



Getting Started with the CY3663 Development Kit

Contents

Introduction	1
Review of Kit Components	2
CD Installation.....	3
Directory Structure	3
Documentation Index.....	4
USB Multi-Role Device Design By Example	6
Navigator / Design Examples	6
CY16 Development Tools.....	8
USB Software Stack Support.....	12
Compliance Testing.....	12
Support	13

Introduction

Welcome to the EZ-OTG / EZ-Host Development Kit! Cypress's new embedded host and On-The-Go (OTG) products have many powerful features that allow creation of never before seen USB applications. The OTG Supplement itself opens up a new world of possibilities for USB with the creation of a system allowing point-to-point communications for mobile devices. Using EZ-OTG or EZ-Host, with multiple USB ports and two Serial Interface Engines, each with support for dynamic swapping of the host and peripheral roles, the possibilities go well beyond those envisioned with OTG.

EZ-OTG and EZ-Host also share the same 16-bit RISC processor, an 8K x 16 RAM, and 4K x 16 ROM with integrated BIOS. A configurable I/O block with a number of different options rounds out the basic features of the chips. With all of these attributes there's a lot of capability available to the developer.

Features of EZ-OTG and EZ-Host aside, implementing a dual role device can be a complex undertaking – but it doesn't have to be. Cypress has been in the USB business since its inception. We have a great deal of experience to offer our customers, and we have consolidated much of that knowledge in the CY3663 development kit.

Due to the nature of our USB host chips, we have included a lot of material in this development kit. Every USB embedded host design is different. Although most of the core components of the chips will be used by every application, there are many features that will not be needed in every design. And there are different ways to move data through the system, depending upon the overall requirements. That said, there will inevitably be pieces of information that you won't need to spend time on.

This document is aimed at providing an overview of the kit and a roadmap to guide you to the pieces of information that will help you the most. With that in mind, direct your attention to a key item in the kit: the book *USB Multi-Role Device Design By Example*. Cypress is pleased to have the support of John Hyde, a well-known and respected author in USB circles, as he employs the EZ-OTG and EZ-Host as a demonstration vehicle for the design and implementation of combination host/peripheral/OTG devices.

You can certainly jump right to the book now, but if you want to work with the corresponding example code you'll need to read through a few more sections of this document. We'll get back to the book shortly. The next sections will introduce you to the basic components of the kit and take you through the CD installation. In addition to the example code and development tools that the book refers to, there is also one tool on the CD that can be very helpful at this stage: The EZ-OTG / EZ-Host Navigator. This tool has a couple of functions, but one in particular will prove useful while you are becoming familiar with the kit. The Navigator features an HTML document viewer. You can use this portion of the tool to learn more about the chips, the hardware, and USB On-The-Go.

Review of Kit Components

To make sure that you have everything you should, here is a list of the components that are physically included in your kit, along with a brief description:

- *USB Multi-Role Device Design By Example*, introductory book by John Hyde
- Documents
In addition to the book, a few items are included in hard-copy for convenience. Descriptions of these documents are covered in the Documentation Index section.
 - Cover Letter
 - Getting Started (this guide)
 - License Agreement
 - EZ-OTG Datasheet
 - EZ-OTG Silicon Errata
 - EZ-Host Datasheet
 - EZ-Host Silicon Errata
 - CY3663 Development Kit Release Notes
- EZ-OTG and EZ-Host Development boards
These boards look very similar and are about 4" x 6" in size. Look for the names "EZ-OTG" and "EZ-Host" about an inch in from the top right corner of the respective boards. These boards can be used for stand-alone development or as daughter cards for the co-processor board below. They can also be used independently or together to see the interaction between two dual-role devices. When used in a stand-alone mode you may also see reference to them as the stand-alone board (SAB).
- Co-processor board
This board, utilizing a StrongARM 1110 processor, is used to run the code examples for EZ-OTG and EZ-Host in co-processor mode. You'll also see it referred to as the StrongARM Single Board Computer (SBC).
- CD-ROM
The CD contains all of the documentation, development tools, example code, drivers, hardware design files – everything you should need to design your embedded host device. Installation is discussed below.
- Power supplies
2 power supplies are included. They support operation of both the EZ-OTG and EZ-Host boards simultaneously. Note that if using one board in co-processor mode the StrongARM board and daughter card share the same power supply.
- Cables
Commonly required cables are included to support development:
 - Serial cable
 - 2 standard USB cables (Standard A to Standard B plug)
 - 1 mini USB cable (Mini A to Mini B plug)

CD Installation

Installation of the materials on the CD-ROM is straight forward. Insert the CD, open up My Computer and double-click the CD icon. Now double click the setup.exe icon to start.

