

EMBEDDED SYSTEMS:

INTRODUCTION TO ARM® CORTEX™-M MICROCONTROLLERS

Volume 1
Fourth Edition

Jonathan W. Valvano

Fourth *edition*

May 2013

ARM and uVision are registered trademarks of ARM Limited.

Cortex and Keil are trademarks of ARM Limited.

Stellaris is a registered trademark Texas Instruments.

Code Composer Studio is a trademark of Texas Instruments.

All other product or service names mentioned herein are the trademarks of their respective owners.

In order to reduce costs, this college textbook has been self-published. For more information about my classes, my research, and my books, see <http://users.ece.utexas.edu/~valvano/>

For corrections and comments, please contact me at: valvano@mail.utexas.edu. Please cite this book as:
J. W. Valvano, Embedded Systems: Introduction to ARM® Cortex™-M Microcontrollers, Volume 1,
<http://users.ece.utexas.edu/~valvano/>, ISBN: 978-1477508992.

Copyright © 2013 Jonathan W. Valvano

All rights reserved. No part of this work covered by the copyright herein may be reproduced, transmitted, stored, or used in any form or by any means graphic, electronic, or mechanical, including but not limited to photocopying, recording, scanning, digitizing, taping, web distribution, information networks, or information storage and retrieval, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the publisher.

ISBN-13: 978-1477508992

ISBN-10: 1477508996

Table of Contents

Preface to the Fourth Edition.....	ix
Preface.....	x
Acknowledgements	xi
1. Introduction to Computers and Electronics	1
1.1. Review of Electronics.....	2
1.2. Binary Information Implemented with MOS transistors	7
1.3. Digital Logic	8
1.4. Digital Information stored in Memory	16
1.5. Numbers	17
1.6. Character information	29
1.7. Computer Architecture.....	30
1.8. Flowcharts and Structured Programming	33
1.9. Concurrent and Parallel Programming	37
1.10. Exercises.....	40
2. Introduction to Embedded Systems.....	47
2.1. Embedded Systems.....	48
2.2. Applications Involving Embedded Systems	50
2.3. Product Development Cycle	53
2.4. Successive Refinement.....	57
2.5. Quality Design	60
2.5.1. Quantitative Performance Measurements	60
2.5.2. Qualitative Performance Measurements	61
2.5.3. Attitude	61
2.6. Debugging Theory	63
2.7. Switch and LED Interfaces.....	66
2.8. Introduction to C.....	69
2.9. Exercises.....	77
3. Introduction to the ARM® Cortex™-M Processor	79

3.1. Cortex™-M Architecture.....	80
3.1.1. Registers.....	80
3.1.2. Reset.....	82
3.1.3. Memory	82
3.1.4. Operating Modes	85
3.2. The Software Development Process.....	85
3.3. ARM® Cortex™-M Assembly Language	88
3.3.1. Syntax	88
3.3.2. Addressing Modes and Operands	90
3.3.3. Memory Access Instructions.....	95
3.3.4. Logical Operations	97
3.3.5. Shift Operations	99
3.3.6. Arithmetic Operations	101
3.3.7. Stack.....	118
3.3.8. Functions and Control Flow	120
3.3.9. Assembler Directives	122
3.3.10. First Example Project	123
3.4. Simplified Machine Language Execution	127
3.5. CISC versus RISC.....	131
3.6. Details Not Covered in this Book.....	132
3.7. Exercises.....	134
4. Introduction to Input/Output.....	137
4.1. Stellaris® LM3S/TM4C I/O pins	138
4.1.1. LM3S811 I/O pins	139
4.1.2. LM3S1968 I/O pins	141
4.1.3. Stellaris® LM4F120/TM4C123 LaunchPad I/O pins	144
4.2. Basic Concepts of Input and Output Ports	148
4.2.1. I/O Programming and the Direction Register	150
4.2.2. Switch Inputs and LED Outputs	156
4.3. Phase-Lock-Loop	166

