

Contents

Foreword	xiii
Preface	xv
Acknowledgments	xvii
Introduction	xviii

Part I — Design

Chapter 1	Design Requirements	3
	Overview	3
	The Development Teams.	7
	Hardware Group (Embedded Device Design/Implementation) . . .	7
	Device Interface Group (Device Interface and Device Drivers) . .	8
	User Application Group (User Applications for UNIX and Windows)	
		8
	Concerns and Issues	8
	Platform-Independent Software Requirements	9
	Embedded Device Requirements	11
	Device Interface Requirements	12
	Desktop Software Requirements.	13
	Operating Systems Support	14
	Requirements Summary	15
	Data and Control Flow Management.	16
	Project Management and Data Modeling Tools	19
	Data Flow Diagrams.	19
	Embedded Device Functional Layout.	19
	RS-232 Interface	20
	USB Interface	21
	Ethernet Interface	22
	Peer-to-Peer Device Networking	23
	Many-to-Many Peer Networking.	24
	Internet Accessibility.	24
	X-10 Monitoring and Control.	24
	Unlimited Expansion Capabilities	24
	Chapter Summary.	26
Chapter 2	Embedded System Design	27
	Overview.	27
	Special Skills.	28
	The Functional Design	30
	The Embedded View of the World	32



Serial Communications	34
Asynchronous Serial Communications	35
Byte Ordering	35
Baud Rate	36
Character Length	36
Start and Stop Bits	36
Flow Control	36
Parity	37
Synchronous Serial Communications	37
Comparing Asynchronous and Synchronous Communications	38
Synchronous Transfer Types	39
Parallel Data Communications	39
The Communications World of Windows and UNIX	41
The Protocol, a Bridge between Different Worlds	42
Embedded System Protocol Functions	43
Communications Protocol History	45
Protocol Definition Considerations	46
Designing the Device Protocol	49
Protocol Message and Variable Size Considerations	51
Embedded System Protocol Responses	52
Embedded System Summary	55
Protocol Summary	55
Chapter Summary	56

Chapter 3 **Designing Portable Device Interfaces**

for Windows and UNIX.	57
Overview	57
The Basic Driver Design	59
Language Portability	61
Existing Operating System Support	62
The RS-232 Interface	62
The Ethernet Interface	63
The USB Interface	65
The Windows Driver Architecture	67
The Windows Boot Process	67
When the Boot Process Fails (The Recovery Console)	68
The UNIX System Architecture	69
Device Driver Types	71
Character Devices	71
Block Devices	71
Network Devices	72
USB Devices	73
Portable Device Driver Requirements	74
Hardware Function Calls	77
Generic Device Functions	77
Device-Specific Functions	78



RS-232-Specific Functions	78
Ethernet-Specific Functions	79
USB-Specific Functions.	80
Putting It All Together	83
Device State Category.	84
Device Setup Category	84
Device I/O Control Category	84
Data Category.	84
Function Parameters	85
Designing for True Portability	86
Chapter Summary	87
Chapter 4 Designing Cross-Platform User Applications	89
Overview.	89
The Windows Application Programming Interface (API).	90
Windows Application Programming Layers	91
The Dynamic-Link Library (DLL)	91
The Visual Basic Control (VBX)	91
The ActiveX Controls	92
The Component Object Model (COM)	92
Network COM (COM+).	92
The .NET Specification	92
The UNIX Application Programming Model.	94
X Windows in Depth — A True Client/Server GUI.	96
Alternatives to the X Windows Approach	99
Alternatives to Cross-Platform Development	101
Emulation vs. Native Execution	103
Programming Languages	103
Employee Skill Sets	104
Visual Application Development or Hand Coding	104
Visual Development vs. Development Framework.	105
Free vs. Commercial Products.	106
GNU vs. GPL Licensing	106
Programming Language Selection.	107
Programmer Editors	109
A Final Word about Free Software	110
The Cost of Free	110
Shareware — The Middle Ground.	111
Chapter Summary	111
Chapter 5 Developing the Desktop User Application	113
Overview	113
Functional Requirements	113
Multi-Language Support.	114
Chinese, Japanese, and Korean Language Support.	114
Graphic or Text Mode Based Application.	115