When the installer launches, you will be presented with a dialog box that allows you to specify the destination. The default is C:\Cypress\USB\OTG-Host. Select the default or specify a location of your choice – this document will assume the default in referring to the locations of files. The installation may take up to about 20 minutes and requires just under 1GB of disk space, so make sure that you have adequate room. The large amount of data results primarily from the development tools, but there is also a significant amount associated with the Linux sources for the co-processor mode examples. At the completion of the install you will be presented with a window showing the items that have been added to the Start Menu. Drag them to the desktop to create Shortcuts if you desire.

If for any reason you decide that you need to re-install the development kit materials there are two things to keep in mind. First, if you have modified any of the examples, you should back up the modified files if you wish to keep them. Second, you first perform an un-install before re-installing the CD-ROM contents. The same precautions should also be taken in the event of an upgrade to a newer version of the development kit software.

Directory Structure

There is a lot of content included in the installation. This list goes over the higher level directory structures and does not completely explore the lower level directories or attempt to describe the location of every file. The default installation is in the location C:\Cypress\USB\OTG-Host\. If you have selected a new location for the installation, the “OTG-Host” directory structure will be laid down in the location of your choice.

```
C:\Cypress\USB\  
|-- OTG-Host  
|   |-- CY3663 Binaries  
|   |   |-- stand-alone  
|   |   |-- coprocessor  
|   |  
|   |-- Docs  
|   |   |-- OTG-Host Tools  
|   |   |   |-- ...Tools docs...  
|   |   |-- Datasheets  
|   |   |-- User's Guides  
|   |   |-- ...other docs...  
|   |  
|   |-- Drivers  
|   |   |-- Windows drivers  
|   |  
|   |-- Hardware  
|   |   |-- EZ-Host  
|   |   |-- EZ-OTG  
|   |   |-- StrongARM SBC  
|   |  
|   |-- Source  
|   |   |-- stand-alone  
|   |   |   |-- common  
|   |   |   |-- de1  
|   |   |   |-- de2  
|   |   |   |-- de3  
|   |   |   |-- de4
```

“Binaries” contains the object code images created from the source code in the “Source” directory. The binary files are grouped into the “stand-alone” and “coprocessor” subdirectories based on their function – also see “Source” below.

“Docs” contains most of the kit documentation. All items are in Adobe Acrobat Portable Document Format (.pdf). If you need the free Acrobat Reader go to Adobe’s website at www.adobe.com. Descriptions of these documents are contained in the Documentation Index immediately following this section. The “OTG-Host Tools” subfolder groups all of the documents for the development tools together.

“Drivers” contains the Windows driver files to enumerate EZ-OTG and EZ-Host as USB peripherals for development support

“Hardware” contains all of the design files used in development of the boards in the kit. There is a subdirectory for each board. Within those directories you will find .pdf schematics and Bill of Materials, Orcad source files, and Gerber databases.

“Source” contains the source code files for the Frameworks and all of the design examples.

“stand-alone” is source for the design examples that runs in stand-alone mode on the EZ-OTG/EZ-Host boards.

“common” contains the Frameworks code that is common to all of the designs. “de1” through “de4” contain the sources for the 4 design examples (described later in this document).

		-- se1	"se1" through "se10" contain the examples from John Hyde that are referenced in his book "USB Multi-Role Device Design By Example".
		-- ...	
		-- se10	
		-- coprocessor	"coprocessor" contains the source files for the coprocessor based examples. The "linux" subdirectory contains the source files for the linux kernel running on the StrongARM board, and "de_app" contains the linux portion of the examples.
		-- linux	
		-- de_app	
		--Tutorial	"Tutorial" contains the Navigator, which is the graphical tutorial and control panel application for the design examples. "Content" is a subfolder containing all of the necessary support files for the Navigator application.
		--OTG-Host Navigator.exe	
		--Content	
		--Tools	"Tools" contains the CY16 processor development tools. The tools are GNU based and so contain all of the standard GNU support materials. The "eclipse" subdirectory contains the Eclipse IDE from IBM (more information later in this document). The "utilities" subdirectory contains various utilities that you will need during the development process.
		-- cy16-020830	
		-- sa1100-020830	
		-- eclipse	
		-- Utilities	
		-- gnu-pro-00r1	
		-- usertools-00r1	

