

Contents

Introduction	xxi
Chapter 1: Windows Embedded CE	1
What Is Embedded?	1
Embedded Devices	1
Embedded Software	2
Embedded Devices and Software	3
The Windows Embedded Family	3
Windows XP Embedded	4
Windows Embedded Point of Service	4
Windows Embedded CE 6.0	4
Modular and Compact OS	5
Real-Time Operating System	6
Supported Hardware	6
New Features in CE 6.0 R2	7
Customizable UI	8
Wired and Wireless Connectivity	8
Graphics and Multimedia	9
Multilanguage, International Localization	9
Real-Time Communication and VoIP	10
OS Design Templates	11
Developing CE Applications	12
Testing and Debugging	12
What Can Windows Embedded CE Do?	13
Summary	14
Chapter 2: Development Environment and Tools	15
Windows Embedded CE Platform Builder	15
Windows Embedded CE 6.0 Installation	16
Supported Processors	17
Installation Sequence	18
Quick Fix Engineering and Update	19
Windows Embedded CE Terminology	20
Windows Embedded CE Environment Variables	23

Contents

Windows Embedded CE Files and Directories	32
Third-Party Components	35
Building CE Runtime Images	35
Summary	37
Chapter 3: Board Support Package	39
BSP Overview	39
BSP Development	42
Cloning the Device Emulator BSP	43
Cloning the CEPC BSP	44
Cloning the ICOP_eBox4300_60E BSP	52
BSP Components, Files, and Folders	53
Adding Files and Modules to the BSP	58
Summary	60
Chapter 4: Building a Customized CE 6.0 Runtime Image	61
Creating the Initial OS Design	62
The OS Design Wizard	63
OS Design Project Folders and Files	69
OS Design Project View from VS2005 IDE	70
Generating the OS Runtime Image	76
OS Runtime Image for MyCEPCBSP	78
OS Runtime Image for MyeBox4300BSP	80
Summary	83
Chapter 5: Connecting to Target Device	85
Target Device Connectivity	85
Connecting to Emulator	86
Creating the MyEmulator Target Device Profile	86
Configuring the MyEmulator Target Device Settings	88
Downloading Runtime Image to the Emulator	90
Connecting to eBox-4300-MSJK	91
Connecting to eBox-4300-MSJK with DHCP	92
Connecting to eBox-4300-MSJK with Static IP	99
Connecting to the CEPC	105
Creating the CEPC Boot Floppy Disk	105
Connecting to CEPC with the Serial Port	110
Connecting to CEPC with Ethernet	112
Summary	112

Chapter 6: Debugging and Debugging Tools	113
Debugging Environment	114
CETK	114
CoreCon	114
Debugging and Release Configuration	114
Debugging the OS Design Build	115
The CE Build Process	116
Build Error — Missing File	116
Remote Tools	120
Remote File Viewer	121
Remote Heap Walker	124
Remote Zoom-In	126
Remote Process Viewer	127
Remote Registry Editor	129
Remote System Information	132
Remote Performance Monitor	132
Remote Spy	135
Remote Kernel Tracker	138
Remote Call Profiler	140
Remote Target Control	141
Serial Debug	144
Summary	147
Chapter 7: Boot Loader Overview	149
What Is a Boot Loader?	149
BIOS Loader—x86 BIOS Loader	150
Eboot Loader	150
Loadcepc	151
Romboot Loader	152
Sboot Loader	152
BIOS Loader for x86 Devices	152
x86 Device Boot Process	152
BIOS Loader Code	153
Building the BIOS Loader Code	154
Summary	162
Chapter 8: The Registry	163
Windows Embedded CE Registry	163
RAM-Based Registry	165
Hive-Based Registry	165

Contents

Hive-Based Registry Triggers Two Boot Phases	166
Persistent Registry with Hive-Based Registry	166
Registry Flushing	169
Windows Embedded CE Registry Files	170
Registry for Windows Embedded CE Components	170
Serial Debugging	173
Useful Registry References	175
Auto-Flush and Save Registry Settings	175
Device Name for USB Removable Storage	175
Disable Suspend from the Start Menu	175
Internet Explorer Startup Page	175
Static IP Address	175
Windows Embedded CE Device Name	176
Accessing the Registry	176
Summary	176
Chapter 9: Testing with CETK	177
Windows Embedded CE Test Kit	177
CETK Tests	178
Connecting to CETK with KITL and Platform Manager	179
Connecting CETK to Target Device Using Sockets	188
Summary	196
Chapter 10: Application Development	197
The Development Landscape	198
The Desktop Windows Development Environment	199
The CE Development Environment	199
The .NET Compact Framework	202
Other Third-Party Libraries	204
New Breed of Embedded Devices	204
Summary	205
Chapter 11: Visual C# 2005 Applications	207
Developing C# Applications for CE	207
Configuring and Building the Runtime Image	208
Downloading the Runtime Image to the Emulator	211
Creating the C# Smart Device Application Project	212
Adding Controls to the Application	213
Adding Codes to the Application	214
Connecting to the Target Device	214