Text Mode Interfaces	115
The Modern Graphic Desktop	116
The Microsoft Windows Operating Systems	116
Memory Usage	117
Developer Libraries and Tools	117
Qt Library	117
wxWindows	119
Zinc Interface Library	121
GTK+	124
Comparing GTK+, Qt, wxWindows, and Zinc	125
GTK+, Qt, wxWindows, and Zinc Interface Library Summary	128
The .NET Framework	129
Cross-Platform Compilers	131
Borland Products	131
Delphi	132
C++Builder	134
Intel C++	136
The GNU Compiler Collection (GCC)	136
Commercial Compilers in Comparison	138
Cross-Platform Development Environments	139
C++BuilderX	139
Emacs	141
GDB	144
Choosing the Right Tools for the Design	147
And the Winners Are...	150
Chapter Summary	151
Chapter 6 Final Design	153
Overview	153
Master Design Document	153
Purpose	153
Embedded System Final Design	154
Design Overview	154
External Device Interfaces	155
Temperature Measurement	156
Output Display	157
User Input	157
Command Mode	158
Menu Navigation	158
Time – Set Time of Day	160
Date – Set Current Date	161
Host Interface Type	162
High Temperature Alarm Trigger	163
Low Temperature Alarm Trigger	164
Exit Menu	165
Temperature Sensor Reset	165



System Reset	165
Host Interfaces	165
RS-232	166
USB	166
Ethernet	166
Host Side Design	167
Device Layer	167
Application Layer	168
Differences between Windows and UNIX	168
The Communication Protocol	169
Embedded System Flowcharts	170
Host System Flowcharts	180
Device Interface Layer	180
User Application Layer	181
Chapter Summary	182
Part I Summary	182

Part II — Implementation

Chapter 7	Hardware Development	185
	Overview	185
	Types of Prototype Circuit Boards	186
	Solderless Breadboards	187
	Soldered Breadboards	188
	Wire Wrap Boards	188
	Required Tools	189
	Soldering Techniques	191
	Building the Prototype Circuit Boards	192
	The RS-232 Interface Board	193
	RS-232 Signal Voltage Levels	196
	The MAX232 Serial Interface Chip	196
	The DSR/DTR Handshake Described	198
	The RTS/CTS Handshake Described	198
	The XON/XOFF Handshake Described	198
	Hardware and Software Handshake Considerations	199
	Building the RS-232 Interface	203
	The RS-232 Schematic	206
	Testing the RS-232 Interface	211
	Setting Up the BASIC Stamp Development Board	211
	Troubleshooting	213
	Finishing Touches	214
	Running uComm	215
	The Time and Temperature Board	217
	The Dallas 1-Wire Protocol	217
	The DS1820 and DS1822 Temperature Chips	218
	The DS2404 Time/Date/Static RAM Chip	219



The DS2890 Potentiometer	219
The DS2406 Digital Switch	219
The Dallas 3-Wire Protocol	220
A Light-Duty 1-Wire Network	221
A Simple 1-Wire Test Program	224
Adding the DS2404	227
Test Software for the DS2404.	229
From Solderless Breadboard to Circuit Board	233
The Keyboard, Alarm, and Power Supply Board	234
Testing the LCD and I/O Board	243
Design Considerations	245
How to Make Your Own Printed Circuit Boards	246
Circuit Board Artwork	249
Chapter Summary	252

Chapter 8 The BASIC Stamp 2p Prototype	253
Overview	253
Our Development Road Map	254
Using the BASIC Stamp 2p24.	257
Implementation Strategy	258
Developing Our BASIC Stamp Board.	259
Power Sources	260
Connecting the Reset and Host Communications	261
Remaining I/O Pin Requirements	263
The LCD Display.	264
The I ² C Communications Interface	264
BASIC Stamp I ² C Communications Support	264
LCD Command Software.	266
Creating Formatted LCD Output	268
Alarm, LED, and Fan Control Outputs	269
Wiring the Alarm LED	271
Wiring the Alarm Buzzer.	272
Keyboard Wiring	272
Keyboard Codes.	272
Menu Functionality	273
Using the DS2404 Time and Memory Chip	275
The DS2404 Timers	275
The DS2404 Alarm Time Registers	276
DS2404 Interrupt Registers	277
Timers and the BASIC Stamp 2p	277
Static RAM vs. EEPROM	279
Communications Selection Options.	280
Ethernet for the Embedded Developer	282
The NetBurner SB72.	286
Building the Ethernet Circuit	286
Setting up the SB72	291



