

# USB COMPLETE FOURTH EDITION

Add USB to your designs

Access devices with .NET

Use On-The-Go and  
wireless technologies

Discover the power of  
USB 3.0 and SuperSpeed

**JAN AXELSON**

## THE DEVELOPER'S GUIDE

author of  
**SERIAL PORT COMPLETE**

---

# **USB Complete**

**The Developer's Guide**

**Fourth Edition**

**Jan Axelson**

Lakeview Research LLC  
Madison, WI 53704

---

## **USB Complete: The Developer's Guide, Fourth Edition**

by Jan Axelson

Copyright 1999-2009 by Janet L. Axelson

All rights reserved. No part of the contents of this book, except the program code, may be reproduced or transmitted in any form or by any means without the written permission of the publisher. The program code may be stored and executed in a computer system and may be incorporated into computer programs developed by the reader.

The information, computer programs, schematic diagrams, documentation, and other material in this book are provided "as is," without warranty of any kind, expressed or implied, including without limitation any warranty concerning the accuracy, adequacy, or completeness of the material or the results obtained from using the material. Neither the publisher nor the author shall be responsible for any claims attributable to errors, omissions, or other inaccuracies in the material in this book. In no event shall the publisher or author be liable for direct, indirect, special, incidental, or consequential damages in connection with, or arising out of, the construction, performance, or other use of the materials contained herein.

Many of the products and company names mentioned herein are the trademarks of their respective holders. PIC and MPLAB are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. PICBASIC PRO is a trademark of Microchip Technology Inc. in the U.S.A. and other countries.

Published by Lakeview Research LLC, 5310 Chinook Ln., Madison WI 53704

[www.Lvr.com](http://www.Lvr.com)

Distributed by Independent Publishers Group ([ipgbook.com](http://ipgbook.com)).

14 13 12 11 10 9 8 7 6 5 4 3 2 1

Printed and bound in the United States of America

ISBN13 978-1-931448-08-6

---

# Contents

<b>Introduction . . . . .</b>	<b>xvii</b>
<b>1 USB Basics</b>	
<b>Uses and Limits. . . . .</b>	<b>1</b>
Benefits for Users . . . . .	2
Benefits for Developers . . . . .	5
What USB Can't Do . . . . .	7
USB versus Ethernet . . . . .	10
USB versus IEEE-1394 . . . . .	11
<b>Evolution of an Interface . . . . .</b>	<b>11</b>
USB 1.0 . . . . .	11
USB 1.1 . . . . .	12
USB 2.0 . . . . .	12
USB 3.0 . . . . .	13
USB On-The-Go . . . . .	14
Wireless USB . . . . .	14

---

<b>Bus Components . . . . .</b>	<b>15</b>
Topology . . . . .	15
Bus Speed Considerations . . . . .	16
Terminology . . . . .	18
<b>Division of Labor . . . . .</b>	<b>19</b>
The Host's Duties . . . . .	20
The Device's Duties. . . . .	22
Bus Speeds and Data Throughput . . . . .	24
<b>Developing a Device . . . . .</b>	<b>24</b>
Components . . . . .	25
Tools for Developing. . . . .	25
Steps in Developing a Project . . . . .	25
<b>USB 3.0 Frequently Asked Questions . . . . .</b>	<b>27</b>
Features . . . . .	27
Compatibility . . . . .	28
Cables . . . . .	29
Power . . . . .	30
 <b>2 Inside USB Transfers</b>	
<b>Transfer Basics . . . . .</b>	<b>31</b>
The Essentials . . . . .	31
Purposes for Communication . . . . .	32
Managing Data on the Bus . . . . .	33
<b>Elements of a Transfer. . . . .</b>	<b>34</b>
Endpoints: the Source and Sink of Data . . . . .	34
Transaction Types . . . . .	35
Pipes: Connecting Endpoints to the Host . . . . .	36
Types of Transfers . . . . .	36
Stream and Message Pipes . . . . .	38
Initiating a Transfer. . . . .	39
<b>USB 2.0 Transactions . . . . .</b>	<b>39</b>
Transaction Phases . . . . .	41
Packet Sequences . . . . .	44
Timing Constraints and Guarantees . . . . .	45
Split Transactions . . . . .	46

