



CYPRESS

A USB Resource List: Tools, Documents, Web Sites, Courses and Techniques

Introduction

This document is intended to help you assemble the many documents and pieces of equipment you will need to develop a successful EZ-USB product.

In Cypress' USB Applications department, we've had much experience in this area, and we hope you will follow our advice. We believe we can save you many headaches and months in your schedule by following the path we've outlined here.

Resources within each category are listed in approximate order of importance, most important first.

For all **technical** questions about Cypress' USB products not covered in this or the documents referenced, please email usbapps@cypress.com.

For questions about **price and availability**, please contact your local Cypress Sales Office.

Note: Our convention in this note is to use **exact**, not **approximate**, URLs (web addresses), for example

www.microsoft.com/hwdev/busbios/usbwin98.htm

not just

www.microsoft.com

Although URLs are often changed we think you will find exact URLs to normally be more helpful than approximate ones.

How To Best Use Cypress' USB Apps Support

Do:

- Email usbapps@cypress.com. This is the main email address for both the EZ-USB (mostly in San Diego, California) and M8 (mostly in Woodinville, Washington State) groups. When you email to that alias, the email is distributed to the best available person by the Queue Manager of the week.

For a number of reasons, we prefer email over phone calls. Especially, we find the level of detail that can be expressed in an email is generally superior to that which can be communicated over the phone. Email also allows us to forward you question intact to the best person for the task.

If you've been working with a particular Apps engineer on an issue, that Apps Engineer will typically stay with the issue (the "thread") until it is resolved. Hence it is OK to email that engineer **in addition** to usbapps.

Don't:

- Email individual Apps Engineers. If that engineer is out of the office, no one here will even know that your email has arrived.
- Email cyapps@cypress.com. That email address is normally used for PLD and other non-USB questions. To get to the USB experts fastest, email usbapps@cypress.com.

Tools

EZ-USB Development Kit (DK)

Unless you have already developed a USB product or you are a USB expert, a DK is essential to learn USB and the EZ-USB methodology.

If you are new to USB, we estimate that skipping this step will cost you a minimum of 2 months in your schedule. (At \$100 per (fully-loaded) engineer-hour, that's about \$32,000! At about \$500, the DK costs about half an engineer-day!)

The CD included with the DK is an essential source of documents, programming examples, etc. The source code for the example driver for Win98 and Win2K ([ezusbdrv](#)) is only available via this CD. It cannot be downloaded from our web site.

As of this writing, there are two DKs:

- EZ-USB (Series 2100) family
- EZ-USB FX (CY7C646xx) family

Both DKs use the same **shared** CD, which has all source code, schematics, etc., for both EZ-USB families.

You *can* download the CD (except for the Drivers tree) from our web site at

<ftp://download.cypress.com/cysolutions>

or go to the Development Kit pages on the Cypress USB web site starting at

<http://www.cypress.com/usb>.

Full Keil Tools

The Keil tools included with the DKs are *demo* tools. Only rarely can they be used to build a completed product.

We recommend purchasing the Full Keil Tools from Keil at the same time you purchase the DK from Cypress.

Since we have daily, hands-on experience with Keil tools, the EZ-USB DK was developed and tested with Keil tools, we do not recommend the use of tool sets other than Keil.

Limitation of the Demo Tools

The demo tools (the "4K Tools") cannot locate CODE **inside** the EZ-USB chips, hence the memory inside the EZ-USB is unused (for CODE). This is due to the fact that regardless of the linker switches, the 4K Tools' linker locates CODE starting at 0x4000, above the topmost address of internal RAM. In spite of this limitation, the 4K Tools **can** be used with no difficulty with the DK, since the DK has SRAM at 0x4000 (to 0xFFFF).



A USB Resource List: Tools, Documents, Web Sites, Courses and Techniques

Microsoft Visual C++ 5.x or VC++ 6.x

We encourage developers to use the same host-side IDEs (Integrated Development Environments) we use, namely Microsoft Visual C++ 5.x or VC++ 6.x

Unfortunately, Borland tools do not work well with Microsoft DDKs, so we do not currently recommend or support them.

