6.10—Serial Communication-Based Calculator	352
Chapter 7: Advanced PIC18 Projects—SD Card Projects	371
7.1 The SD Card.....	371
7.1.1 The SPI Bus.....	373
7.1.2 Operation of the SD Card in SPI Mode	377
7.2 mikroC Language SD Card Library Functions	384
Project 7.1—Read CID Register and Display on a PC Screen	385
Project 7.2—Read/Write to SD Card Sectors.....	392
Project 7.3—Using the Card Filing System	392
Project 7.4—Temperature Logger	397
Chapter 8: Advanced PIC18 Projects—USB Bus Projects	409
8.1 Speed Identification on the Bus	413
8.2 USB States	413
8.3 USB Bus Communication.....	414
8.3.1 Packets.....	414
8.3.2 Data Flow Types.....	416
8.3.3 Enumeration.....	417
8.4 Descriptors	418
8.4.1 Device Descriptors	418
8.4.2 Configuration Descriptors.....	421
8.4.3 Interface Descriptors	423
8.4.4 HID Descriptors	425
8.4.5 Endpoint Descriptors	426

8.5 PIC18 Microcontroller USB Bus Interface	427
8.6 mikroC Language USB Bus Library Functions	429
Project 8.1—USB-Based Microcontroller Output Port	430
Project 8.2—USB-Based Microcontroller Input/Output	456
Project 8.3—USB-Based Ambient Pressure Display on the PC	464
Chapter 9: Advanced PIC18 Projects—CAN Bus Projects	475
9.1 Data Frame.....	481
9.1.1 Start of Frame (SOF)	482
9.1.2 Arbitration Field.....	482
9.1.3 Control Field.....	484
9.1.4 Data Field	484
9.1.5 CRC Field.....	484
9.1.6 ACK Field	485
9.2 Remote Frame	485
9.3 Error Frame.....	485
9.4 Overload Frame.....	485
9.5 Bit Stuffing	486
9.6 Types of Errors	486
9.7 Nominal Bit Timing	486
9.8 PIC Microcontroller CAN Interface	489
9.9 PIC18F258 Microcontroller.....	491
9.9.1 Configuration Mode	493
9.9.2 Disable Mode.....	493
9.9.3 Normal Operation Mode.....	493
9.9.4 Listen-only Mode	493
9.9.5 Loop-Back Mode	494
9.9.6 Error Recognition Mode.....	494
9.9.7 CAN Message Transmission.....	494
9.9.8 CAN Message Reception.....	494
9.9.9 Calculating the Timing Parameters	496
9.10 mikroC CAN Functions	498
9.10.1 CANSetOperationMode	499
9.10.2 CANGetOperationMode	500
9.10.3 CANInitialize	500
9.10.4 CANSetBaudRate	501
9.10.5 CANSetMask	501
9.10.6 CANSetFilter	502
9.10.7 CANRead.....	502
9.10.8 CANWrite.....	503
9.11 CAN Bus Programming	504
Project 9.1—Temperature Sensor CAN Bus Project	504

Chapter 10: Multi-Tasking and Real-Time Operating Systems.....	515
10.1 State Machines	516
10.2 The Real-Time Operating System (RTOS)	518
10.2.1 The Scheduler.....	518
10.3 RTOS Services	521
10.4 Synchronization and Messaging Tools	521
10.5 CCS PIC C Compiler RTOS.....	522
10.5.1 Preparing for RTOS	523
10.5.2 Declaring a Task	524
Project 10.1—LEDs.....	524
Project 10.2—Random Number Generator.....	528
Project 10.3—Voltmeter with RS232 Serial Output	532
Index.....	541