Expert .NET Micro Framework

Jens Kühner

Apress®

Expert .NET Micro Framework

ISBN-13: 978-1-59059-973-0

ISBN-10: 1-59059-973-X

ISBN-13 (electronic): 978-1-4302-0608-8

ISBN-10 (electronic): 1-4302-0608-X

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Lead Editor: Dominic Shakeshaft
Technical Reviewer: Fabio Claudio Ferracchiati
Editorial Board: Clay Andres, Steve Anglin, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Matthew Moodie, Joseph Ottinger, Jeffrey Pepper, Frank Pohlmann, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Project Manager: Richard Dal Porto
Copy Editor: Heather Lang
Associate Production Director: Kari Brooks-Copony
Production Editor: Liz Berry
Compositor: Susan Glinert Stevens
Proofreader: Lisa Hamilton
Indexer: Becky Hornyak
Artist: April Milne
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit http://www.springeronline.com.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales—eBook Licensing web page at http://www.apress.com/info/bulksales.

The source code for this book is available to readers at http://www.apress.com. You may need to answer questions pertaining to this book in order to successfully download the code.

To my wonderful son Marek and beautiful wife Iryna: you have added a wonderful new dimension to my life that I didn't even know I was missing.

Contents at a Glance

About the Author xv
About the Technical Reviewer xvii
Acknowledgments xix
Introduction xxi

CHAPTER 1 Introducing the .NET Micro Framework 1
CHAPTER 2 Devices 15
CHAPTER 3 Getting Started 35
CHAPTER 4 Introducing the .NET Micro Framework Base Class Library 47
CHAPTER 5 Accessing Hardware 73
CHAPTER 6 Networking 117
CHAPTER 7 Wireless Communication 173
CHAPTER 8 Cryptography 179
CHAPTER 9 Advanced .NET Micro Framework Programming 189
CHAPTER 10 Globalization and Localization 213
CHAPTER 11 Graphics and Graphical User Interfaces 235
CHAPTER 12 Hardware Emulation 317
CHAPTER 13 Emulator Components 361

INDEX 409

Contents

About the Author xv
About the Technical Reviewer xvii
Acknowledgments xix
Introduction xxi

CHAPTER 1 Introducing the .NET Micro Framework 1

What Is the .NET Micro Framework? 1
.NET Micro Framework History 1
Motivation 3
Embedded Development in the Past. 3
A Different Approach 4
Benefits of Managed Code. 4
Where the .NET Micro Framework Fits. 5
What the Framework Is Not. 6
Licensing 6
Benefits of the .NET Micro Framework 6
The .NET Micro Framework and Windows SideShow. 7
Technical Overview 9
Introducing the Bootable Runtime Environment 9
Architecture. 10
Compiling for the .NET Micro Framework 11
Target platforms 12
Platform Porting 12
Links for Further Information 13
Summary 13

CHAPTER 2 Devices 15

Freescale i.MXS Development Kit 15
Device Solutions Meridian and Tahoe 17
Meridian CPU. 17
Tahoe Development Kit 19
Digi International's Embedded Development Products 19
Digi Connect ME 19
Digi ConnectCore 9P 9215 22

SJJ Embedded Micro Solutions Embedded Development Kit 23
Crossbow Imote2.NET Edition 24
Module 25
Multisensor Board 26
Development Kit 26
Adeneo Ports 27
PHYTEC phyCORE-LPC3180 27
Atmel AT91SAM9261 28
GHI Electronics Embedded Master 30
The Embedded Master Module 30
Development Kit 32
emtrion HiCO.ARM9 33
The HiCO.ARM9-Core Module 33
Development Kit 33
Summary 34