<b>Ensuring Successful Transfers . . . . .</b>	<b>46</b>
Status and Control . . . . .	46
Reporting the Status of Control Transfers. . . . .	49
Error Checking. . . . .	50
<b>SuperSpeed Transactions . . . . .</b>	<b>53</b>
Packet Types. . . . .	53
Transferring Data . . . . .	54
Link Management Packets . . . . .	59

### **3 A Transfer Type for Every Purpose**

<b>Control Transfers . . . . .</b>	<b>61</b>
Availability . . . . .	61
Structure. . . . .	62
Data Size . . . . .	68
Speed . . . . .	68
Detecting and Handling Errors. . . . .	69
Device Responsibilities . . . . .	70
<b>Bulk Transfers. . . . .</b>	<b>71</b>
Availability . . . . .	71
Structure. . . . .	71
Data Size . . . . .	74
Speed . . . . .	74
Detecting and Handling Errors. . . . .	75
Device Responsibilities . . . . .	75
<b>Interrupt Transfers. . . . .</b>	<b>75</b>
Availability . . . . .	76
Structure. . . . .	76
Data Size . . . . .	77
Speed . . . . .	78
Detecting and Handling Errors. . . . .	79
Device Responsibilities . . . . .	80
<b>Isochronous Transfers . . . . .</b>	<b>80</b>
Availability . . . . .	80
Structure. . . . .	80
Data Size . . . . .	83
Speed . . . . .	84
Detecting and Handling Errors. . . . .	85
Device Responsibilities . . . . .	85

<b>More about Time-critical Transfers . . . . .</b>	<b>86</b>
Bus Bandwidth . . . . .	86
Device Capabilities . . . . .	86
Host Capabilities . . . . .	87
Host Latencies . . . . .	88
<b>4 Enumeration:</b>	
<b>How the Host Learns about Devices</b>	
<b>The Process . . . . .</b>	<b>90</b>
Enumeration Steps . . . . .	90
Device Removal . . . . .	96
Tips for Successful Enumeration . . . . .	96
<b>Descriptors . . . . .</b>	<b>97</b>
Types . . . . .	97
Device . . . . .	99
Device_Qualifier . . . . .	102
Configuration . . . . .	103
Other_Speed_Configuration . . . . .	105
Interface Association . . . . .	105
Interface . . . . .	107
Endpoint . . . . .	110
SuperSpeed Endpoint Companion . . . . .	112
String . . . . .	113
Binary Object Store and Device Capability . . . . .	114
Other Standard Descriptors . . . . .	115
Microsoft OS Descriptors . . . . .	116
Updating Descriptors to USB 2.0 . . . . .	116
<b>5 Control Transfers:</b>	
<b>Structured Requests for Critical Data</b>	
<b>Elements of a Control Transfer . . . . .</b>	<b>117</b>
Setup Stage . . . . .	118
Data Stage . . . . .	119
Status Stage . . . . .	121
Handling Errors . . . . .	122
Device Firmware . . . . .	123

<b>Standard Requests</b>	<b>124</b>
Get Status	126
Clear Feature	127
Set Feature	128
Set Address	129
Get Descriptor	130
Set Descriptor	131
Get Configuration	132
Set Configuration	132
Get Interface	133
Set Interface	133
Synch Frame	134
Set SEL	135
Set Isochronous Delay	135
<b>Other Requests</b>	<b>136</b>
Class-Specific Requests	136
Vendor-Defined Requests	136
<b>6 Chip Choices</b>	
<b>Components of a USB Device</b>	<b>137</b>
Inside a USB 2.0 Controller	138
Other Device Components	139
<b>Simplifying Device Development</b>	<b>142</b>
Device Requirements	142
Chip Documentation	143
Driver Choices	144
Debugging Tools	144
<b>USB Microcontrollers</b>	<b>147</b>
Microchip PIC18F4550	148
Cypress EZ-USB	152
ARM	156
<b>Controllers that Interface to CPUs</b>	<b>156</b>
ST-NXP Wireless ISP1582	157
PLX Technology NET2272	157
FTDI USB UART and USB FIFO	159



