UM10017-01
WASABI-Hot!

A Demo Program for the ISP1362

User's Guide
Rev: 1.0

05-Aug-2002: application (code) version 1.0d

OKANO, Akifumi
HIGASHIYAMA, Ken
HARASHIMA, Hisashige
+81-3-3740-4744
wired.support@philips.com
Computing Segment/BL-Connectivity
Philips Semiconductors @Tokyo, Philips Japan, Ltd.
http://www.semiconductors.philips.com/buses/usb

Revision History:

Rev.	Date	Descriptions	Author
1.0	September 2002	First draft.	Okano/Higashiyama

We welcome your feedback. Send it to wired.support@philips.com.

Philips Semiconductors - Asia Product Innovation Centre
Visit www.semiconductors.philips.com/buses/usb

PHILIPS

This is a legal agreement between you (either an individual or an entity) and Philips Semiconductors. By accepting this product, you indicate your agreement to the disclaimer specified as follows:

DISCLAIMER

PRODUCT IS DEEMED ACCEPTED BY RECIPIENT. THE PRODUCT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, PHILIPS SEMICONDUCTORS FURTHER DISCLAIMS ALL WARRANTIES, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANT ABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. THE ENTIRE RISK ARISING OUT OF THE USE OR PERFORMANCE OF THE PRODUCT AND DOCUMENTATION REMAINS WITH THE RECIPIENT. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL PHILIPS SEMICONDUCTORS OR ITS SUPPLIERS BE LIABLE FOR ANY CONSEQUENTIAL, INCIDENTAL, DIRECT, INDIRECT, SPECIAL, PUNITIVE, OR OTHER DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR OTHER PECUNIARY LOSS) ARISING OUT OF THIS AGREEMENT OR THE USE OF OR INABILITY TO USE THE PRODUCT, EVEN IF PHILIPS SEMICONDUCTORS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CONTENTS

FIGURES

TABLES

1. WASABI Hot!

1.1.   Introduction

"WASABI-Hot!" is demonstration software for the Philips ISP1362. The ISP1362 is a single-chip Universal Serial Bus (USB) On-The-Go (OTG) Controller integrated with the advanced Philips Slave Host Controller (HC) and the Philips ISP1181B. WASABI-Hot! demonstrates the capabilities of the ISP1362 with another OTG host or peripheral, and with USB hubs, speakers, printers and HIDs.

WASABI-Hot! can be used to demonstrate the following:

- Play audio data (linear PCM) stored in the current or other OTG host or peripheral

- Print raster image

- Capability and communication of OTG Session Request Protocol (SRP) and Host Negotiation Protocol (HNP)

- Hubs operation

- HIDs operation

- Enumeration for any device.

In addition, WASABI-Hot! can be used as a reference code to build OTG application firmware. WASABI-Hot! is distributed with complete source code files. For more details on the internal mechanism of WASABI-Hot!, please refer to these files.

You may port the code to other platforms easily with our without the OS. This is because the basic parts of WASABI-Hot! (Host Controller (HC) unit, Device Controller (DC) unit and On-The-Go (OTG) unit) do not use any OS features.

1.2.   Working

The WASABI-Hot! program runs on IBM-compatible PC running Microsoft® Windows® 95, Windows 98 or MS-DOS® operating systems that allow the DOS program to directly access the hardware. For more information, refer to the *ISP1362 PCI Evaluation Board User's Guide,* and the *ISP1362 ISA Evaluation Board User's Guide*.

WASABI-Hot! runs from a single executable file. No drivers are required to run this program. WASABI-Hot! consists of these units as can be seen in Figure 1-1:
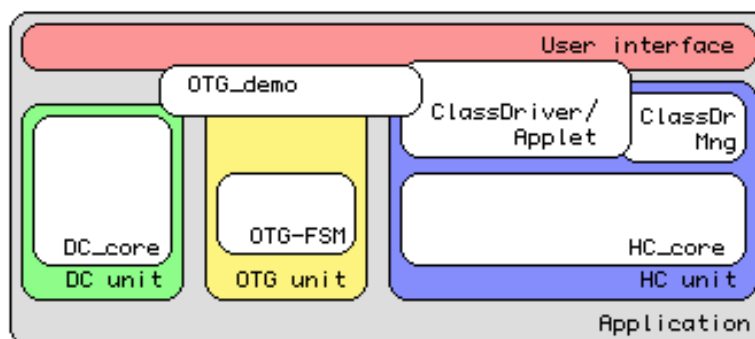
- HC unit
- DC unit
- OTG unit.

Figure 1-1: Software Functional Blocks

The HC unit processes the Host Controller on the ISP1362. The basic design of and the structure of the modules in the HC unit are based on Yebisu, which is a demonstration program for the ISP116x. The HC unit, as defined by the USB spec, supports all USB transfer types at the same time. ClassDriver/Applet is a part of this unit. An applet is a set of drivers and user interfaces. The applet drives USB speakers and printers. The class driver drives HIDs and hubs.

