

CONTENTS

ABOUT THE AUTHORS

ii

PREFACE

xi

CHAPTER

1

ARM EMBEDDED SYSTEMS

3

1.1 The RISC Design Philosophy

4

1.2 The ARM Design Philosophy

5

1.3 Embedded System Hardware

6

1.4 Embedded System Software

12

1.5 Summary

15

CHAPTER

2

ARM PROCESSOR FUNDAMENTALS

19

2.1 Registers

21

2.2 Current Program Status Register

22

2.3 Pipeline

29

2.4 Exceptions, Interrupts, and the Vector Table

33

2.5 Core Extensions

34

2.6 Architecture Revisions

37

2.7 ARM Processor Families

38

2.8 Summary

43

CHAPTER

3

INTRODUCTION TO THE ARM INSTRUCTION SET

47

3.1 Data Processing Instructions

50

3.2 Branch Instructions

58

3.3 Load-Store Instructions

60

3.4 Software Interrupt Instruction

73

3.5 Program Status Register Instructions

75

3.6 Loading Constants

78

3.7 ARMv5E Extensions

79

3.8 Conditional Execution

82

3.9 Summary

84

INTRODUCTION TO THE THUMB INSTRUCTION SET

87

4.1	Thumb Register Usage	89
4.2	ARM-Thumb Interworking	90
4.3	Other Branch Instructions	92
4.4	Data Processing Instructions	93
4.5	Single-Register Load-Store Instructions	96
4.6	Multiple-Register Load-Store Instructions	97
4.7	Stack Instructions	98
4.8	Software Interrupt Instruction	99
4.9	Summary	100

EFFICIENT C PROGRAMMING

103

5.1	Overview of C Compilers and Optimization	104
5.2	Basic C Data Types	105
5.3	C Looping Structures	113
5.4	Register Allocation	120
5.5	Function Calls	122
5.6	Pointer Aliasing	127
5.7	Structure Arrangement	130
5.8	Bit-fields	133
5.9	Unaligned Data and Endianness	136
5.10	Division	140
5.11	Floating Point	149
5.12	Inline Functions and Inline Assembly	149
5.13	Portability Issues	153
5.14	Summary	155

WRITING AND OPTIMIZING ARM ASSEMBLY CODE

157

6.1	Writing Assembly Code	158
6.2	Profiling and Cycle Counting	163
6.3	Instruction Scheduling	163
6.4	Register Allocation	171
6.5	Conditional Execution	180
6.6	Looping Constructs	183
6.7	Bit Manipulation	191
6.8	Efficient Switches	197

6.9	Handling Unaligned Data	201
6.10	Summary	204

CHAPTER

7

OPTIMIZED PRIMITIVES	207
-----------------------------	-----

7.1	Double-Precision Integer Multiplication	208
7.2	Integer Normalization and Count Leading Zeros	212
7.3	Division	216
7.4	Square Roots	238
7.5	Transcendental Functions: log, exp, sin, cos	241
7.6	Endian Reversal and Bit Operations	248
7.7	Saturated and Rounded Arithmetic	253
7.8	Random Number Generation	255
7.9	Summary	256

CHAPTER

8

DIGITAL SIGNAL PROCESSING	259
----------------------------------	-----

8.1	Representing a Digital Signal	260
8.2	Introduction to DSP on the ARM	269
8.3	FIR filters	280
8.4	IIR Filters	294
8.5	The Discrete Fourier Transform	303
8.6	Summary	314

CHAPTER

9

EXCEPTION AND INTERRUPT HANDLING	317
---	-----

9.1	Exception Handling	318
9.2	Interrupts	324
9.3	Interrupt Handling Schemes	333
9.4	Summary	364

CHAPTER

10

FIRMWARE	367
-----------------	-----

10.1	Firmware and Bootloader	367
10.2	Example: Sandstone	372
10.3	Summary	379

11

EMBEDDED OPERATING SYSTEMS

381

11.1	Fundamental Components	381
11.2	Example: Simple Little Operating System	383
11.3	Summary	400

12

CACHES

403

12.1	The Memory Hierarchy and Cache Memory	404
12.2	Cache Architecture	408
12.3	Cache Policy	418
12.4	Coprocessor 15 and Caches	423
12.5	Flushing and Cleaning Cache Memory	423
12.6	Cache Lockdown	443
12.7	Caches and Software Performance	456
12.8	Summary	457

13

MEMORY PROTECTION UNITS

461

13.1	Protected Regions	463
13.2	Initializing the MPU, Caches, and Write Buffer	465
13.3	Demonstration of an MPU system	478
13.4	Summary	487

14

MEMORY MANAGEMENT UNITS

491

14.1	Moving from an MPU to an MMU	492
14.2	How Virtual Memory Works	493
14.3	Details of the ARM MMU	501
14.4	Page Tables	501
14.5	The Translation Lookaside Buffer	506
14.6	Domains and Memory Access Permission	510
14.7	The Caches and Write Buffer	512
14.8	Coprocessor 15 and MMU Configuration	513
14.9	The Fast Context Switch Extension	515
14.10	Demonstration: A Small Virtual Memory System	520
14.11	The Demonstration as mmuSLOS	545
14.12	Summary	545

THE FUTURE OF THE ARCHITECTURE**BY JOHN RAYFIELD**

549

15.1	Advanced DSP and SIMD Support in ARMv6	550
15.2	System and Multiprocessor Support Additions to ARMv6	560
15.3	ARMv6 Implementations	563
15.4	Future Technologies beyond ARMv6	563
15.5	Summary	566

APPENDIX

A**ARM AND THUMB ASSEMBLER INSTRUCTIONS**

569

A.1	Using This Appendix	569
A.2	Syntax	570
A.3	Alphabetical List of ARM and Thumb Instructions	573
A.4	ARM Assembler Quick Reference	620
A.5	GNU Assembler Quick Reference	631

APPENDIX

B**ARM AND THUMB INSTRUCTION ENCODINGS**

637

B.1	ARM Instruction Set Encodings	637
B.2	Thumb Instruction Set Encodings	638
B.3	Program Status Registers	645

APPENDIX

C**PROCESSORS AND ARCHITECTURE**

647

C.1	ARM Naming Convention	647
C.2	Core and Architectures	647

APPENDIX

D**INSTRUCTION CYCLE TIMINGS**

651

D.1	Using the Instruction Cycle Timing Tables	651
D.2	ARM7TDMI Instruction Cycle Timings	653
D.3	ARM9TDMI Instruction Cycle Timings	654
D.4	StrongARM1 Instruction Cycle Timings	655
D.5	ARM9E Instruction Cycle Timings	656
D.6	ARM10E Instruction Cycle Timings	658
D.7	Intel XScale Instruction Cycle Timings	659
D.8	ARM11 Cycle Timings	661

APPENDIX

E

SUGGESTED READING

E.1	ARM References	667
E.2	Algorithm References	667
E.3	Memory Management and Cache Architecture (Hardware Overview and Reference)	667
E.4	Operating System References	668

INDEX

669