

EMBEDDED SYSTEM DESIGN

A Unified Hardware/Software
Introduction



Frank Vahid / Tony Givargis

Contents

Preface	vii
Purpose	vii
Coverage	vii
How to Use This Book	viii
Laboratory	ix
Additional Materials	x
Acknowledgments	x
About the Authors	xi
CHAPTER 1: Introduction	1
1.1 Embedded Systems Overview	1
1.2 Design Challenge — Optimizing Design Metrics	4
Common Design Metrics	4
The Time-to-Market Design Metric	6
The NRE and Unit Cost Design Metrics	7
The Performance Design Metric	8
1.3 Processor Technology	9
General-Purpose Processors — Software	9
Single-Purpose Processors — Hardware	10
Application-Specific Processors	12
1.4 IC Technology	13
Full-Custom/VLSI	13
Semicustom ASIC (Gate Array and Standard Cell)	13
PLD	14
Trends	14
1.5 Design Technology	16
Compilation/Synthesis	17
Libraries/IP	18

Test/Verification	18
More Productivity Improvers	18
Trends	19
1.6 Trade-offs	19
Design Productivity Gap	22
1.7 Summary and Book Outline	24
1.8 References and Further Reading	25
1.9 Exercises	25
CHAPTER 2: Custom Single-Purpose Processors: Hardware	29
2.1 Introduction	29
2.2 Combinational Logic	30
Transistors and Logic Gates	30
Basic Combinational Logic Design	32
RT-Level Combinational Components	33
2.3 Sequential Logic	34
Flip-Flops	34
RT-Level Sequential Components	35
Sequential Logic Design	36
2.4 Custom Single-Purpose Processor Design	38
2.5 RT-Level Custom Single-Purpose Processor Design	44
2.6 Optimizing Custom Single-Purpose Processors	47
Optimizing the Original Program	47
Optimizing the FSMD	48
Optimizing the Datapath	50
Optimizing the FSM	50
2.7 Summary	51
2.8 References and Further Reading	51
2.9 Exercises	52
CHAPTER 3: General-Purpose Processors: Software	55
3.1 Introduction	55
3.2 Basic Architecture	56
Datapath	56
Control Unit	57
Memory	58
3.3 Operation	59
Instruction Execution	59
Pipelining	60

Superscalar and VLIW Architectures	61
3.4 Programmer's View	61
Instruction Set	62
Program and Data Memory Space	64
Registers	64
I/O	65
Interrupts	65
Example: Assembly-Language Programming of Device Drivers	66
Operating System	67
3.5 Development Environment	69
Design Flow and Tools	69
Example: Instruction-Set Simulator for a Simple Processor	71
Testing and Debugging	71
3.6 Application-Specific Instruction-Set Processors (ASIPs)	74
Microcontrollers	74
Digital Signal Processors (DSP)	75
Less-General ASIP Environments	75
3.7 Selecting a Microprocessor	75
3.8 General-Purpose Processor Design	77
3.9 Summary	80
3.10 References and Further Reading	80
3.11 Exercises	81
CHAPTER 4: Standard Single-Purpose Processors: Peripherals	83
4.1 Introduction	83
4.2 Timers, Counters, and Watchdog Timers	84
Timers and Counters	84
Example: Reaction Timer	87
Watchdog Timers	88
Example: ATM Timeout Using a Watchdog Timer	89
4.3 UART	90
4.4 Pulse Width Modulators	92
Overview	92
Example: Controlling a DC Motor Using a PWM	94
4.5 LCD Controllers	95
Overview	95
Example: LCD Initialization	97
4.6 Keypad Controllers	97