The DC unit handles all the Device Controller processing. The DC unit supports standard requests defined in Chapter 9 of the USB specification to complete the enumeration process. Descriptors for Device, Configuration, Interface, Endpoint, String and OTG are supported. OTG feature selector required for HNP is also available. Vendor-specific requests are defined for communicating with another OTG host or peripheral.

The OTG unit has been ported from OTGC (version 1.0). OTGC is ISP1362 OTG-compliant reference software. This unit handles the SRP and the HNP protocols, and includes some sample codes to demonstrate the OTG feature.

1.3. Distribution

WASABI-Hot! is distributed as a part of the ISP1362 PCI/DOS OTG Mini Kit (ISP1362 evaluation kit for the MS-DOS platform). It includes executable binary code with a complete set of the source code. You may freely distribute and modify the code but at your own risk.

1.4. Version

Application version: 1.0d.

A known issue in WASABI-Hot! application version 1.0d is:

- **D13_test Application Cannot Drive WASABI-Hot!:** The D13_test application version 1.2 is a test program on the PC to evaluate the ISP1181x, and the Device Controller portion of the ISP1161x and the ISP1362. The D13_test application cannot handle the OTG descriptor that the WASABI-Hot! returns as part of the configuration descriptor , causing application error.

2. Setting Up the Environment

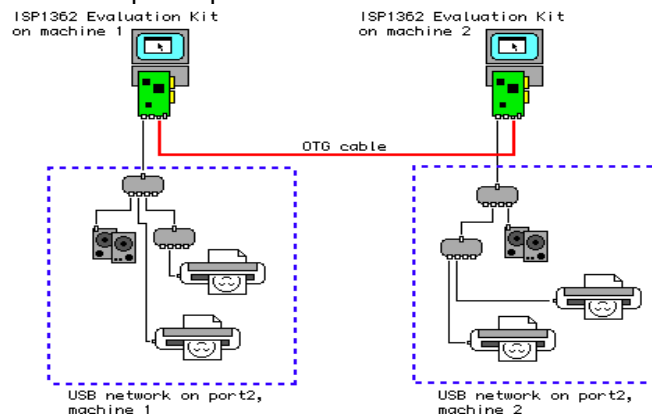Figure 2-1 shows the OTG demo setup example for "WASABI-Hot!."

Figure 2-1: WASABI-Hot! OTG Demo Setup Example

2.1.   Hardware Setup

2.1.1. ISP1362 Evaluation Kit

The mini kit comes with a PCI board; see Figure 2-2.

Figure 2-2: ISP1362 PCI Evaluation Board

2.1.2. PC and OS

Prepare one PC for one evaluation hardware. WASABI-Hot! does not support operation of two or more boards on one PC at the same time. Therefore, if you want to try the demo with two WASABI-Hot! programs simultaneously running, you need to have two PCs as connecting OTG devices.

Intel® Pentium-II 350 MHz CPU and above, or equivalent are recommended for good quality demo. Since WASABI-Hot! is an MS-DOS application (has direct access to the hardware), the OS for the PC must be 16-bit Windows (Windows 95, Windows 98 or Windows Me) or pure MS-DOS.

You must not install any driver software for the board. If you are using the PCI board, Windows may prompt you to install driver at the first system boot. Follow the Windows message steps. For detailed information, refer to the *ISP1362 Evaluation Board User's Guide*.

2.1.3. USB Peripheral Devices

The WASABI-Hot! demo program needs USB peripheral devices to show driving capabilities of the USB host. The default enumeration process can enumerate any device.

WASABI-Hot! contains drivers for USB speakers, printer, hub and HID in the application code. These drivers can drive basic functions of devices. For speakers and printer, WASABI-Hot! emulates driving operations.

WASABI-Hot! enables two OTG devices to communicate with each other. Therefore, WASABI-Hot! running on one PC can drive WASABI-Hot! running on another PC using OTG protocols. These connected OTG devices communicate using vendor-specific commands.

2.1.3.1. USB Hubs

You can use USB hubs to expand the USB port; see Figure 2-3.

Figure 2-3: Ports on the PCI Board

Port 1 on the evaluation board is the OTG port. If you do not use the OTG feature, you can connect a hub or a non-OTG device to port 1 with mini-A plug to standard-B plug cable. To start operation, press the **z**[1] key. The z key command is the "Bus Request" command. It will start to feed V_{BUS} on port 1.

Port 2 on the evaluation board has standard-A receptacle. This is a USB Root Hub downstream port that can be operated by just connecting devices or hubs to it.