Microsoft DDKs

Win98, Win2K and other DDKs can be downloaded free from the Microsoft web site at www.microsoft.com/ddk.

Note: Because of its serious limitations, Win95 OEM Service Releases (OSR) 2.1 and 2.5 are not recommended for USB development. For more information on the significant enhancements to USB in Win98 see www.microsoft.com/hwdev/busbios/usbwin98.htm.

Logic Analyzer

Of course you will need a logic analyzer to observe the details of your application circuit. Many of our Apps team prefers a logic analyzer to a debugger, because it allows one to watch the **dynamic** behavior of a program far better than a debugger.

USB Analyzer

All our USB analyzers, including USB 2.0, are CATCs ("cat-sees", Computer Access Technology Corporation). We rate the formatting of the data and ease of use as excellent. See www.catc.com.

SoftICE and DriverStudio by Compuware® NuMega

SoftICE is the best debugger we know of for device driver development.

Although we have not yet tested it, we have high hopes for the new NuMega DriverStudio.

See www.numega.com/drivercentral/driverstudio.shtml

SoftICE is included in the DriverStudio tool suite.

Other tools

Active Wire Inc. has an inexpensive (\$59 as of 7/5/2000) EZ-USB-based board product and software support for interfacing with Visual Basic and other development environments.

See www.activewireinc.com.

Courses

These are fast-moving, intensive courses. We recommend hands-on experience with WDM **before** attending. You could work through the examples in at least one WDM book. See Walter Oney's and Chris Cant's books below.

- WDM/USB Device Drivers

Formerly offered by Solsem, is now offered by Azius (see www.azius.com). The instructor is Brian Catlin, same as when the course was offered by Solsem. Is an excellent course -- highly recommended.

Other excellent courses are offered by the Azius team.

- Drivers for Windows 2000

See www.oneysoft.com.

- Windows NT/2000 Internal Architecture
Covers much of the information from David Solomon's book *Inside Windows NT, 2nd edition* (Microsoft Press). See www.solsem.com.

Documents

- EZ-USB FX TRM (Technical Reference Manual)

To download, go to the Full Speed Peripherals pages on the Cypress web site starting at www.cypress.com/usb. For a printed copy, contact your local Cypress sales office. A printed copy is also included with the EZ-USB FX Development Kit.

- EZ-USB FX Datasheet

To download, go to the Full Speed Peripherals pages on the Cypress web site starting at www.cypress.com/usb. For a printed copy, contact your local Cypress sales office. A printed copy is also included with the EZ-USB FX Development Kit.

- EZ-USB (Series 2100) TRM

To download, go to the Full Speed Peripherals pages on the Cypress web site starting at www.cypress.com/usb. For a printed copy, contact your local Cypress sales office. A printed copy is also included with the EZ-USB Development Kit.

- EZ-USB (Series 2100) Datasheet

As of this writing, the information normally in Cypress datasheets (pinouts, timing, etc.) is documented in the EZ-USB TRM. We are in the process of creating an EZ-USB (Series 2100) Datasheet, to be made available via the same means as other Cypress datasheets.

- Universal Serial Bus Revision 2.0 Specification

This is an essential reference for any USB project. See www.usb.org/developers/usb20/ to download the spec, and for many other useful documents and links.

Note: v1.1 is available for historical reference from the same site.

The following two books are the essential minimum for any USB project with host-side device driver work.

- Walter Oney, *Programming the Microsoft Windows Driver Model*, Microsoft Press, 1999.

See Walter's web site at www.oneysoft.com.

- Chris Cant, *Writing Windows WDM Device Drivers*, R&D Books; ISBN: 0879305657, July 1999. Covers WinNT 4, Win98, and Win 2000.

This book also has a good resource list in Appendix A and a much needed Glossary (a 'Rosetta stone' of Microsoft acronyms).