---

## 7 Device Classes

### Elements and Use . . . . . 163

Approved Specifications. . . . . 164

Elements of a Class Specification . . . . . 164

### Defined Classes . . . . . 165

Audio . . . . . 165

Communications . . . . . 169

Content Security . . . . . 176

Device Firmware Upgrade . . . . . 177

Human Interface . . . . . 180

IrDA Bridge. . . . . 183

Mass Storage . . . . . 184

Personal Healthcare . . . . . 189

Printer . . . . . 190

Smart Card . . . . . 192

Still Image Capture . . . . . 194

Test and Measurement . . . . . 197

Video. . . . . 198

### Implementing Non-standard Functions . . . . . 203

Choosing a Driver . . . . . 203

Using a Generic Driver . . . . . 203

Converting from RS-232 . . . . . 204

Converting from the Parallel Port . . . . . 204

PC-to-PC Communications . . . . . 205

## 8 How the Host Communicates

### Device Drivers . . . . . 209

The Layered Driver Model . . . . . 209

User and Kernel Modes . . . . . 210

### Inside the Layers . . . . . 211

Applications. . . . . 212

User-mode Client Drivers . . . . . 213

Kernel-mode Client Drivers. . . . . 213

Bus and Host-Controller Drivers. . . . . 215

<b>Writing Drivers</b> . . . . .	<b>217</b>
Kernel-mode Drivers . . . . .	218
User-mode Drivers . . . . .	218
Testing Tools . . . . .	218
<b>Using GUIDs</b> . . . . .	<b>219</b>
Device Setup GUIDs . . . . .	219
Device Interface GUIDs . . . . .	220
<b>9 Matching a Driver to a Device</b>	
<b>Using the Device Manager</b> . . . . .	<b>223</b>
Viewing Devices . . . . .	224
Property Pages . . . . .	225
<b>Device Information in the Registry</b> . . . . .	<b>226</b>
The Hardware Key . . . . .	227
The Class Key . . . . .	229
The Driver Key . . . . .	231
The Service Key . . . . .	231
<b>Inside INF Files</b> . . . . .	<b>231</b>
Structure and Syntax . . . . .	232
Device-specific Values . . . . .	235
<b>Using Device Identification Strings</b> . . . . .	<b>236</b>
Finding a Match . . . . .	238
When to Provide an INF File . . . . .	239
<b>Tools and Diagnostic Aids</b> . . . . .	<b>240</b>
Tips for Using INF Files . . . . .	240
What the User Sees . . . . .	241
<b>10 Detecting Devices</b>	
<b>A Brief Guide to Calling API Functions</b> . . . . .	<b>243</b>
Managed and Unmanaged Code . . . . .	244
Managing Data . . . . .	248
<b>Finding Your Device</b> . . . . .	<b>252</b>
Obtaining the Device Interface GUID . . . . .	252
Requesting a Pointer to a Device Information Set . . . . .	254
Identifying a Device Interface . . . . .	255
Requesting a Structure with the Device Path Name . . . . .	258
Extracting the Device Path Name . . . . .	261
Closing Communications . . . . .	261