	Before the Installation.	291
	After the Installation	292
	Testing the SB72 Setup	292
	Debugging an SB72 Program	301
	Testing Ethernet Communications	303
	Connecting the SB72 to the BASIC Stamp	309
	The USB Interface	319
	Implementing the USB Circuits	320
	The Parallax USB2SER Device	327
	The Maxim/Dallas Hardware Interfaces	331
	The 1-Wire Interface Hardware	331
	The 3-Wire Hardware Interface	334
	Adding RS-232 Communications	341
	Finishing Touches for Our BS2p Prototype	343
	Chapter Summary	344
Chapter 9	The PIC Prototype	345
	Similarities between the BASIC Stamp and the PIC	346
	Differences between the BASIC Stamp and the PIC	347
	Types of PIC Microcontrollers and Development Languages.	348
	The PIC Development Cycle	349
	Development Strategy.	351
	What Is HIDmaker?	353
	The dsPIC30F2010 Processor	356
	The Ethernet Interface	358
	The dsPIC Code Development Environments	359
	The MPLAB IDE	359
	The RS-232 Interface	360
	The dsPIC Peripheral Libraries.	361
	The SPI Interface	365
	The I ² C Interface.	368
	The Dallas 1-Wire Interface	373
	Implementing USB and Ethernet Communications	374
	The PIC 16C745 USB Chip	375
	The Microchip USB Developer Kits	377
	The Microchip USB Engine	378
	Defining a USB Device to the Host System	380
	Finishing the Communications Tasks.	385
	The NetBurner SB72 Board.	386
	The TTL Serial Routine	390
	The Finished Circuit.	392
	Chapter Summary	395
Chapter 10	The PSoC Prototype	397
	What Is the PSoC?	397
	Comparing the PIC and the PSoC.	398



The PSoC Design Process	400
The PSoC Designer	400
PSoC Assembly Language	402
The PSoC C Compiler	402
Available Hardware Development Tools	403
The EZ-USB Chip	406
Development Strategy	407
Using the PSoC Designer	408
The PSoC I ² C Function Library	419
Adding User Modules to the PSoC Designer	420
The PSoC 1-Wire Interface	420
The PSoC SPI User Module	421
The PSoC RS-232 Interface	423
The Cypress EZ-USB Chip	429
The EZ-USB UARTs	438
The Ethernet Interface	439
The NetBurner SB72 Board	439
PSoC Schematics	443
Chapter Summary	444
Hardware Summary	445

Chapter 11 Cross-Platform Application Development 447

Ethernet, TCP/IP, and UDP	448
RS-232 Support	449
USB Support	449
The User Application Layer	450
Commercial Compilers	450
Delphi/Kylix	450
C++Builder	450
C++BuilderX	451
GNU Compilers	451
Application Frameworks	451
Common Application Layers	452
The Device Interface Layer	452
The Ethernet Software Interface	453
Welcome to the World of Sockets	453
The Standard Function Library	454
Alternative Cross-Platform Choices	455
Visual vs. Code Solutions	456
Socket Implementation	458
Platform-Independent USB Development	459
Comparing Delphi/Kylix to C++BuilderX	460
The End User Application Layer	461
Delphi/C++Builder/Kylix	461
C++BuilderX	461
Mono and the .NET Framework	462



wxWidgets and DialogBlocks	462
It's Time to Develop the User Application	462
The Delphi Development Effort	463
The wxWidgets Development Effort	465
Distributing the End Results	467
The Sky Is the Limit.	467
Index	468