Documentation Index

As you have probably noticed from the list above or a review of the actual CD installation, there are a lot of documents included in the development kit. It is not expected that you will need to use all of this information to complete your design. We have endeavored to provide comprehensive materials that explain all of the interfaces and features. Based upon your design requirements you will be able to pick and choose the items that are appropriate for your review based on the following information. This list is also generally organized from high level documents to low level documents. Scan through it so that you know what's available when you want to come back for more information. A similar list of documents specific to the GNU development tools is included in the CY16 Development Tools section later in this document.

USB Multi-Role Device Design By Example

Location: Print and in the Docs directory

File Name: USB Multi-Role Device Design By Example.pdf

Description: As discussed above, this should be the starting point for your exploration of the Cypress chips and development kit. Spend time on it first and move to the other documents only when you need a lower level of information.

Getting Started with the CY3663 Development Kit

Location: Print and in the Docs directory

File Name: CY3663 Getting Started.pdf

Description: Introduction to the development kit components. This document.

Cypress Software License Agreement

Location: Printed and in Docs directory

File Name: CY3663 SW License Agreement.pdf

Description: This document provides the standard Cypress license for use of materials supplied in the development kit. This kit also contains elements from open sources that have their own license agreements. The document explains which licenses apply to which components.

CY3663 Hardware User's Manual

Location: Docs directory



File Name: CY3663 Hardware User Manual.pdf

Description: This document provides some detail on the hardware components in the kit. There is a section for each board in the kit that covers the basic components, as well as sections focused more on development concerns.

CY3663 Development Kit Release Notes

Location: Print and in the Docs directory

File Name: CY3663 Dev Kit Release Notes.pdf

Description: This Release Notes document focuses explicitly on the development kit materials. It covers any last minute information that relates to deviations from functionality explained elsewhere in the documentation.

GNUPro Toolkit User's Guide for Cypress Development

Location: Docs\OTG-Host Tools directory

File Name: CY16.pdf

Description: This is the primary source of information for use of the GNU development tools for the CY16 processor (the processor within the EZ-OTG and EZ-Host). Note that the folder where it is located contains a number of other documents that may also be of use as you work through the development process. These are detailed further in the CY16 Development Tools section of this document.

CY16 Binary Utilities Reference

Location: Docs directory

File Name: CY16 Binary Utilities Reference.pdf

Description: This document describes the operation of some of the commonly used CY16 utilities.

EZ-OTG Datasheet, EZ-Host Datasheet

Location: Print and in the Docs directory

File Name: CY7C67200.pdf, CY7C67300.pdf

Description: These are the official public documents that describe the respective chips. CY7C67200 is the EZ-OTG and CY7C67300 is the EZ-Host. They are focused primarily on the hardware aspects of the design, but do cover the general overview of each part. Cypress always maintains the latest documents on its website for download, so it's worth a periodic check during the design process to see if there are updated datasheets available. Go to www.cypress.com and look under "USB Embedded Hosts."

EZ-OTG Silicon Errata, EZ-Host Silicon Errata

Location: Print and in the Docs directory

File Name: CY7C67200 Errata.pdf, CY7C67300 Errata.pdf

Description: The Errata documents will reference their respective chips and will only be present if applicable. The Errata covers any significant information that did not make it into the latest release of the datasheets, or parameters that may deviate somewhat from the datasheet specifications.

CY7C67200 and CY7C67300 Hardware Design Notes

Location: Docs directory

File Name: OTG-Host Hardware Design Notes.pdf

Description: This document amplifies some of the information in the datasheet and provides additional recommendations for hardware design and layout.

Frameworks Reference Guide

Location: Docs directory

File Name: OTG-Host Frameworks Reference Guide.pdf

Description: The Frameworks is a set of common routines that almost all applications will need to make use of. The Frameworks handles most of the interface to the BIOS, which in turn handles the low level hardware and

register accesses. This document describes the structure and APIs of the Frameworks. It is intended to supplement the information in USB Multi-Role Device Design By Example.

Linux USB Driver User's Guide

Location: Docs directory

File Name: OTG-Host Linux USB Driver.pdf

Description: Included in the kit is a complete USB On-The-Go Linux driver solution for use in a co-processor environment. It includes a host controller driver, peripheral (function) driver, and a low level OTG layer. This document describes the driver architecture and implementation. It is primarily intended for customers who will use Linux in their applications, or who wish to understand the Linux solution so that they can adapt it to their own application.