<b>Obtaining a Handle . . . . .</b>	<b>262</b>
Requesting a Communications Handle . . . . .	262
Closing the Handle . . . . .	264
<b>Detecting Attachment and Removal. . . . .</b>	<b>265</b>
About Device Notifications . . . . .	265
Registering for Device Notifications . . . . .	265
Capturing Device Change Messages . . . . .	269
Reading Device Change Messages . . . . .	270
Retrieving the Device Path Name in the Message . . . . .	271
Stopping Device Notifications. . . . .	275
<b>11 Human Interface Devices: Using Control and Interrupt Transfers</b>	
<b>What is a HID? . . . . .</b>	<b>278</b>
Hardware Requirements . . . . .	279
Firmware Requirements. . . . .	280
<b>Descriptors. . . . .</b>	<b>281</b>
The HID Interface . . . . .	285
HID Class Descriptor . . . . .	286
Report Descriptors. . . . .	286
<b>HID-specific Requests . . . . .</b>	<b>289</b>
Get Report . . . . .	290
Get Idle . . . . .	290
Get Protocol . . . . .	291
Set Report . . . . .	291
Set Idle . . . . .	292
Set Protocol. . . . .	292
<b>Transferring Data . . . . .</b>	<b>293</b>
Writing Firmware . . . . .	293
Tools . . . . .	293
<b>12 Human Interface Devices: Reports</b>	
<b>Report Structure . . . . .</b>	<b>295</b>
Using the HID Descriptor Tool . . . . .	296
Control and Data Item Values. . . . .	296
Item Format . . . . .	296

---

<b>The Main Item Type . . . . .</b>	<b>298</b>
Input, Output, and Feature Items. . . . .	298
Collections . . . . .	302
<b>The Global Item Type . . . . .</b>	<b>302</b>
Identifying the Report . . . . .	303
Describing the Data's Use. . . . .	304
Converting Units . . . . .	306
Converting Raw Data. . . . .	307
Describing the Data's Size and Format . . . . .	311
Saving and Restoring Global Items. . . . .	311
<b>The Local Item Type . . . . .</b>	<b>311</b>
Physical Descriptors . . . . .	314
Padding . . . . .	314
<b>13 Human Interface Devices: Host Application</b>	
<b>HID API Functions . . . . .</b>	<b>315</b>
Requesting Information about the HID . . . . .	317
Sending and Receiving Reports. . . . .	317
Providing and Using Report Data. . . . .	318
Managing HID Communications. . . . .	319
<b>Identifying a Device . . . . .</b>	<b>320</b>
Reading the Vendor ID and Product ID. . . . .	321
Getting a Pointer to Device Capabilities. . . . .	323
Getting the Device's Capabilities . . . . .	324
Getting the Capabilities of the Buttons and Values. . . . .	327
<b>Sending and Receiving Reports . . . . .</b>	<b>327</b>
Sending an Output Report to the Device . . . . .	328
Reading an Input Report from the Device . . . . .	330
Writing a Feature Report to the Device . . . . .	339
Reading a Feature Report from a Device. . . . .	341
Closing Communications . . . . .	343

---

<b>14 Using WinUSB for Vendor-Defined Functions</b>	
<b>Capabilities and Limits . . . . .</b>	<b>345</b>
Device Requirements. . . . .	345
Host Requirements . . . . .	346
Device Firmware . . . . .	346
Assigning the WinUSB Driver. . . . .	349
<b>Accessing the Device . . . . .</b>	<b>350</b>
Obtaining a WinUSB Handle . . . . .	350
Requesting an Interface Descriptor . . . . .	352
Identifying the Endpoints . . . . .	354
Setting Pipe Policies. . . . .	359
Writing Data via Bulk and Interrupt Transfers . . . . .	364
Reading Data via Bulk and Interrupt Transfers . . . . .	366
Using Vendor-defined Control Transfers. . . . .	368
Closing Communications . . . . .	371
<b>15 All About Hubs</b>	
<b>USB 2.0. . . . .</b>	<b>374</b>
The Hub Repeater. . . . .	376
The Transaction Translator. . . . .	377
The Hub Controller . . . . .	383
Speed. . . . .	383
Maintaining Active Links . . . . .	385
<b>USB 3.0. . . . .</b>	<b>385</b>
Bus Speeds. . . . .	386
Components . . . . .	386
Managing Traffic. . . . .	387
<b>The Hub Class . . . . .</b>	<b>388</b>
Hub Descriptors . . . . .	388
Hub Class Requests. . . . .	388
Port Indicators. . . . .	388