Deploying C# Application to Target Device	219
Debugging the C# Application on Target Device	220
Summary	222
Chapter 12: VB 2005 Applications	223
<hr/>	
Developing VB Applications for CE	224
Configuring and Building the Runtime Image	224
Establishing a Connection to the eBox-4300	229
Downloading Runtime Image to eBox-4300	233
Creating a VB 2005 Smart Device Application	234
Establishing a Connection to the eBox-4300	242
Deploying the SerialPortApp to the eBox-4300	246
Testing the SerialPortApp on the eBox-4300	247
Summary	249
Chapter 13: Native-Code Applications	251
<hr/>	
Visual C++ 2005 Native Applications	251
The Autolaunch Component	252
Configuring and Building the Runtime Image	254
Configuring and Building an SDK	258
Downloading a Runtime Image to the Emulator	259
Creating the Visual C++ 2005 Smart Device Application	261
Connecting to the Emulator with CoreCon	263
Deploying Visual C++ 2005 Application to the Emulator	266
Platform Builder Native Application	266
CE Subproject Wizard	266
Deploying the PBNativeApp Application	269
Summary	271
Chapter 14: Autolaunch Applications	273
<hr/>	
Autolaunch with the Registry	273
Autolaunch with Windows\Startup	276
The AutoLaunchApp Utility	280
Summary	290
Chapter 15: Customizing the UI	291
<hr/>	
Input and Output for the CE Device	292
The Input	292
The Output	293

Contents

Custom UI for the CE	294
CE with NMD Custom UI	295
VB 2005 Application as the Custom UI	300
Configure the OS Design for Testing	300
Develop the VB 2005 Application	301
Add the VB 2005 Application to the OS Design	304
Summary	306
Chapter 16: Thin-Client Applications	307
The Thin-Client Terminal	307
The Windows Thin Client OS	308
The Initial WTC OS Design	309
Configuring a Target Device and Downloading an Image	311
Customizing the WTC OS Design	313
The Autolaunch Subproject	316
Disabling the Explorer Shell	323
Summary	326
Chapter 17: Home Automation Applications	327
Home Automation Control	327
Control Applications	329
Controllers, Relays, Switches, and Sensors	330
Hardware and Peripherals	330
The eBox-4300-MSJK	330
Phidget Devices	331
BSP and Software Library	335
Electronic 101 for Input and Output	336
The eBoxPhidget OS Design	337
Development Environment	337
Creating the OS Design	338
Additional Needed Components	339
Configuring the Registry	339
Configuring the Build Option and Build	340
Target Device Connectivity and Download	341
Configuring and Generating the SDK	343
Installing the eBoxPhidgetSDK	343
Home Automation Applications	343
Temperature Sensor Application	343
Preparing eBox-4300-MSJK for Application Deployment	348
Remote Display Application	350

Deploying Applications to eBox-4300-MSJK	351
Phidget Relay Switching Application	356
Other Phidget Sensors	357
Summary	360
Chapter 18: RFID Security Access Control Applications	361
<hr/>	
Radio Frequency Identification — RFID	361
The PhidgetRFID Reader	362
RFID Reader Applications	362
Hardware Component for the Application	362
Software Component for the Application	363
Creating the RFIDApp	363
Summary	370
Chapter 19: Robotic Applications	371
<hr/>	
The Stinger CE Robot	371
The Serializer WL Robot Controller	373
How Does the Serializer Work?	374
Simple Robotic Control Applications	375
Windows Embedded CE Design	375
Robotic Application Using the Serial Port Class	375
Source Code Listing for SerialPortClass Project	383
Robotic Application Using the Serializer .NET Library	387
Launching the Robotic Application	393
Configuring the OS Design for SerialPortClass Application	393
Launching <code>serialportclass.exe</code> on the Stinger CE Robot	394
Configuring the OS Design for SerializerNET Application	394
Launching <code>serializernet.exe</code> on the Stinger CE Robot	395
Summary	395
Chapter 20: Deploying a CE 6.0 Device	397
<hr/>	
Windows Network Projector	398
Windows Network Projector OS Design	399
Creating the Initial OS Design	399
Configuring, Customizing, and Building the OS Design	400
Target Connectivity and OS Image Download	401
Cloning the Pictor Application	402
Configuring to Boot from BIOS Loader	409
Windows Network Projector Usage	415
Summary	416

Contents

Appendix A: Windows Embedded CE References	417
Appendix B: Installation and Software	423
Appendix C: Sample Applications and OS Design Projects	427
Index	435