CHAPTER 3 Getting Started 35

Getting Your System Ready 35
System Requirements 35
Installing the .NET Micro Framework SDK 36
Using Visual Studio and the .NET Micro Framework 36
Creating a New Project 36
Your First .NET Micro Framework Application 37
The Visual Studio Workspace 38
Setting Project Properties 39
Building, Deploying, Executing, and Debugging 40
The .NET Micro Framework SDK 40
File Structure 40
The Base Class Library Runtime Assemblies 41
The MFDeploy Tool 41
The TFConvert Tool 45
Summary 46

CHAPTER 4 Introducing the .NET Micro Framework Base Class Library 47

Text Output for Diagnostics 47
Pausing Program Execution 49
Setting and Getting the System Time and Time Zone 50
Using Timers 51

Using Strings 54
The System.String Class 54
Concatenating Strings 56
Encoding Strings 57
Using Arrays 58
Multidimensional Arrays 58
Combining Byte Arrays 58
Extracting Ranges from Arrays 59
Combining and Copying Nonbyte Arrays 59
Integer and Byte Arrays 60
Using Collections 61
Understanding the ArrayList Class 61
Using the ArrayList Class 62
Working with Numbers 63
Real Decimal Numbers 63
Hexadecimal Number Output 63
Parsing Numbers 64
Mathematical Functions 66
Exception Handling 67
Throwing Exceptions 68
Catching an Exception 69
The finally Block 70
Handling Multiple Exception Types 70
Getting Information from an Exception 71
Rethrowing Exceptions 71
Summary 72

CHAPTER 5 Accessing Hardware 73

What Are GPIO Ports? 73
Output Ports 74
Input Ports 77
Interrupt Ports 79
Triggering an Edge 80
Triggering to a Level 81
Tristate Ports 82
The RS232 Serial Port 83
The SerialPort Class 83
Sending Data 85
Receiving Data 86
The Request-Response Approach 87

Pin Usage and Reservation 88
Why Reserve Pins? 88
The HardwareProvider Class 88
Custom Hardware Providers 89
The I²C Bus 91
Accessing I²C Devices with the .NET Micro Framework 93
10-Bit Addressing 95
A Managed Driver for the TI TMP100 Temperature Sensor 96
Hardware Providers for I²C 101
The SPI Bus 101
Accessing SPI Devices with the .NET Micro Framework 102
A Managed Driver for the AD124S101 AD Converter 104
Hardware Providers for SPI 108
Other Hardware 108
PWM 108
Mass Storage Devices with a File System 109
USB 109
Obtaining Power Supply and Charging State Information 109
Obtaining Processor Speed Information 113
System Identification 115
Summary 116

CHAPTER 6 Networking 117

Sockets 118
The Socket Class 118
TCP 120
UDP 125
Network Configuration 127
Learning More About Sockets 129
Web Services for Devices 130
DPWS Architecture 130
Discovery 131
Description 146
Control 153
Events 167
Learning More About DPWS 170
Summary 170

CHAPTER 7 Wireless Communication 173

Wireless LAN 173
Bluetooth 174
ZigBee 175
Z-Wave 176
Comparing Wireless Technologies 177
Links to Bluetooth, ZigBee, and Z-Wave OEM Modules 178
Summary 178

CHAPTER 8 Cryptography 179

XTEA: A Symmetric Algorithm 179
Using XTEA 180
XTEA for the Full .NET and .NET Compact Frameworks 181
RSA: An Asymmetric Algorithm 181
Common Key Pair Scenarios 181
Creating Private and Public Keys 183
Encryption and Decryption with RSA 186
Creating and Verifying Signatures 186
Summary 187

CHAPTER 9 Advanced .NET Micro Framework Programming 189

Multithreading and Synchronization 189
Using Threads 190
Synchronization 191
Events 193
Weak Delegates 194
Execution Constraints 195
Binary Serialization 197
Usage 197
Storage Format and Configuration 200
Configuring Serialization with Attributes 202
Notes, Hints, and Lessons Learned 206
Exchanging Data with Other Devices 206
Storing Data Permanently in Flash Memory 207
Summary 211