<b>16 Managing Power</b>	
<b>Power Options</b>	<b>389</b>
Voltages	390
Using Bus Power	390
Power Needs	392
Informing the Host	393
Battery Charging	393
<b>Hub Power</b>	<b>396</b>
Power Sources	396
Over-current Protection	397
Power Switching	399
<b>Saving Power</b>	<b>399</b>
USB 2.0 Link Power Management	399
Suspend State	400
Sleep State	401
SuperSpeed Power Management	402
Power Management under Windows	407
<b>17 Testing and Debugging</b>	
<b>Tools</b>	<b>409</b>
Hardware Protocol Analyzers	410
Software Protocol Analyzers	412
Traffic Generators	414
<b>Testing</b>	<b>415</b>
Compliance	415
Windows Logo	422
<b>18 Packets on the Bus</b>	
<b>USB 2.0</b>	<b>425</b>
Low Speed and Full Speed Bus States	425
High Speed Bus States	428
Data Encoding	430
Staying Synchronized	431
Timing Accuracy	432
Packet Format	433
Inter-Packet Delay	434
Test Modes	435

---

<b>SuperSpeed</b> . . . . .	<b>436</b>
Data Scrambling . . . . .	436
Encoding . . . . .	436
Link Layer . . . . .	437
Reset . . . . .	438
<b>19 The Electrical and Mechanical Interface</b>	
<b>USB 2.0 Transceivers</b> . . . . .	<b>439</b>
Cable Segments . . . . .	440
Low- and Full-Speed Transceivers . . . . .	441
High-speed Transceivers . . . . .	444
Signal Voltages . . . . .	448
<b>USB 2.0 Cables</b> . . . . .	<b>450</b>
Conductors . . . . .	450
Connectors . . . . .	451
Detachable and Captive Cables . . . . .	453
Cable Length . . . . .	454
Bus Length . . . . .	455
Inter-Chip Connections . . . . .	455
<b>USB 3.0</b> . . . . .	<b>456</b>
Transmitters and Receivers . . . . .	457
Cables . . . . .	457
<b>Ensuring Signal Quality</b> . . . . .	<b>461</b>
Sources of Noise . . . . .	462
Balanced Lines . . . . .	462
Twisted Pairs . . . . .	463
Shielding . . . . .	464
Edge Rates . . . . .	464
Isolated Interfaces . . . . .	465
<b>Going Wireless</b> . . . . .	<b>465</b>
Certified Wireless USB . . . . .	466
Cypress WirelessUSB . . . . .	466
Other Options . . . . .	468

---

## 20 Hosts for Embedded Systems

<b>USB On-The-Go</b> . . . . .	<b>472</b>
Capabilities and Limits . . . . .	472
The OTG Connector . . . . .	472
The A-Device and B-Device . . . . .	472
Requirements for an OTG Device . . . . .	473
The OTG Descriptor . . . . .	479
Feature Codes for HNP . . . . .	480
<b>Other Host Options</b> . . . . .	<b>480</b>
Requirements . . . . .	481
Device Ports . . . . .	481
<b>Controller Chips</b> . . . . .	<b>482</b>
Microcontrollers . . . . .	483
Interface Chips . . . . .	484
<b>Index.</b> . . . . .	<b>487</b>





---

# Introduction

This book is for developers who are involved with designing or programming devices that use the Universal Serial Bus (USB) interface. If you are a hardware designer, if you write firmware that resides inside USB devices, or if you write applications that communicate with devices, this book is for you.

USB is versatile enough to serve a multitude of device functions. Familiar USB peripherals include mice, keyboards, drives, printers, speakers, and cameras. USB is also suitable for data-acquisition units, control systems, and other devices with specialized functions, including one-of-a-kind designs. The right choices of device hardware, software drivers and development tools and techniques can ease the path to designing devices that perform their functions without error or user aggravation. This book will guide you along the way.

