

EMBEDDED SYSTEMS:

INTRODUCTION TO THE MSP432 MICROCONTROLLER

Volume 1

First Edition

June 2015

Jonathan W. Valvano

*First edition
3rd printing
June 2015*

The true engineering experience occurs not with your eyes and ears, but rather with your fingers and elbows. In other words, engineering education does not happen by listening in class or reading a book; rather it happens by designing under the watchful eyes of a patient mentor. So, go build something today, then show it to someone you respect!

ARM and uVision are registered trademarks of ARM Limited.

Cortex and Keil are trademarks of ARM Limited.

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 the MSP432 Microcontroller, Volume 1, <http://users.ece.utexas.edu/~valvano/>, ISBN: 978-1512185676.

Copyright © 2015 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: 978-1512185676

ISBN-10: 1512185671

Table of Contents

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 Life 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	79
3. Introduction to the ARM® Cortex™-M Processor	81
3.1. Cortex™-M Architecture	82

3.1.1. Registers.....	82
3.1.2. Reset.....	84
3.1.3. Memory.....	84
3.1.4. Operating Modes.....	87
3.2. The Software Development Process.....	87
3.3. ARM Cortex-M Assembly Language.....	90
3.3.1. Syntax.....	90
3.3.2. Addressing Modes and Operands.....	92
3.3.3. Memory Access Instructions.....	97
3.3.4. Logical Operations.....	99
3.3.5. Shift Operations.....	101
3.3.6. Arithmetic Operations.....	103
3.3.7. Stack.....	120
3.3.8. Functions and Control Flow.....	122
3.3.9. Assembler Directives.....	124
3.3.10. First Example Project.....	125
3.4. Simplified Machine Language Execution.....	129
3.5. CISC versus RISC.....	133
3.6. Details Not Covered in this Book.....	134
3.7. Exercises.....	136
4. Introduction to Input/Output.....	139
4.1. Texas Instruments MSP432 I/O pins.....	140
4.2. Basic Concepts of Input and Output Ports.....	146
4.2.1. I/O Programming and the Direction Register.....	147
4.2.2. Switch Inputs and LED Outputs.....	152
4.3. Clock System.....	162
4.4. SysTick Timer.....	165
4.5. Standard I/O Driver and the printf Function.....	169
4.6. Debugging monitor using an LED.....	173
4.7. Performance Debugging.....	174

4.7.1. Instrumentation	174
4.7.2. Measurement of Dynamic Efficiency	175
4.8. Exercises	177
4.9. Lab Assignments.....	177
5. Modular Programming.....	179
5.1. C Keywords and Punctuation	180
5.2. Modular Design using Abstraction	184
5.2.1. Definition and Goals	184
5.2.2. Functions, Procedures, Methods, and Subroutines.....	186
5.2.3. Dividing a Software Task into Modules	187
5.2.4. How to Draw a Call Graph	190
5.2.5. How to Draw a Data Flow Graph.....	192
5.2.6. Top-down versus Bottom-up Design	193
5.3. Making Decisions	194
5.3.1. Conditional Branch Instructions	194
5.3.2. Conditional <code>if-then</code> Statements.....	196
5.3.3. <code>switch</code> Statements	202
5.3.4. While Loops	203
5.3.5. Do-while Loops.....	204
5.3.6. For Loops	205
5.4. *Assembly Macros.....	206
5.5. *Recursion.....	208
5.6. Writing Quality Software	213
5.6.1. Style Guidelines	213
5.6.2. Comments.....	217
5.6.3. Inappropriate I/O and Portability	219
5.7. How Assemblers Work.....	219
5.8. Functional debugging	221
5.8.1. Stabilization.....	221
5.8.2. Single Stepping	221