BIOS User's Manual

Location: Docs directory

File Name: OTG-Host BIOS User Manual.pdf

Description: This is a detailed reference that describes the functional blocks and operation of the BIOS. The main topics include boot-up / initialization, Link Control Protocol for use in co-processor mode of operation, and the data transfer mechanisms for host mode and peripheral mode.

CY16 Programmer's Guide

Location: Docs directory

File Name: CY16 Programmers Guide.pdf

Description: This document describes the basic operation of the CY16 processor. It includes the internal registers, instruction set, timing, and also an assembly language reference.

EZ-OTG / EZ-Host Boot Code Design

Location: Docs directory

File Name: OTG-Host Boot Code Design.pdf

Description: In order to keep your CY16 code at a minimal size, this document explains how to avoid pulling in large amounts of init/exit code, the default with GNU tools.

USB Multi-Role Device Design By Example

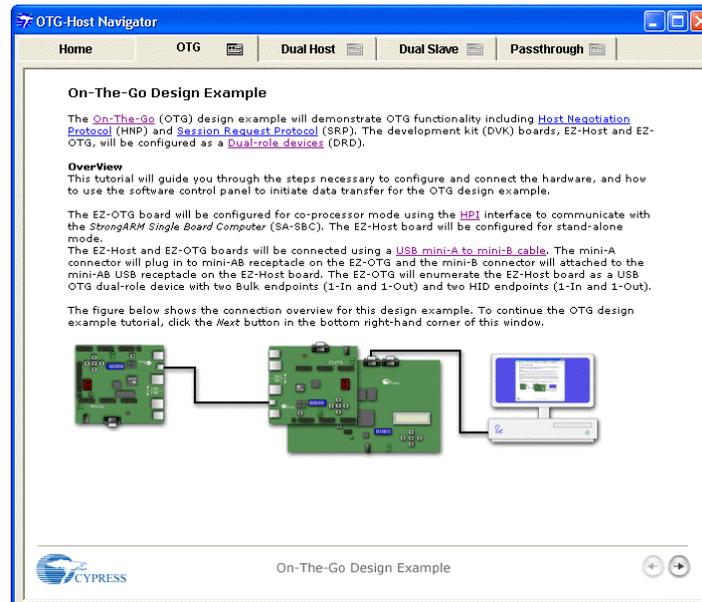
Now that we're through the installation and introductory materials, it's time to start learning how to design multi-role USB devices. Cypress recommends starting with this book and referring to other components of the kit at the appropriate times. The book builds a series of examples, each one adding simple, incremental capabilities to the design so that you end up with a complex system – but without the development ever appearing complex. At the very least, look through the table of contents now. The first few chapters cover introductions to dual-role devices, the Cypress chips, and the development tools. The next few work through host, dual-role, and OTG applications using stand-alone operation of EZ-OTG and EZ-Host. The final chapter discusses the use of EZ-OTG and EZ-Host in conjunction with a system microprocessor (co-processor mode). Once you start reading you will probably find that you want to work through everything, but there may be pieces that can be delayed until later. As an example, if you know that you are using EZ-OTG or EZ-Host as a stand-alone device (not in conjunction with another system microprocessor), then you may desire to skip the last chapter on co-processor operation for now.

Navigator / Design Examples

After (or while) working with *USB Multi-Role Device Design By Example*, the next place to go is the EZ-OTG / EZ-Host Navigator. The executable is located in the \OTG-Host\Tutorial\ directory. You can launch it directly, from the Start Menu, or from a shortcut if you created one as described in the Installation section. As mentioned in the introduction, the Navigator has multiple functions.

Tutorial

The first function of the Navigator that you will use is the HTML viewer. Since it's a tutorial by design, it doesn't require much explanation here. Launch the application and you will instantly see that you can click on links and images to learn more about all aspects of the chips, the kit, and the OTG concepts – at your own pace and in the direction that you want to go.



Design Examples

At the top of the Navigator are five tabs: Home, OTG, Dual Host, Dual Slave, and Passthrough. Home gets you back to the top level of the tool. Each of the other tabs corresponds to one of the Design Examples included in the kit. The Design Examples are key pieces of the development kit. Each example actually demonstrates a number of functions. A brief background of the levels of hardware and software is worthwhile here, although *USB Multi-Role Device Design By Example* covers it in much more detail.