## What's Inside

The USB specifications are the ultimate authority on the USB interface, but by design they omit advice, example code, and other information that applies to specific device hardware, software, and other tools and products. This book

---

bridges the gap between the specifications and real-world designs and will save you time and trouble when developing devices and the software to access them.

These are some of the questions this book answers:

- *How do USB devices communicate?* I don't attempt to restate everything in the USB specifications. Instead, my focus is on what you need to know to develop devices that communicate efficiently and reliably.
- *How can I decide if my device should use a USB interface?* Find out whether your device should use USB or another interface. If the choice is USB, you'll learn how to decide which of USB's four speeds—including USB 3.0's SuperSpeed—and which of USB's four transfer types are appropriate for your application.
- *What controller chip should my device use?* Every USB device contains an intelligent controller to manage USB communications. Dozens of silicon providers offer controller chips with different architectures and abilities. This book will help you select a controller based on your project's needs, your budget, and your preferences for chip architecture, programming languages, and tools.
- *How can applications communicate with my devices?* On a PC, an application accesses a USB device by communicating with a driver the operating system has assigned to the device. You'll learn if your device can use a class driver provided by the host's operating system. For devices that don't fit a supported class, you can explore options such as Microsoft's WinUSB driver, other generic drivers, and custom drivers. Example code shows how to detect and communicate with devices from Visual Basic and Visual C# applications.
- *What firmware does my device need to support USB communications?* Find out how to write firmware that enables your device to respond to USB requests and events and exchange data for any purpose.
- *Does my device need its own power supply?* The USB interface can provide power to devices, including charging current for battery-powered devices. Learn how to determine if a design can obtain all of its power from the bus, how to meet USB's requirements for conserving power, and how to charge battery-powered devices from the bus.
- *How can I implement wireless communications?* The Wireless USB specification defines a way for USB devices to communicate without wires. Other industry standards and vendor technologies offer additional options. Learn which technology is right for your device.

- 
- *How can my device access other USB devices?* Find out how to develop a host for an embedded system or a USB On-The-Go device that can function as both a USB device and a limited-capability host that accesses other USB devices.
  - *How can I ensure reliable communications?* All devices must respond to requests and other events on the USB port. The host computer must detect attached devices, locate appropriate drivers, and exchange data with the devices. This book provides tips, example code, and information about debugging software and hardware to help with these tasks.

To understand the material in the book, it's helpful to have some experience with digital logic, application programming for PCs and writing embedded code for peripherals. You don't have to know anything about USB.

## What's New

The core of USB has remained much the same since the release of USB 1.0 in 1996. But the interface has expanded to support faster bus speeds, improved power management, more device classes, wireless communications, dual-role devices (device and host), and more. Plus, new and improved chips and development tools have eased the task of developing devices and software to access them.

This edition is revised and updated throughout. All new in the Fourth Edition is an introduction to USB 3.0 and the SuperSpeed bus. You'll also learn how to use Microsoft's WinUSB driver to access devices that perform vendor-specific functions. Topics with major updates include device-controller chips, technologies for wireless USB communications, protocols for conserving power, and USB device classes.

I provide example code for applications in both Visual Basic and Visual C#. For device firmware, I discuss using both microengineering Labs' PICBASIC PRO™ and Microchip Technology's MPLAB® C compiler.

## Updates and More

To find out more about developing USB devices and the software that communicates with them, I invite you to visit my USB Central page at [www.Lvr.com](http://www.Lvr.com). You'll find code examples and links to articles, products, tools, and other information related to developing USB devices.

Corrections and updates to the book will also be available at [www.Lvr.com](http://www.Lvr.com). If you find an error, please let me know.

---

## Example Code

At the start of each code example, a sidehead indicates the programming language:

Sidehead	Programming Language	Provider
VB	Visual Basic .NET	Microsoft
VC#	Visual C# .NET	Microsoft
PBP	PICBASIC PRO	microEngineering Labs, Inc.
C18	MPLAB C compiler for PIC18 CPUs	Microchip Technology Inc.