5.8.3. Breakpoints with Filtering.....	222
5.8.4. Instrumentation: Print Statements	222
5.8.5. Desk checking	223
5.9. Exercises.....	223
5.10. Lab Assignments.....	225
6. Pointers and Data Structures	227
6.1. Indexed Addressing and Pointers.....	228
6.2. Arrays	231
6.3. Strings	236
6.4. Structures	238
6.5. Finite State Machines with Linked Structures	240
6.5.1. Abstraction	240
6.5.2. Moore Finite State Machines.....	241
6.5.3. Mealy Finite State Machines	245
6.6. *Dynamically Allocated Data Structures	247
6.6.1. *Fixed Block Memory Manager.....	248
6.6.2. *Linked List FIFO	249
6.7. Matrices and Graphics	252
6.8. *Tables.....	257
6.9. Functional Debugging.....	260
6.9.1. Instrumentation: Dump into Array without Filtering	260
6.9.2. Instrumentation: Dump into Array with Filtering.	261
6.10. Exercises.....	262
6.11. Lab Assignments.....	264
7. Variables, Numbers, and Parameter Passing.....	265
7.1. Local versus global.....	266
7.2. Stack rules	268
7.3. Local variables allocated on the stack	270
7.4. Stack frames	272
7.5. Parameter Passing	275

7.5.1. Parameter Passing in C.....	276
7.5.2. Parameter Passing in Assembly Language.....	278
7.5.3. C Compiler Implementation of Local and Global Variables.....	281
7.6. Fixed-point Numbers.....	283
7.7. Conversions.....	287
7.8. *IEEE Floating-point numbers.....	290
7.9. Exercises.....	295
7.10. Lab Assignments.....	298
8. Serial and Parallel Port Interfacing.....	299
8.1. General Introduction to Interfacing.....	300
8.2. Universal Asynchronous Receiver Transmitter (UART).....	302
8.2.1. Asynchronous Communication.....	304
8.2.2. MSP432 UART Details.....	306
8.2.3. UART Device Driver.....	308
8.3. Synchronous Peripheral Interface, SPI.....	309
8.4. Nokia 5110 Graphics LCD Interface.....	314
8.5. Scanned Keyboards.....	316
8.6. Binary actuators.....	317
8.6.1. Interface.....	317
8.6.2. Electromagnetic and Solid State Relays.....	319
8.6.3. Solenoids.....	320
8.7. *Pulse-width modulation.....	321
8.8. *Stepper motors.....	322
8.9. Exercises.....	326
8.10. Lab Assignments.....	328
9. Interrupt Programming and Real-time Systems.....	329
9.1. I/O Synchronization.....	329
9.2. Interrupt Concepts.....	334
9.3. Interthread Communication and Synchronization.....	338
9.4. NVIC on the ARM Cortex-M Processor.....	340

9.5. Edge-triggered Interrupts	345
9.6. SysTick Periodic Interrupts	351
9.7. Periodic Interrupts using TimerA.....	354
9.8. Hardware debugging tools	356
9.9. Profiling.....	357
9.9.1 Profiling using a software dump to study execution pattern.....	357
9.9.2. Profiling using an Output Port.....	358
9.9.3. *Thread Profile	359
9.10. Exercises.....	361
9.11. Lab Assignments.....	362
10. Analog I/O Interfacing	363
10.1. Approximating continuous signals in the digital domain.....	363
10.2. Digital to Analog Conversion	365
10.3. Music Generation	367
10.4. Analog to Digital Conversion	371
10.4.1. MSP432 ADC14 details	372
10.4.2. ADC Resolution.....	375
10.5. Real-time data acquisition	376
10.6. Exercises.....	377
10.7. Lab Assignments.....	378
11. Communication Systems	379
11.1. Introduction	379
11.2. Reentrant Programming and Critical Sections	382
11.3. Producer-Consumer using a FIFO Queue	387
11.3.1. Basic Principles of the FIFO Queue.....	387
11.3.2. FIFO Queue Analysis	393
11.3.3. FIFO Queue Implementation.....	394
11.3.4. Double Buffer	397
11.4. Serial port interface using interrupt synchronization	398
11.5. *Distributed Systems.	400

11.6. Exercises	403
11.7. Lab Assignments	405
11.8. Best Practices	405
Appendix 1. Glossary	406
Appendix 2. Solutions to Checkpoints	419
Appendix 3. How to Convert Projects from Keil to CCS	427
Appendix 4. Assembly Reference	429
Index	460