At the very bottom is the EZ-OTG or EZ-Host hardware. These platforms are both extremely powerful, but using them at the hardware level can be complex. Running in the internal ROM is the BIOS. BIOS has a number of functions that handle most of the low level hardware interaction for moving data through the chips. BIOS's functionality is described in *USB Multi-Role Device Design By Example* and also in the BIOS User's Manual.

On top of that, Cypress has provided a firmware Frameworks. Frameworks uses the BIOS to perform many common functions associated with being a host or a peripheral device. It is also described in *USB Multi-Role Device Design By Example* and in the Frameworks User's Guide. It is a collection of routines that almost every customer will need. Since everyone needs them, it makes sense for Cypress to provide them. It simplifies development as well as support.

Still, the Frameworks doesn't make an application all by itself. On top of that are our Design Examples. These four examples show common uses of the EZ-OTG and EZ-Host chips. Consider them as templates for the start of your own application. Choose the one that most closely matches what you wish to do. Embedded within the four examples are also demonstrations of both stand-alone and co-processor operation, as well as examples of the different co-processor interfaces – Host Port Interface (HPI), Serial Peripheral Interface (SPI), and High Speed Serial (HSS).

Following is the list of design examples in the order they appear in the Navigator, along with a cross-reference to their locations on the CD-ROM. Note that for the co-processor portions of the examples, the StrongARM code is

in the C:\Cypress\USB\OTG-Host\Source\coprocessor directory. There are two subdirectories in this location. “de_app” contains the application code for all of the examples while “linux” contains the linux kernel and drivers.

OTG (DE1)

CY16 code location: C:\Cypress\USB\OTG-Host\Source\stand-alone\de1

This example shows a complete USB On-The-Go device. As such, it is one example that demonstrates a significant amount of functionality. A complete OTG solution requires both host and peripheral functionality, as well as compliance with the OTG protocol. Since you need to have two dual-role devices to interact with one another, both EZ-OTG and EZ-Host are used – one in stand-alone operation and the other in co-processor mode with the StrongARM board. The key things that are not shown in this example are the use of multiple SIEs and USB ports, and the HSS and SPI interfaces.

Dual Host (DE4)

CY16 code location: C:\Cypress\USB\OTG-Host\Source\stand-alone\de4

Dual Host is actually targeted at customers who wish to implement multi-port host devices. This example uses both of the chip’s Serial Interface Engines as a host. It also uses the HPI interface on the EZ-OTG board to function as one of the peripherals. A USB keyboard is supported as the other device, but this must be obtained separately.

Dual Slave (DE2)

CY16 code location: C:\Cypress\USB\OTG-Host\Source\stand-alone\de2

This one sounds unusual – there aren’t many times when you would want to appear as two peripherals at the same time, other than USB networking. Consider it as a basic demonstration of the peripheral mode of the chip – it just happens to show two implementations at the same time because there are two SIEs handy. One SIE will be handled in stand-alone mode – the internal processor will manage the USB data flow. The other SIE will operate in co-processor mode – the data will be passed on to the StrongARM board. In this example we also demonstrate the use of all three of the system microprocessor interfaces: HPI, SPI and HSS.

Passthrough (DE3)

CY16 code location: C:\Cypress\USB\OTG-Host\Source\stand-alone\de3

This final example is really showing simultaneous use of the SIEs in separate host/peripheral modes. One SIE is configured as a host to a downstream peripheral (a USB keyboard again) and the other SIE is configured as a peripheral to a PC. One of the features of this example is that data can be received from the downstream peripheral device and “passed through” to the upstream host.

Control Panel

On the right hand portion of the four Design Example tabs is a small icon resembling a dialog box. This icon launches the Control Panel portion of the tool. The tutorial itself walks you through this. The Control Panels provide you with the means of running and interacting with each of the examples.

CY16 Development Tools

If you intend to work through the simple examples while you are reading *USB Multi-Role Device Design By Example*, then you will get to the tools in Chapter 3. If your intention was to read first and get back to the examples later, then after working with the book and the Navigator, it’s now the appropriate time to investigate the tools.

The CY16 development tools are a port of the GNU development environment developed for Cypress by RedHat. Complete documentation is included in the \OTG-Host\Docs\OTG-Host Tools\ directory. The basic tools include a C compiler, assembler, linker, and debugger, but there are many supporting utilities.