4.4. SysTick Timer	170
4.5. OLED Display Driver and the printf Function.....	175
4.6. Debugging monitor using an LED.....	179
4.7. Performance Debugging	179
4.7.1. Instrumentation	179
4.7.2. Measurement of Dynamic Efficiency	180
4.8. Exercises.....	181
4.9. Lab Assignments.....	183
5. Modular Programming.....	185
5.1. C Keywords and Punctuation.....	186
5.2. Modular Design using Abstraction	190
5.2.1. Definition and Goals	190
5.2.2. Functions, Procedures, Methods, and Subroutines.....	192
5.2.3. Dividing a Software Task into Modules	193
5.2.4. How to Draw a Call Graph.....	196
5.2.5. How to Draw a Data Flow Graph.....	198
5.2.6. Top-down versus Bottom-up Design	199
5.3. Making Decisions	200
5.3.1. Conditional Branch Instructions	200
5.3.2. Conditional <code>if-then</code> Statements.....	202
5.3.3. <code>switch</code> Statements	207
5.3.4. While Loops	209
5.3.5. Do-while Loops.....	210
5.3.6. For Loops	210
5.4. *Assembly Macros.....	212
5.5. *Recursion.....	214
5.6. Writing Quality Software	218
5.6.1. Assembly Language Style Guidelines.....	218
5.6.2. Comments.....	223
5.6.3. Inappropriate I/O and Portability	224

5.7. How Assemblers Work.....	225
5.8. Functional debugging	226
5.8.1. Stabilization	226
5.8.2. Single Stepping	227
5.8.3. Breakpoints without Filtering.....	227
5.8.4. Instrumentation: Print Statements	228
5.9. Exercises.....	228
5.10. Lab Assignments.....	230
6. Pointers and Data Structures	231
6.1. Indexed Addressing and Pointers.....	232
6.2. Arrays	235
6.3. Strings	240
6.4. Structures	242
6.5. Finite State Machines with Linked Structures	244
6.5.1. Abstraction	244
6.5.2. Moore Finite State Machines.....	245
6.5.3. Mealy Finite State Machines	250
6.6. *Dynamically Allocated Data Structures	252
6.6.1. *Fixed Block Memory Manager	253
6.6.2. *Linked List FIFO	254
6.7. *Matrices	257
6.8. *Tables.....	262
6.9. Functional Debugging.....	265
6.9.1. Instrumentation: Dump into Array without Filtering	265
6.9.2. Instrumentation: Dump into Array with Filtering.	266
6.10. Exercises.....	267
6.11. Lab Assignments.....	270
7. Variables, Numbers, and Parameter Passing.....	271
7.1. Local versus global.....	272
7.2. Stack rules	274

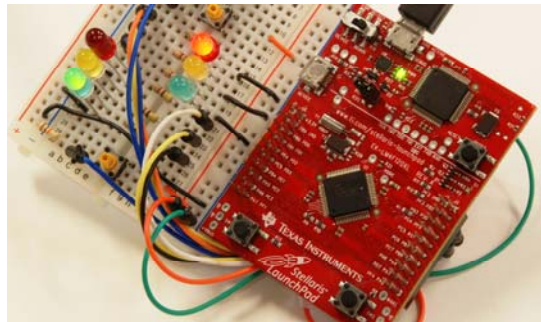
7.3. Local variables allocated on the stack	276
7.4. Stack frames	278
7.5. Parameter Passing	281
7.5.1. Parameter Passing in C.....	282
7.5.2. Parameter Passing in Assembly Language.....	284
7.5.3. C Compiler Implementation of Local and Global Variables.....	287
7.6. Fixed-point Numbers.....	289
7.7. Conversions.....	293
7.8. LCD Interface with the HD44780 Controller.....	295
7.9. *IEEE Floating-point numbers	300
7.10. Exercises.....	305
7.11. Lab Assignments.....	308
8. Serial and Parallel Port Interfacing.....	309
8.1. General Introduction to Interfacing.....	310
8.2. Universal Asynchronous Receiver Transmitter (UART).....	312
8.2.1. Asynchronous Communication	314
8.2.2. LM3S/TM4C UART Details	316
8.2.3. UART Device Driver	318
8.3. Synchronous Serial Interface, SSI	320
8.4. Scanned Keyboards	324
8.5. Binary actuators	325
8.5.1. Interface	325
8.5.2. Electromagnetic and Solid State Relays	327
8.5.3. Solenoids.....	328
8.6. *Pulse-width modulation.....	329
8.7. *Stepper motors.....	330
8.8. Exercises.....	335
8.9. Lab Assignments.....	336
9. Interrupt Programming and Real-time Systems.....	337
9.1. I/O Synchronization	337