Note: If you are using the internal charge pump as the source for V_{BUS}, hubs and devices on the OTG port may need to be self-powered. If hubs or devices are bus-powered and consumes large current, it will exceed the limited capability of V_{BUS} on the OTG port.

Note: WASABI-Hot! has limitations regarding the number of hubs and HIDs in the USB tree. Hubs and HIDs use interrupt transfer to poll the state of devices. WASABI-Hot! does not have flexibility to adjust the number of interrupt transfers. Version1.0 can support seven interrupt transfer pipes in the USB tree. Therefore, the total number of hubs and HIDs in WASABI-Hot! is limited to seven.

2.1.3.2. USB Speakers

You need USB speakers or headphones for the audio output. It converts the USB data to air vibration for your ears. The audio has been tested using **Roland MA-150U** and **Philips 350/00** (see Figure 2-4**)**. Both these systems use the Philips USB audio codec chip (UDA1321).

Figure 2-4: USB Speakers (Philips 350/00)

WASABI-Hot! 1.0d and later versions support devices. You can drive USB audio devices based on the Philips UDA1325 based devices. These devices are widely used in the USB enabled audio components, stereo sets and USB headsets.
WASABI-Hot! does not support HID features for audio devices. (It only supports independent HID devices, such as keyboard and mouse). Some audio control features on the front panel of the speakers are not supported.

The volume level can be controlled on the host side interface by using a keyboard on the host machine or supported HID devices.

---

1 In this document, items that you type or click are indicated in **bold**.

2.1.3.3. Printers

The operation test has been done using the following printers:

- HP® 895Cxi

- Epson® CL-700

- Canon® BJ-F850.

These printers work fine for raster image printing. You may use an equivalent printer. Among these printers, the HP printer is the only printer that works for text data printing. If you want to print a text file, choose a text file in the print command prompt instead of the raster image command file. You must add a file suffix for the text file. This is because WASABI-Hot! adds a suffix to the file name if it does not have one. WASABI-Hot! automatically selects the suffix name by detecting the Vendor ID code that can be obtained in the USB enumeration process.

Figure 2-5: USB Printer (HP 895Cxi)

2.1.3.4. Human Interface Devices (HIDs)

The keyboard and the mouse are supported as HIDs. A keyboard on USB can be used for input of commands to WASABI-Hot!. WASABI-Hot! accepts key inputs from both USB keyboards on the ISP1362 port and keyboards on the PC. To enter the keyboard test mode, press the **k** key. In this mode, you can test the keyboard by typing characters that will be displayed on the screen. To abort the keyboard test mode, press the **Enter** key.

The mouse movement and mouse button events will be shown as cursor movements and its color changes on the screen, see Section 4.4 for more details. You can also use the mouse to control the volume level of USB speakers. To control the volume level, drag using the right mouse button.

Griffin Technologies PowerMate (USB-volume device) (see Figure 2-6) is a special device for WASABI-Hot!. With this device, you can control the audio volume level (rotate knob) and mute (click). It is a mouse device on USB. WASABI-Hot! detects it by using the Vendor ID and Product ID in the standard device descriptor. These controls are done at the application level by using the HID class driver API.
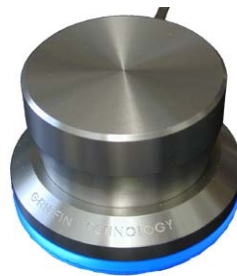

Figure 2-6: Griffin Technologies PowerMate

2.1.3.5.OTG Devices

An OTG device can be connected to the OTG port. WASABI-Hot! can process Session Request Protocol (SRP) and Host Negotiation Protocol (HNP).

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **10 of 37**

WASABI-Hot! allows the Host function to be transferred between two directly connected dual-role devices that is operated by WASABI-Hot! and OTGC ver. 1.0 (see Figure 2-7). OTGC is ISP1362 OTG compliant reference software. OTGC is delivered as a software tool in the ISP1362 PCI/DOS OTG Mini Kit.

Figure 2-7: OTGC Version 1.0

2.1.3.6.Other Devices

Other devices can be used to show that it can be enumerated.

2.2. Software Setup

2.2.1. Installing WASABI-Hot!

Copy WASABI-H.exe to any directory. Some preparation may be needed, however, for specific demonstration using USB devices, except hubs and HIDs.

In the distribution package, the WASABI-H.exe file is stored in the "obj" directory.

2.2.2. Preparing the Audio Data

You may need to prepare the audio file for the USB audio demo. WASABI-Hot! can play the contents of the audio file. You can also play audio by using the beep sound-pressing keyboard.