If you have worked through the book, you will note that John Hyde makes use of the tools from a command line environment running the bash shell. Many developers may prefer such an environment. During the CD installation a batch file, BASH_ENV.bat, is placed in the C:\Cypress\USB\OTG-Host\Tools directory. A shortcut is also placed in the start menu under Start → Programs → Cypress → USB → OTG-Host → BASH Environment. This batch file will set up all of the required environment variables as outlined in the CY16.pdf document, and then

invoke a bash shell to begin command line development. If you are developing on a Windows ME or Windows 98SE system then you will have to modify this file to set the initial environment size to 1024.

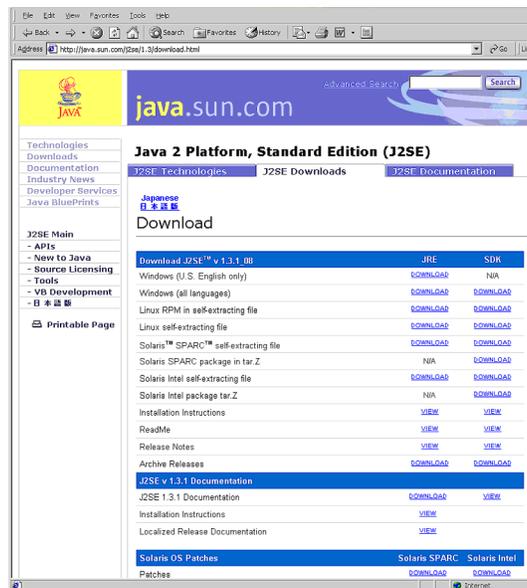
For those developers who may not like the command line environment, the tools also include an Integrated Development Environment – the Eclipse IDE.

Eclipse IDE

This is a powerful tool that we recommend using. Project files are set up for the Navigator design examples to make it easy for you to work with our code within this environment. The tool itself is very similar to other common IDEs. It also has an html based help system that provides complete documentation on the setup and use of the tool. In order to use it, however, you will need a Java Runtime Environment (JRE) installed. The sections below reference specific links to obtain two JREs. If the links are no longer active as a result of future website changes, a quick search of those sites or the web should point you to new locations.

Sun Java

One possible JRE is the Java 2 Runtime Environment, Standard Edition from Sun Microsystems. The current 2.0 release of the IDE is built to work with v1.3 or 1.4 of the Java Runtime Environment. If you don't already have it, you can download it freely from the Sun website. java.sun.com. For the 1.3 version browse to <http://java.sun.com/j2se/1.3/download.html>.



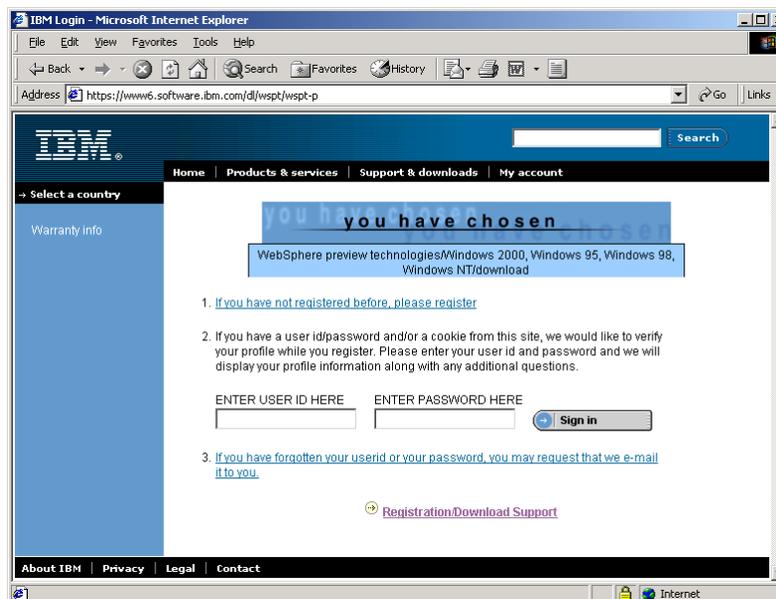
Select the appropriate Windows download, for example the “Windows (U.S. English only)” option. You will be presented with a license agreement – accept to proceed.