4.7	Stepper Motor Controllers	98
	Overview	98
	Example: Using a Stepper Motor Driver	99
	Example: Controlling a Stepper Motor Directly	101
4.8	Analog-to-Digital Converters	102
	Example: Successive Approximation	103
4.9	Real-Time Clocks	105
4.10	Summary	106
4.11	References and Further Reading	106
4.12	Exercises	107
CHAPTER 5: Memory		109
5.1	Introduction	109
5.2	Memory Write Ability and Storage Permanence	111
	Write Ability	111
	Storage Permanence	112
	Trade-offs	112
5.3	Common Memory Types	112
	Introduction to “Read-Only” Memory — ROM	112
	Mask-Programmed ROM	114
	OTP ROM — One-Time Programmable ROM	114
	EPROM — Erasable Programmable ROM	115
	EEPROM — Electrically Erasable Programmable ROM	116
	Flash Memory	117
	Introduction to Read-Write Memory — RAM	118
	SRAM — Static RAM	119
	DRAM — Dynamic RAM	120
	PSRAM — Pseudo-Static RAM	120
	NVRAM — Nonvolatile RAM	120
	Example: HM6264 and 27C256 RAM/ROM Devices	120
	Example: TC55V2325FF-100 Memory Device	122
5.4	Composing Memory	123
5.5	Memory Hierarchy and Cache	125
	Cache Mapping Techniques	126
	Cache-Replacement Policy	128
	Cache Write Techniques	128
	Cache Impact on System Performance	128
5.6	Advanced RAM	130

The Basic DRAM	130
Fast Page Mode DRAM (FPM DRAM)	131
Extended Data Out DRAM (EDO DRAM)	132
Synchronous (S) and Enhanced Synchronous (ES) DRAM	132
Rambus DRAM (RDRAM)	133
DRAM Integration Problem	133
Memory Management Unit (MMU)	134
5.7 Summary	134
5.8 References and Further Reading	135
5.9 Exercises	135
CHAPTER 6: Interfacing	137
6.1 Introduction	137
6.2 Communication Basics	138
Basic Terminology	138
Basic Protocol Concepts	140
Example: The ISA Bus Protocol — Memory Access	143
6.3 Microprocessor Interfacing: I/O Addressing	144
Port and Bus-Based I/O	144
Memory-Mapped I/O and Standard I/O	145
Example: The ISA Bus Protocol — Standard I/O	147
Example: A Basic Memory Protocol	147
Example: A Complex Memory Protocol	148
6.4 Microprocessor Interfacing: Interrupts	148
6.5 Microprocessor Interfacing: Direct Memory Access	153
Example: DMA I/O and the ISA Bus Protocol	158
6.6 Arbitration	159
Priority Arbiter	160
Daisy-Chain Arbitration	160
Network-Oriented Arbitration Methods	162
Example: Vectored Interrupt Using an Interrupt table	162
6.7 Multilevel Bus Architectures	164
6.8 Advanced Communication Principles	166
Parallel Communication	166
Serial Communication	166
Wireless Communication	167
Layering	168
Error Detection and Correction	168

6.9	Serial Protocols	169
	I ² C	169
	CAN	171
	FireWire	172
	USB	172
6.10	Parallel Protocols	173
	PCI Bus	173
	ARM Bus	173
6.11	Wireless Protocols	174
	IrDA	174
	Bluetooth	174
	IEEE 802.11	174
6.12	Summary	176
6.13	References and Further Reading	176
6.14	Exercises	176
CHAPTER 7:	Digital Camera Example	179
7.1	Introduction	179
7.2	Introduction to a Simple Digital Camera	179
	User's Perspective	180
	Designer's Perspective	180
7.3	Requirements Specification	185
	Nonfunctional Requirements	186
	Informal Functional Specification	187
	Refined Functional Specification	187
7.4	Design	194
	Implementation 1: Microcontroller Alone	195
	Implementation 2: Microcontroller and CCDPP	195
	Implementation 3: Microcontroller and CCDPP/Fixed-Point DCT	200
	Implementation 4: Microcontroller and CCDPP/DCT	203
7.5	Summary	205
7.6	References and Further Reading	205
7.7	Exercises	205
CHAPTER 8:	State Machine and Concurrent Process Models	207
8.1	Introduction	207
8.2	Models vs. Languages, Text vs. Graphics	209
	Models vs. Languages	209
	Textual Languages vs. Graphical Languages	210