See Chris's web site at www.phdcc.com/wdmbook.

- Jan Axelson, *USB Complete: Everything You Need to Develop Custom USB Peripherals*. Written for Cypress CY7C63000 USB microcontroller, and requires Cypress 3650 Developer's Kit.

This is currently our favorite **USB** book. Although it focuses on the Cypress CY7C63001 chip, it is also useful for EZ-USB work, especially for HIDs (Human Interface Devices).



A USB Resource List: Tools, Documents, Web Sites, Courses and Techniques

es), which can be **any** device with a data rate of less than 64 Kbytes/s.

See Jan's web site at www.lvr.com.

- Don Anderson, *Universal Serial Bus System Architecture*, Mindshare, Inc.

This book is currently included with the EZ-USB DKs, and is a good book to get you started on USB. But for fundamentals, the Anchor EZ-USB Technical Reference Manual (TRM) is just as good in many respects; and for details, you'll need the USB spec.

- John Hyde, *USB Design By Example*, Intel/Wiley

Also see John's web site at www.usb-by-example.com.

John Hyde says "One of my goals in writing the book was to show how much can be done without having to write a device driver. There are MANY built into the operating system, and my plan was to show how to utilize these. In many cases a custom device driver is not required. Having said that I know there is a need for a 'cookbook' for a device driver and I plan to write this extra chapter and load it up onto [my] web site."

- Mike Zerkus, John Lusher and Jonathan Ward (president of Keil Software), *USB Primer, Practical Design Guide*. May through August 1999, Circuit Cellar Ink.

- J. Lyle, *USB Primer: Classes and Drivers*, Circuit Cellar 107, June 1999.

- David Solomon, *Inside Windows NT*, 2nd edition, Microsoft Press.

- Walter Oney, *Systems Programming for Windows 95*, Microsoft Press, 1996. Is especially good for documentation on .inf files. See www.oneysoft.com/errata.htm.

- Viscarola & Mason, *Windows NT Device Driver Development*, OSR (Open Systems Resources, Inc.). Order directly from OSR (www.osr.com) to make sure you get the latest printing, with the latest fixes to the text.

Except for a short Appendix C, this book does *not* cover WDM. However, this book can be useful to understand NT4 drivers, which are the basis for WDM drivers.

Web Sites

Note: A Microsoft browser is recommended for all Microsoft sites.

- www.usb.org
- www.usb.org/developers
- www.usb.org/developers/hidpage.html

These are the key web sites for USB developers. There you will also find a collection of vital HID documents.

- www.cypress.com/usb

We recommend that you immediately go to the Cypress USB site to familiarize yourself with all our USB pages. It would be a good idea to download all App Notes, print them

and use them as references throughout your project. You can download most parts of the DK CD before you purchase a DK.

- www.microsoft.com/hwdev

To download the Win98 and Win2K DDKs (Driver Development Kits). Note: the DDKs contain documentation on the .inf files.

- msdn.microsoft.com

For many developer tools in addition to the DDKs.

- www.microsoft.com/hwtest

This is a good page, because it has links to all (?) of the important Microsoft pages for hardware developers (DDKs, WHQL, .inf files, PC design and WHQL requirements. WHQL certification is essential to get a Microsoft-certified logo for your device.

- www.microsoft.com/hwdev/desguid.htm

This is an essential site to for the PC99, etc. documents, a roadmap for Microsoft's hardware requirements.

- support.microsoft.com/support

This is a useful site for Service Packs, etc.

- www.keil.com

Is extremely useful for assistance with the Keil tools, and has excellent application notes on the 8051.

(972) 735-8052

- www.phoenix.com

Phoenix has a number of software and BIOS products for USB.

- www.thesycon.de/usbio/eng/usbio.htm

Thesycon (th-e-si-con) Systemsoftware & Consulting GmbH, Ilmenau, Germany, offers a generic device driver called USBIO that enables direct access to USB devices by Win32 applications.