CHAPTER 10 Globalization and Localization 213

Resources 213
Creating and Editing Resources 214
Working with Resources 219
Cultures 219
Creating and Installing Custom Cultures 220
Formatting Dates, Times, and Numbers 225
Persisting the Current Culture 228
Multilingual Applications 228
Satellite Assemblies 228
Creating Satellite Assemblies 229
Changing the Language at Runtime 231
Summary 233

CHAPTER 11 Graphics and Graphical User Interfaces 235

Drawing on the Display Using the Bitmap class 235
Using Bitmaps 238
Flushing Bitmaps onto the Display 239
Using Colors 240
Drawing Rectangles 245
Drawing Ellipses 250
Drawing Images 250
Drawing Text 255
Creating Complex User Interfaces with Interaction 261
WPF 261
Your First WPF Application 261
The Application Class 264
Display Element Hierarchy 265
Laying Out Elements with a Panel Container 268
Backgrounds for Controls Using Brushes 275
Drawing Borders Around Elements 278
Displaying Shapes 282
The Text Element 283
The TextFlow Element 284
The ScrollViewer Element 287
List Box Menus 291
Handling User Input and Events 307
Creating Custom UI Elements 311
Summary 314

CHAPTER 12 Hardware Emulation 317

Why Use Hardware Emulation? 317
Which Hardware Can Be Emulated? 317
User Interfaces for Emulators 317
The Extensible Emulator 318
Getting Started with the Emulator 318
Emulator Project Templates 319
Emulator Components 322
The EmulatorComponent Class 323
Sharing Reusable Emulator Components 324
Creating a New Library Project for Emulator Components 325
XML Configuration 326
Programmatic vs. XML Configuration 326
Registering Components 328
Deleting or Updating Existing Components 333
Component Properties with Simple Types 335
Parsing Property Values 337
The format Specifier 339
Complex Data Types 339
The type Specifier 341
Array Properties 343
Component Collections 346
More About the Emulator 348
Members of the Emulator Class 348
The Emulator Start-up Process 350
Linked Components 350
Addressing Registered Components 351
Debugging Emulators 351
Command-Line Options 355
The Emulator and the Registry 356
Summary 359

CHAPTER 13 Emulator Components 361

GPIO Ports 361
XML Configuration 361
Input Ports 362
Output Ports 363
Synchronizing the Emulator and GUI Thread 365
The MaxPorts Property of GpioCollection 366

The Serial Port 368
Communication Between Emulator and Application 369
Routing Data to and from Another Windows Application 372
Exchanging Serial Data with a Real Hardware Component 375
Exchanging Serial Data with HyperTerminal 378
Exchanging Data with Sockets Acting As a Client 381
The I2C bus 383
The I2cDevice Component 383
The I2cBus Component 385
An Emulator Component for the TI TMP100 Temperature Sensor 385
The SPI Bus 388
The SpiDevice Component 389
The SpiBus Class 390
An Emulator Component for the ADC124S101 AD Converter 391
Emulator Timing 393
Simulating the Processor Speed 393
Continuations 394
Completions 394
Timers 394
An Oscillator Component 395
Memory Management 398
Emulating Persistent Flash Memory 403
Simulating a Battery Cell 405
LCD Displays 406
Summary 408

INDEX 409

About the Author

JENS KÜHNER works as principal software engineer for Vallon GmbH in Germany, a company that develops and manufactures metal detectors and ferrous locators. He creates software for data acquisition and evaluation using the .NET Framework and .NET Compact Framework. Since this software must be incorporated closely with the detectors' hardware, an interest in embedded systems was only natural.

Jens has been involved with the .NET Micro Framework from the very start, when he saw it presented at MEDC Europe. Since then, he's been an active beta tester of the technology and a regular contributor to the .NET Micro Framework forum.

You can reach him through his blog at `http://bloggingabout.net/blogs/jens`.