8.3	An Introductory Example	210
8.4	A Basic State Machine Model: Finite-State Machines	211
8.5	Finite-State Machine with Datapath Model: FSMD	213
8.6	Using State Machines	213
	Describing a System as a State Machine	214
	Comparing State Machine and Sequential Program Models	214
	Capturing State Machines in Sequential Programming Language	215
8.7	HCFSM and the Statecharts Language	217
8.8	Program-State Machine Model (PSM)	220
8.9	The Role of an Appropriate Model and Language	221
8.10	Concurrent Process Model	222
8.11	Concurrent Processes	225
	Process Create and Terminate	226
	Process Suspend and Resume	227
	Process Join	227
8.12	Communication among Processes	227
	Shared Memory	227
	Message Passing	231
8.13	Synchronization among Processes	232
	Condition Variables	233
	Monitors	235
8.14	Implementation	236
	Creating and Terminating Processes	236
	Suspending and Resuming Processes	238
	Joining a Process	239
	Scheduling Processes	239
8.15	Dataflow Model	241
8.16	Real-Time Systems	242
	Windows CE	242
	QNX	243
8.17	Summary	243
8.18	References and Further Reading	244
8.19	Exercises	244
CHAPTER 9:	Control Systems	245
9.1	Introduction	245
9.2	Open-Loop and Closed-Loop Control Systems	246
	Overview	246

A First Example: An Open-Loop Automobile Cruise Controller	248
A Second Example: A Closed-Loop Automobile Cruise Controller	251
9.3 General Control Systems and PID Controllers	256
Control Objectives	256
Modeling Real Physical Systems	257
Controller Design	258
9.4 Software Coding of a PID Controller	261
9.5 PID Tuning	262
9.6 Practical Issues Related to Computer-Based Control	263
Quantization and Overflow Effects	263
Aliasing	264
Computation Delay	265
9.7 Benefits of Computer-Based Control Implementations	265
Repeatability, Reproducibility, and Stability	266
Programmability	266
9.8 Summary	266
9.9 References and Further Reading	266
9.10 Exercises	267
CHAPTER 10: IC Technology	269
10.1 Introduction	269
10.2 Full-Custom (VLSI) IC Technology	273
10.3 Semi-Custom (ASIC) IC Technology	276
Gate Array Semi-Custom IC Technology	276
Standard Cell Semi-Custom IC Technology	276
10.4 Programmable Logic Device (PLD) IC Technology	277
10.5 Summary	280
10.6 References and Further Reading	280
10.7 Exercises	280
CHAPTER 11: Design Technology	281
11.1 Introduction	281
11.2 Automation: Synthesis	282
“Going up”: The Parallel Evolution of Compilation and Synthesis	282
Synthesis Levels	285
Logic Synthesis	287
Register–Transfer Synthesis	293
Behavioral Synthesis	293
System Synthesis and Hardware/Software Codesign	294

Temporal and Spatial Thinking	295
11.3 Verification: Hardware/Software Co-Simulation	296
Formal Verification and Simulation	296
Simulation Speed	298
Hardware-Software Co-Simulation	299
Emulators	301
11.4 Reuse: Intellectual Property Cores	301
Hard, soft and firm cores	302
New Challenges Posed by Cores to Processor Providers	302
New Challenges Posed by Cores to Processor Users	303
11.5 Design Process Models	304
11.6 Summary	306
11.7 Book Summary	307
11.8 References and Further Reading	307
11.9 Exercises	308
APPENDIX A: Online Resources	311
A.1 Introduction	311
A.2 Summary of the ESD Web Page	312
A.3 Lab Resources	312
Chapter 2	312
Chapter 3	314
Chapter 4	314
Chapter 5	315
Chapter 6	315
Chapter 7	315
A.4 About the Book Cover	315
Outdoors	315
Indoors	316
Index	319