- www.sysinternals.com

This is an excellent site for Windows system internals. Also has a great resource list.

- www.cmkrnl.com/

Kernel Mode Systems is Jamie Hanrahan's (an Azius founder, see Courses) site for NT kernel-mode issues.

- developer.intel.com/design/litcentr/index.htm

Go to the page above, then do a Find on that page for MCS® 51/151/251 Microcontrollers (or just "51")

This is useful page for introductory 8051 manuals, which are essential if you've never worked with an 8051 processor before.

The following manual on MCS 51 is especially helpful. Read Chapter 1 early in your development project:

developer.intel.com/design/auto/mcs51/manuals/272383.htm

- www.intel.com/ial/sm/usenix/

Has a paper *A Comparison of Windows Driver Model Latency Performance on Windows NT and Windows 98*, by Erik Cota-Robles and James P. Held, Intel Architecture Labs



A USB Resource List: Tools, Documents, Web Sites, Courses and Techniques

Of course you could just go to www.google.com and type in any of a number of search terms. This actually works well for "Windows Driver Model."

Techniques

This section discusses ways to maximize your efficiency by selecting the right starting point and development environment.

Frameworks

Start all projects using the Frameworks (fw.c and associated code, contained on the DK CD). This is extremely important. (The term *Frameworks* comes from the Device Frameworks chapter (Chapter 9) of the USB spec.)

We wrote the Frameworks to:

- provide a working example of initializing the EZ-USB chip. It is not trivial to get this initialization correct.
- implement the functionality of the Frameworks Chapter 9 of the USB spec. All compliant USB device must implement this functionality to pass the USB Compatibility Workshops.

As we gain experience from working with other devices in the USB community, we continuously improve this body of code. We have gained the experience and literally coded it into the Frameworks. Why would you want to lengthen you time to market and increase your development costs by not using our extensive experience with real hosts and hubs?

If you are required or prefer to write your code in assembly language, we recommend that you start with the latest Frameworks code (in C), compile it, then use a snapshot of the resultant assembly language to start your project.

In our opinion, the best technique is to use the Frameworks code in C for the bulk (95%?) of the code, but fine tune the inner loops by writing routines in assembly language (5% ?). These assembly routines can either be external routines that are linked in (preferred) or inline (somewhat tricky to do with the Keil tools).

Which Windows OS?

Our recommended development path is to use Win2K as your main target platform, then port the code to Win98. There are many ways you can write WDM drivers for Win98 that will be very difficult to port to Win2K. However, if you start with Win2K, we believe the task of porting to Win98 will be quite easy. [Source: Brian Catlin www.azius.com, and others]

Two-machine Driver Development

Here is the current recommended Windows device driver development setup:

PC	Purpose	Operating System
Host	'Builds'	Win2K
Target	Debugging/Testing	Win2K, Win98, Win98SE, WinMe(Millennium)...any Windows OS with a reasonably robust and complete USB protocol stack

It is **possible** to use a single PC for both Host and Target, but that is increasingly falling out of favor, because:

- Especially when developing device drivers, it is likely that you will do something that wipes out your hard drive. It is much better to do that on a Target machine that has the bare necessities, than on your Host machine, where it might take you the better part of a week to rebuild the disk.
- The Target can be swapped out easily - one day desktop PC #1, next day laptop #1, then desktop PC #2, etc. Meanwhile, the host platform stays constant, a 'control' in your development.

Common Mistakes

- Attempting to use the Cypress General Purpose Driver (GPD) unchanged as the end-all be all for your application. For all but the most forgiving applications, latency issues will prevent you from achieving good results if you attempt to do all control from a Ring 3 application.
- Latching onto the GPD as your starting point. You need to investigate Driver Wizards such as those offered by Blue-Water Systems and others before committing to a multi-month driver development project.
- Thinking that driver work is trivial and letting a novice do it. Is by far the hardest part of any EZ-USB project. Hire an expert, or a very smart programmer/engineer.