The .NET code is compatible with the .NET Framework Version 2.0 and later. Example applications are available for free download from *www.Lvr.com*.

---

## Abbreviations

This book uses the abbreviations and symbols below to express quantities and units:

### Multipliers

Symbol	Description	Multiplier
p	pico	$10^{-12}$
n	nano	$10^{-9}$
$\mu$	micro	$10^{-6}$
m	milli	$10^{-3}$
k	kilo	$10^3$
K	kilo	$2^{10}$ (1024)
M	mega	$10^6$ or $2^{20}$ depending on context
G	giga	$10^9$ or $2^{30}$ depending on context

### Electrical

Symbol	Description
A	ampere
F	farad
$\Omega$	ohm
V	volt

### Time

Symbol	Description
s	second
Hz	Hertz (cycles per second)

---

## Distance

Symbol	Description
in.	inch
ft	foot
m	meter

## Data

Symbol	Description
b	bit
B	byte
bps	bits per second

## Number Systems

Binary values have a trailing subscript “b”. Example: 10100011<sub>b</sub>. An exception is when it’s clear from the context that the values are binary. Example: *Set bits 6..5 to 01.*

Hexadecimal values have a trailing “h”. Example: *A3h*.

All other values are decimal. Example: *163*.

---

## Acknowledgements

USB is much too big a topic to write about without help. I have many people to thank.

My technical reviewers provided feedback that helped make the book as complete and accurate as possible. With that said, every error in this book is mine and mine alone. A big thanks to Paul E. Berg, Greg Burk, Robert Dunstan, John Garney, Bill Jacobus, Kosta Koeman, and Matt Leptich.

Others I want to thank for their support are Phyllis Brown of J. Gordon Electronic Design, Michael DeVault of DeVaSys Embedded Systems, Traci Donnell of the USB-IF, David Flowers of Microchip Technology, Inc., Laurent Guinard of Ellisis, Tim Harvey of CWAV, Inc., Blake Henry of Bitwise Systems, John Hyde of *usb-by-example.com*, Rahman Ismail and Jeff Ravencraft of Intel Corporation, Dr. Bob Miller of Trace Systems, Inc., and Jeff Schmoyer of microEngineering Labs, Inc.

For their help with the previous editions this edition builds on, thanks to Joshua Buerger, Gary Crowell, Fred Dart, Wendy Dee, Lucio DiJasio, Keith Dingwall, Dave Dowler, Mike Fahrion, David Goll, John M. Goodman, Lane Hauck, David James, Christer Johansson, Geert Knapen, Alan Lowne, Jon Lueker, Brad Markisohn, Rich Moran, Bob Nathan, Walter Oney, Amar Rajan, Marc Reinig, Rawin Rojvanit, Glenn M. Roberts, Robert Severson, Craig R. Smith, and Dave Wright.

I hope you find the book useful and welcome your comments at *jan@Lvr.com*.



- [\*The Meaning of It All: Thoughts of a Citizen-Scientist online\*](#)
- [click Crusader Castles in the Holy Land 1097-1192 \(Fortress\) book](#)
- [\*Nighttime Is My Time online\*](#)
- [download Radio Shangri-La: What I Learned in Bhutan, the Happiest Kingdom on Earth](#)
- [read Rousseau and the Social Contract \(Routledge Philosophy Guidebooks\)](#)
  
- <http://hasanetmekci.com/ebooks/Austro-Hungarian-Albatros-Aces-of-World-War-1--Aircraft-of-the-Aces.pdf>
- <http://www.uverp.it/library/China-on-the-Mind.pdf>
- <http://test1.batsinbelfries.com/ebooks/Nighttime-Is-My-Time.pdf>
- <http://drmurphreesnewsletters.com/library/Big-Porn-Inc--Exposing-the-Harms-of-the-Global-Pornography-Industry.pdf>
- <http://test1.batsinbelfries.com/ebooks/The-Courage-of-the-Early-Morning--A-Biography-of-the-Great-Ace-of-World-War-I.pdf>