

USB COMPLETE FIFTH EDITION

Includes
SuperSpeed and
SuperSpeedPlus
(USB 3.1)

JAN AXELSON

**THE
DEVELOPER'S
GUIDE**

author of
USB EMBEDDED HOSTS

USB Complete

The Developer's Guide

Fifth Edition

Jan Axelson

Lakeview Research LLC

Madison, WI 53704

USB Complete: The Developer's Guide, Fifth Edition

by Jan Axelson

Copyright 1999-2015 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.

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

janaxelson.com

Distributed by Independent Publishers Group (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 978-1-931448-28-4



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.

The USB interface is versatile enough to serve just about any device function. 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 help you design devices that perform their functions without errors 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 the specification documents omit implementation tips, example code, and 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.

These are some of the questions this book answers:

- *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 five speeds—including USB 3.1's SuperSpeed and SuperSpeedPlus—and which of USB's four transfer types are appropriate for your application.

- *What controller hardware should my device use?* Every USB device contains an intelligent controller to manage USB communications. A variety of chip companies offer controller hardware 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 PCs, applications access a USB device by communicating with the driver the operating system has assigned to the device. You'll learn if your device can use a class driver provided by the host system'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 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?* A variety of USB and other industry standards and technologies enable USB devices to communicate wirelessly. 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 operation?* 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 delivery and management, more device classes, wireless communications, support for embedded systems that access USB devices, and more. New and improved chips and development tools have eased the task of developing devices and the software to access them.

This Fifth Edition is revised and updated throughout. New topics include an introduction to USB 3.1 and SuperSpeedPlus, enhanced power delivery and power management, new abilities using USB Type-C connectors, designing devices that use the WinUSB driver without requiring a vendor-provided INF file, new device classes, and how to use free debugging tools.

Much of the information in this book applies to any device hardware and host computer. The example code for applications uses Visual C#.

Updates and More

To find out more about developing USB devices and the software that communicates with them, I invite you to visit my website, *janaxelson.com*. You'll find code examples and links to articles, products, tools, and other information related to developing USB devices. The website includes a PORTS forum where you can ask questions and discuss topics related to USB and other interfaces.

Corrections and updates to the book will also be available at *janaxelson.com*. If you find an error, please let me know.

Example Code

The .NET example code in this book is compatible with the .NET Framework Version 4.5 and later.

Example applications are available for free download from *janaxelson.com*.

Acknowledgments

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. For their help with this edition, a big thanks to Paul E. Berg, Lane Hauck, Kosta Koeman, Dhanraj Rajput, and Rajaram Regupathy.

Others I want to thank for their support are Traci Donnell of the USB-IF, John Hyde of *usb-by-example.com*, and Jeff Ravenscraft and Brad Saunders of the USB 3.0 Promoter Group.

For their help with the previous editions this edition builds on, thanks to Phyllis Brown, Greg Burk, Joshua Buerger, Gary Crowell, Fred Dart, Wendy Dee, Michael DeVault, Lucio DiJasio, Keith Dingwall, Dave Dowler, Robert Dunstan, Mike Fahrion, David Flowers, John Garney, David Goll, John M. Goodman, Laurent Guinnard, Tim Harvey, Blake Henry, Rahman Ismail, Bill Jacobus, David James, Christer Johansson, Geert Knapen, Matt Leptich, Alan Lowne, Jon Lueker, Brad Markisohn, Dr. Bob Miller, Rich Moran, Bob Nathan, Walter Oney, Amar Rajan, Marc Reinig, Rawin Rojvanit, Glenn M. Roberts, Jeff Schmoyer, Robert Severson, Craig R. Smith, and Dave Wright.

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

Abbreviations

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

Multipliers

Symbol	Description	Multiplier
p	píco	10^{-12}
n	nano	10^{-9}
μ	micro	10^{-6}
m	milli	10^{-3}
k	kilo	10^3
K	kilo	2^{10} (1024)
M	mega	10^6
G	giga	10^9

Electrical

Symbol	Description
A	ampere
F	farad
Ω	ohm
V	volt
W	Watt

Time

Symbol	Description
s	second
Hz	Hertz (cycles per second)

Distance

Symbol	Description
m	meter
mm	millimeter

Data

Symbol	Description
b	bit
B	byte
bps	bits per second

Number Systems

Binary values have a trailing subscript *b*. Example: 10100011_b

Hexadecimal values have a leading 0x. Example: 0xA3

All other values are decimal.