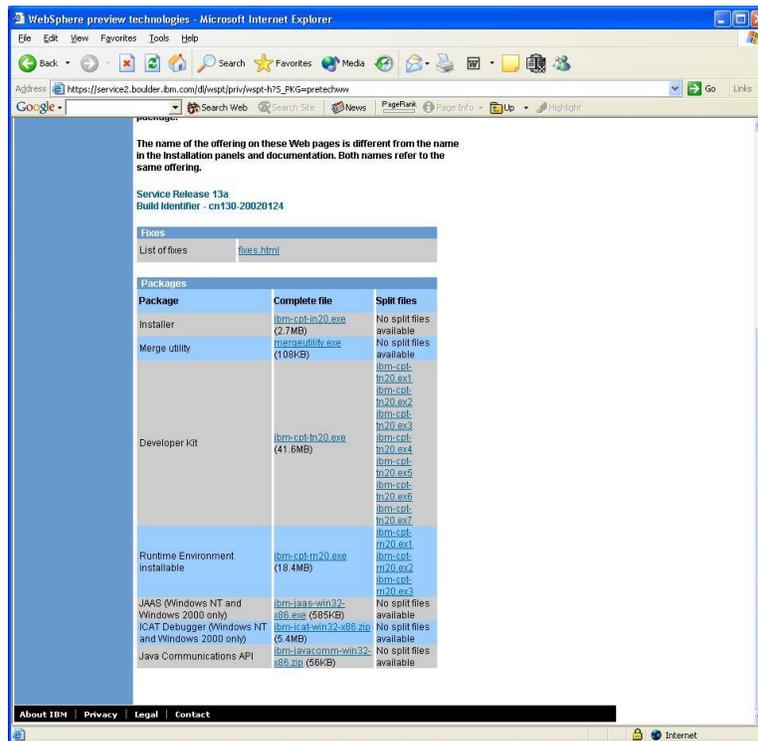
Download the file. After the file is finished downloading, execute it and follow the instructions for installation.

IBM JRE

Alternately you may also download the Java Runtime Environment from IBM. Go to <https://www6.software.ibm.com/dl/wspt/wspt-p>



If you have not registered for this site select the "If you have not registered before, please register" link. Provide the requested information. After signing in, the download files should be available.



Download the “ibm-cpt-in20.exe”, “ibm-cpt-tn20.exe”, and “ibm-cpt-rn20.exe” files. Execute the file “ibm-cpt-in20.exe” to extract the install program. Execute “install.exe”.

Tools Documentation

A set of documents are also provided to describe the operation of the tools. These are all grouped together in the “C:\Cypress\USB\OTG-Host\Docs\OTG-Host Tools” directory. The following list provides a brief description of each document:

1_start.pdf

Provides an overview of the tools, installation instructions, and usage guidelines

2_compile.pdf

Documentation for the compiler and pre-processor

3_debug.pdf

Overview of the debugging tools including gdb and Insight (graphical interface for gdb)

4_libs.pdf

Description of the libraries

5_tools.pdf

Documentation for ld (cy16-elf-ld, the linker), make (build manager), diff, and patch.

6_auxtools.pdf

Overview of the auxiliary tools. Includes documentation for cy16-elf-as (the assembler), the cygwin environment, info (the online documentation for the tools), and the binutils:

- ar (cy16-elf-ar, library manager)
- nm (cy16-elf-nm, lists symbols)
- objcopy (cy16-elf-objcopy, translates executable formats)

- `objdump` (`cy16-elf-objdump`, provides various listings of the contents of executable images – disassembly, section layout, etc.)
- `ranlib` (`cy16-elf-ranlib`, generates indexes of archives (libraries), etc.)
- `size` (`cy16-elf-size`, lists section sizes for object or archive files)
- `strings` (`cy16-elf-strings`, displays ASCII strings from an object file)
- `strip` (`cy16-elf-strip`, removes symbols from object files)
- `addr2line` (`cy16-elf-addr2line`, translates program addresses into filenames and line numbers)
- `readelf` (`cy16-elf-readelf`, displays information about ELF format object files).

7_snavigator.pdf

Documentation for Source Navigator (sn). Note: this is not related to our EZ-OTG / EZ-Host Navigator tool described earlier.

cdk_guide.pdf

Component developer's guide for SID, the simulation framework

cy16.pdf

CY16 specific Toolkit User's Guide – a "Quick Start Guide" for use of the tools with the Cypress parts

quickref.pdf

Quick reference guide for the GnuPro toolkit

sid-guide.pdf

User's Guide for SID including SID architecture, SID configuration file, and SID usage

USB Software Stack Support

If your intention is to use the EZ-OTG or EZ-Host as a stand-alone device, then you should have enough information from the above sections to begin development. If you are going to be using the chips in a co-processor mode, then you may still have questions about software support on your system processor. The system processor is the master of the EZ-OTG or EZ-Host device. This processor will typically run some level of USB software stack to enable the USB capabilities of the system. This may include a host stack, a device (peripheral) stack, and/or an OTG control layer.