9.2. Interrupt Concepts	342
9.3. Interthread Communication and Synchronization.....	346
9.4. NVIC on the ARM® Cortex™-M Processor	348
9.5. Edge-triggered Interrupts	353
9.6. SysTick Periodic Interrupts	360
9.7. Timer Periodic Interrupts.....	362
9.8. Hardware debugging tools	366
9.9. Profiling.....	367
9.9.1 Profiling using a software dump to study execution pattern	367
9.9.2. Profiling using an Output Port.....	368
9.9.3. *Thread Profile	369
9.10. Exercises.....	370
9.11. Lab Assignments.....	372
10. Analog I/O Interfacing	373
10.1. Approximating continuous signals in the digital domain	373
10.2. Digital to Analog Conversion	375
10.3. Music Generation	377
10.4. Analog to Digital Conversion	381
10.4.1. LM3S/TM4C ADC details.....	382
10.4.2. ADC Resolution.....	385
10.6. Real-time data acquisition	386
10.7. Exercises.....	387
10.8. Lab Assignments.....	389
11. Communication Systems.....	391
11.1. Introduction	391
11.2. Reentrant Programming and Critical Sections	394
11.3. Producer-Consumer using a FIFO Queue	399
11.3.1. Basic Principles of the FIFO Queue.....	399
11.3.2. FIFO Queue Analysis.....	405
11.3.3. FIFO Queue Implementation.....	406

11.3.4. Double Buffer	409
11.4. Serial port interface using interrupt synchronization	410
11.5. *Distributed Systems.....	414
11.6. Exercises.....	417
11.7. Lab Assignments.....	419
Appendix 1. Glossary	420
Appendix 2. Solutions to Checkpoints	433
Appendix 3. How to Convert Projects from Keil to CCS.....	441
Appendix 4. Assembly Reference	443
Index	474
Reference Material.....	482

Preface to the Fourth Edition

There are many new features added to this fourth edition. This fourth edition now focuses on the LM4F120 and TM4C123, which are new ARM® Cortex™-M4 microcontrollers from Texas Instruments. The new development platform based on the LM4F120 and TM4C123 is called Stellaris LaunchPad. Material in this book on the TM4C also applies to the LM4F because Texas Instruments rebranded the LM4F series as TM4C (same chips new name), and rebranded StellarisWare™ as TivaWare™. These new microcontrollers run at 80 MHz, include single-precision floating point, have two 12-bit ADCs, and support DMA and USB. A wonderful feature of these new boards is their low cost. As of May 2013, the boards are available on TI.com as part numbers EK-LM4F120XL or EK-TM4C123GXL for \$12.99. They are also available from \$13 to \$20 at regular electronics retailers like arrow.com, newark.com, mouser.com and digikey.com. The book can be used with either a LM3S or TM4C microcontroller. Although the book focuses on the M4, the concepts apply to the M3, and the web site associated with this book has example projects based on the LM3S811, LM3S1968, and LM3S8962. Additional material on C programming, floating point, and debugging were added in the fourth edition.



(see Section 6.5.2)