Contents

Introduction.....	xv
1 USB Basics	1
Uses and limits	1
Benefits for users	2
Benefits for developers	5
Addressing USB's limits	7
USB and Ethernet.....	10
USB and Thunderbolt.....	10
Evolution of an interface	10
USB 1.0.....	11
USB 1.1.....	11
USB 2.0.....	11
USB 2.1.....	12
USB 3.0.....	14

Table of Contents

USB 3.1	14
Embedded Host and On-The-Go	15
Bus components.....	15
Topology.....	16
Bus speed considerations	17
Terminology.....	18
Division of labor	20
Host responsibilities	20
Device responsibilities	22
Bus speeds and data throughput.....	24
Developing a device	25
Components	25
Tools for developing.....	25
Steps in developing a project	26
USB 3.1 essentials.....	27
Features.....	27
Compatibility.....	28
Cables.....	29
Power	30
2 Inside USB Transfers	33
Transfer basics	33
Essentials.....	33
Purposes for communication.....	34
Managing data on the bus	35
Elements of a transfer.....	36
Endpoints: the source and sink of data	36
Transaction types.....	37
Pipes: connecting endpoints to the host.....	37
Transfer types	38
Stream and message pipes.....	39
Initiating a transfer	40
USB 2.0 transactions	41
Transaction phases	43
Packet sequences.....	45
Timing constraints and guarantees.....	46
Split transactions	46
Ensuring successful transfers	47
Status and control	47
Reporting the status of control transfers.....	50

Error checking.....	50
Enhanced SuperSpeed transactions	53
Packet types	53
Transferring data.....	55
Link Management Packets.....	61
3 A Transfer Type for Every Purpose	63
Control transfers	63
Availability.....	63
Structure.....	64
Data size	70
Speed.....	70
Detecting and handling errors	71
Device responsibilities.....	72
Bulk transfers	73
Availability.....	73
Structure.....	73
Data size	76
Speed.....	76
Detecting and handling errors	77
Device responsibilities.....	77
Interrupt transfers	77
Availability.....	78
Structure.....	78
Data size	79
Speed.....	79
Detecting and handling errors	81
Device responsibilities.....	81
Isochronous transfers	81
Availability.....	82
Structure.....	82
Data size	85
Speed.....	86
Detecting and handling errors	86
Device responsibilities.....	87
More about time-critical transfers	87
Bus bandwidth.....	87
Device capabilities	88
Host capabilities.....	88
Host latencies	89

4	Enumeration: How the Host Learns about Devices.....	91
	Events and requests	92
	Getting to the Configured state	92
	Device removal	98
	Tips for successful enumeration	98
	Descriptors.....	99
	Types	99
	Device	101
	Device_qualifier	105
	Configuration	105
	Other_speed_configuration.....	106
	Interface association.....	107
	Interface	109
	Endpoint	111
	SuperSpeed endpoint companion	115
	SuperSpeedPlus isochronous endpoint companion.....	116
	String	117
	Binary device object store (BOS) and device capability	118
	OTG descriptor	120
	Microsoft OS descriptors.....	120
	Updating descriptors to USB 2.0.....	121
	Updating descriptors to USB 3.1.....	121
5	Control Transfers: Structured Requests for Critical Data	123
	Elements of a control transfer	123
	Setup stage	124
	Data stage.....	125
	Status Stage	127
	Handling errors.....	128
	Device firmware	128
	Standard requests	130
	Get Status	132
	Clear Feature	133
	Set Feature	134
	Set Address.....	135
	Get Descriptor.....	136
	Set Descriptor	137
	Get Configuration	138
	Set Configuration.....	138
	Get Interface	139