About the Technical Reviewer

FABIO CLAUDIO FERRACCHIATI is a senior consultant and a senior analyst/developer using Microsoft technologies. He works for Brain Force (`http://www.brainforce.com`) at its Italian branch (`http://www.brainforce.it`). He is a Microsoft Certified Solution Developer for .NET, a Microsoft Certified Application Developer for .NET, a Microsoft Certified Professional, and a prolific author and technical reviewer. Over the past ten years, he's written articles for Italian and international magazines and coauthored more than ten books on a variety of computer topics. You can read his LINQ blog at `http://www.ferracchiati.com`.

Acknowledgments

Writing this book was fun and immensely rewarding, but it took a great deal of effort and time to write. I would like to thank a number of individuals for supporting me and making this book possible.

First, I want to thank my wife, Iryna, and my son, Marek, who suffered most from my absence, for their love, tolerance, and patience. Iryna took care of so many things in order to make my life frictionless while I was writing. Without her active and emotional support, this project would never have gotten completed.

Thanks a lot to my parents, Bärbel and Helmut, for buying me my first computer. They always supported me and believed in me without knowing where these endless hours in front of the screen would end up, and they shaped me to be the person I am.

I'd like to thank the entire .NET Micro Framework team for developing this great technology and especially Zach Libby, Jonathan Kagle, Jim Mateer, and Lorenzo Tessiore for providing me beta bits as well as their advice, feedback, and support. I was very happy that Microsoft and Digi International provided me modules and development kits. Thanks to Frank Prengel (embedded developer evangelist at Microsoft) for providing me photographs and illustrations.

Next, a special mention goes to my boss at Vallon GmbH for believing in and supporting innovative ideas and for letting me always use the latest tools and technologies.

Finally, thanks to my technical reviewer Fabio Ferracchiati for his great suggestions to improve the overall quality of this book. I am deeply indebted to the whole Apress team, especially copy editor Heather Lang and production editor Elizabeth Berry for making my words easier to read.

Introduction

It all started at Microsoft's European Mobile and Embedded Developers Conference (MEDC Europe) 2006 in Nice, France. At this event, I saw the .NET Micro Framework presented for the first time in a session by Jonathan Kagle and Lorenzo Tessiore. As a .NET programmer for desktop and smart device applications, I was very impressed by the idea of being able to program embedded microcontrollers with my everyday development tool and programming language: Microsoft Visual Studio and C#.

I got a CD with the not-yet-released .NET Micro Framework SDK 1.0 from Lorenzo after the presentation; the emulator it included was not customizable and was specially built for the Sumo Robot contest that took place at the conference. The contest's goal was to program a Sumo robot (a small robot supporting the .NET Micro Framework) with Visual Studio and C# so that it was intelligent enough to react to sensor input and push an enemy from the battlefield.

Instead of going to the beach in Nice in the evening, I stayed in my hotel room and tweaked the software development kit's (SDK's) emulation mechanism to launch my own emulator. My first emulator just indicated the activity of a general purpose input/output (GPIO) port on the emulator's user interface using a check box. This allowed me to write my first .NET Micro Framework application that toggled a GPIO port and run it on my first emulator.

Since then, I have been an active beta tester of this technology and a regular contributor to the .NET Micro Framework forums. This passion, combined with the lack of good documentation and practical samples and the users' questions in the forum, motivated me to write this book for you.

What makes a developer productive and efficient? It is the combination of the right tools and the knowledge and skills to use them. When you use .NET Micro Framework devices, the extensible emulator, and the base class library with Visual Studio and the C# programming language, you're using powerful and modern tools. To improve your knowledge of and skill in efficiently using these tools, this book has the ambitious goal of being the best tutorial and reference available for programming with the .NET Micro Framework. Many books just scratch the surface instead of diving deeply into a topic and providing practical samples; those books are over when the fun is just beginning. I know this book is different, and I hope you find plenty of information in it that you can use to create many powerful and effective embedded applications.