A Linux solution is provided with this kit. Linux was chosen to maintain accessibility – it's readily available, it's open source so that code can be provided and re-used, many developers are familiar with it, and there's no license fee or outside cost to the kit to be able to use the examples. Complete details of this solution are provided in the Linux USB Driver User's Guide.

We also recognize that there are many different reasons for choosing an operating system in an actual application, and Linux may not meet your needs. With that in mind, Cypress is continually working on support for additional operating systems and class drivers. This is likely to be one of the most dynamically evolving areas after release of the development kit, so please check the Cypress website or contact your Sales Representative or Field Applications Engineer to obtain the latest information.

Compliance Testing

Although most developers are fairly comfortable with the challenges of the hardware and software design process, there is one aspect that is often neglected until late in the design process: compliance testing. Compliance testing may not be important to everyone. If you are using USB to connect different functions within your own closed system then it may not be an issue. For devices that will be sold on the consumer market, it is often critical.

First of all, start planning for it early on. There are compliance workshops (“PlugFests”) run by the USB Implementer’s Forum approximately once per quarter, but they can fill up quickly. Check the USB-IF website (www.usb.org) for the date of upcoming compliance workshops and reserve a spot early. Also don’t forget that there is a checklist (available on the web) that must be filled out ahead of time and brought with you. This is also a common oversight. The alternative to PlugFest is to use an approved test house to obtain your certification, but there will of course be a cost for this.

It is important to understand the different types of USB compliance testing that exist, and how the design of an embedded host device is impacted. To make certain that you have the most up to date information it is highly recommended that you check the USB-IF website for current compliance policies.

The most common type of certification is for USB peripheral devices. The important thing to note here is that all OTG devices must pass USB peripheral compliance. There is also USB host testing, however the formal test requirements are intended for PC based systems. There may be some embedded host devices that have the capability to pass the complete set of tests, but many will not. For embedded host devices, the only current compliance program is for OTG devices. As mentioned above, OTG devices must pass all of the peripheral tests, but they also have an additional set of test requirements to verify compliance with the OTG Supplement.

Why is this important? When you are designing your device, make sure that you understand what you can possibly test for if compliance is important to you. For example, devices that use the mini-AB receptacle are expected to pass OTG compliance. On the one hand that means that they can’t just be a peripheral to a PC and a host to their desired peripherals. They actually have to implement the OTG protocol and support dual-role devices. On the other hand, that also means that they can currently only have one port. The bottom line is check for the latest guidance so that you aren’t surprised at the last minute.

Support

Hopefully this document has been successful in giving you the guidance that you need to begin work on your design. The last thing to cover is how to get help when you need it. Almost everything that you need should be included in the materials in this kit. If you can’t find it, or if the information you are looking for just isn’t here, then it’s time to get help. There are a few places to go, depending on your preferences or the type of help that you need.

First of all, check the web for updates. Although the complete CD-ROM image for this kit is much too large for a web download, there will be some materials available on-line. We have a section for Development Kits that will probably contain general kit items, but there are also sections for Application Notes, Software Downloads, and other types of content. All of these types of content should be accessible by starting under the Product Family of USB Embedded Hosts.

Next is the Cypress Knowledge Base. This is an on-line system that has a collection of frequently asked questions and answers. Look for it under www.cypress.com/support and search for your topic.

If you don’t find what you need in the Knowledge Base, then you can submit a question on-line through our Case Management System. You can also access this system from www.cypress.com/support. Enter your question and our Applications Engineering team will get back to you with a response. They usually follow up within 24 hours, although some cases may take longer depending upon the nature of the question.

In addition, it is possible to reach Applications Engineering by phone. Dial (800)-541-4736 or (408)-943-2600 and enter 8 for support.

Also please keep your Cypress Sales Representative and Field Applications Engineer informed. Your local FAE may be able to provide on-site assistance to help you resolve your problem. If nothing else, they can serve as liaison to the internal factory resources to manage requests for assistance.