Set Interface	139
Synch Frame	140
Set SEL	141
Set Isochronous Delay	141
Other requests	142
Class-specific requests	142
Vendor-defined requests	142
6 Chip Choices	143
Components of a USB device	143
Inside a USB 2.0 controller	144
Other device components	145
Simplifying device development	147
Device requirements	147
Documentation and example code	148
Host driver	149
Development boards	149
USB microcontrollers	152
Microchip PIC18	153
Cypress EZ-USB	157
ARM processors	159
Controllers that interface to CPUs	160
Maxim MAX3420E	160
PLX Technology USB 3380	161
FTDI interface chips	162
7 Device Classes	165
Purpose	165
Approved specifications	166
Elements of a class specification	166
Defined classes	167
Audio	167
Audio/Video	170
Billboard	172
Communications	172
Content security	179
Device firmware upgrade	180
Human interface	182
IrDA bridge	184
Mass storage	186
Personal healthcare	189

Printer.....	190
Smart card	192
Still image capture.....	194
Test and measurement	196
Video	198
Classes defined by other specifications	202
Implementing non-standard functions.....	204
Choosing a driver.....	204
Using a generic driver.....	204
Converting from RS-232	204
Converting from the parallel port	206
Connecting two PCs	206
8 How the Host Communicates	209
Device drivers	209
The layered driver model	209
User and kernel modes	210
Inside the layers.....	211
Applications.....	211
User-mode client drivers.....	213
Kernel-mode client drivers	213
Low-level host drivers.....	214
USB 3.0 drivers	215
USB 2.0 drivers	216
Writing drivers	217
Kernel mode.....	217
User mode	218
Testing tools	218
Using GUIDs.....	218
Device setup GUIDs.....	219
Device interface GUIDs	220
9 Matching a Driver to a Device	221
Using Device Manager	221
Viewing devices	221
Property pages	223
Device information in the registry	224
The hardware key.....	224
The class key	225
The driver key	227
The services key	228

Using INF files	229
Driver signing requirements	229
File structure	232
Inside an INF file	232
Using device identification strings	234
Finding a match	237
When to provide an INF file	237
Tools and diagnostic aids	238
Tips for using INF files	238
What the user sees	239
10 Detecting Devices	241
A brief guide to calling API functions	241
Managed and unmanaged code	242
Managing data	246
Finding a device	248
Obtaining the device interface GUID	249
Requesting a pointer to a device information set	250
Identifying a device interface	251
Requesting a structure with the device path name	252
Extracting the device path name	254
Closing communications	254
Obtaining a handle	255
Requesting a communications handle	255
Closing the handle	257
Detecting device attachment and removal	257
Using WMI	257
Adding a handler for newly arrived devices	257
Detecting the target device	259
Adding a handler for removed devices	261
11 Human Interface Devices: Capabilities	263
What is a HID?	263
Hardware requirements	264
Firmware requirements	265
Descriptors	266
The HID interface	266
HID class descriptor	269
Report descriptors	269
HID-specific requests	272
Get Report	273

Get Idle.....	273
Get Protocol.....	274
Set Report.....	274
Set Idle.....	275
Set Protocol.....	275
Transferring data.....	276
Writing firmware.....	276
Tools.....	276
12 Human Interface Devices: Reports.....	277
Report structure.....	277
Control and data item values.....	278
Item format.....	278
The Main item type.....	279
Input, Output, and Feature items.....	279
Collections.....	282
The Global item type.....	283
Identifying the report.....	283
Describing the data's use.....	285
Converting units.....	286
Converting raw data.....	287
Describing the data's size and format.....	290
Saving and restoring Global items.....	290
The Local item type.....	290
Physical descriptors.....	293
Padding.....	293
13 Human Interface Devices: Host Application.....	295
HIDClass support routines.....	295
Requesting information about the HID.....	297
Sending and receiving reports.....	297
Providing and using report data.....	298
Managing HID communications.....	298
Identifying a device.....	299
Reading the Vendor ID and Product ID.....	300
Getting a pointer to device capabilities.....	301
Getting the device's capabilities.....	302
Getting capabilities of buttons and values.....	303
Sending and receiving reports.....	304
Sending Output reports with interrupt transfers.....	304
Reading Input reports with interrupt transfers.....	307