The audio file should be in the WAVE format (*.wav). You can create it by using the CD ripping software. You can find this software on the Internet or a CD-R writing software package. The file must have data in the same format as an audio CD. Using the ripping software, save the file in the ".wav" format. The internal format must be in linear-PCM, 16-bit/sample, 2-channel stereo, 44.1 kHz sampling frequency. WASABI-Hot! does not process the audio data (except volume control). The audio data flows down to the target endpoint in the USB audio device through the isochronous pipe.

If you want to make an audio file as the default file, name it "default.wav" and put it in the working directory. The default file is used if you are not using any custom setting options for audio files when launching WASABI-Hot!.

The file name and the directory name must be in the classic DOS format. You can use only eight characters for the file name and three characters for the file extension.

To get details on setting music files for WASABI-Hot!, refer to the description of the "-a" and "-l" option switches in Section 3.2 "Command-Option Switches."

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **11 of 37**

2.2.3.   Preparing the Printer Data

To demonstrate the printer operation, prepare the printer data as a file. WASABI-Hot! cannot handle any graphic format for printing because it does not have any data translator to translate graphic format to printer commands.

WASABI-Hot! merely transfers data from the file to an endpoint of the printer. Therefore, the contents of the file must be in the printer language data format. For example, in the case of the HP 895Cxi printer, the printer language is HP-PCL. The printing source file for HP printers must have HP-PCL commands. In this way, you must convert the image source to the command data. You can do this by using the printer driver on the Windows operating system as follows:

1.   Open the picture file with any graphics software in Windows.

2.   Choose **Print** from the **File** menu.

3.   In the printer driver dialog box, select the check box that will enable you to print the picture to a file (a printer image file).

4.   Save the file. Use the classic DOS format name.

3.   Starting the Program
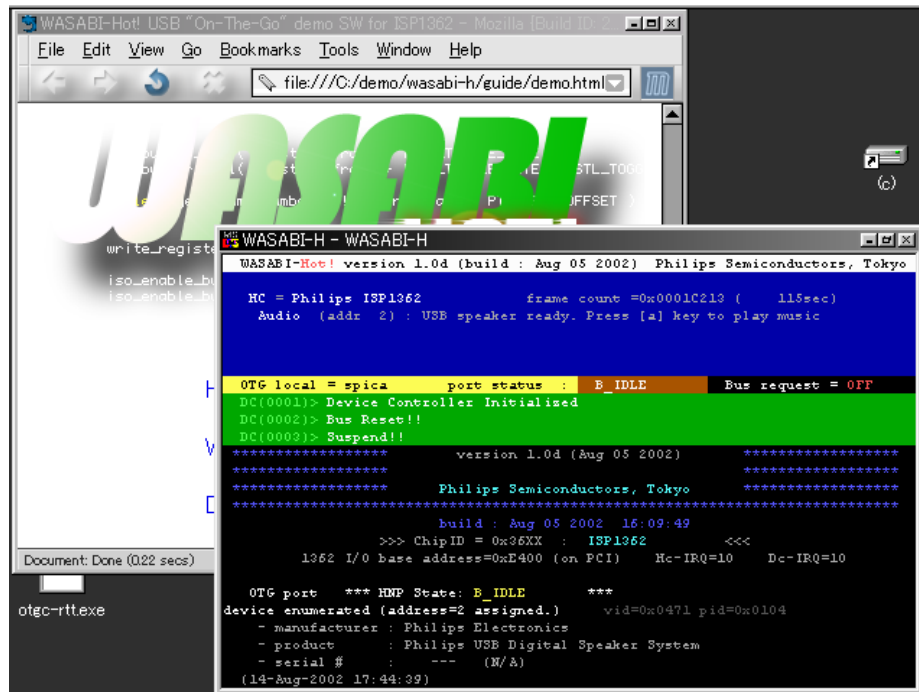
Figure 3-1: WASABI-Hot! in the Windows Environment Running in the DOS Box

3.1.   Launching

Launch WASABI-Hot! in either the Windows or DOS environment. To launch WASABI-Hot! in the Windows environment, double-click the **WASABI-H.exe** icon. To launch it in the DOS environment, type **WASABI-H** at the command prompt and then press the **Enter** key.

Use the "-D-" option switch to run WASABI-Hot! in the DOS-box on the Windows screen. This option will set the screen color that can be used in the Windows environment and disable functions that require fast screen updates.

Note: The refresh rate of the DOS-box window is slower than the full-screen DOS mode.

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **12 of 37**

WASABI-Hot! enables you to perform custom settings using two methods:

• Using a file

• Using a command-option switch as argument when launching WASABI-Hot!

You can use either or both the methods to perform custom settings. If you have set the same option switches using both the methods, the settings that are done using the command-option switch overrides the settings done in the file.

The "setting.wsb" file located in the working directory can be used to perform settings using a file. If you do not have this file, WASABI-Hot! just ignores this process. The content of the "setting.wsb" file is just an array of option switches similar to the command arguments. In this file, space characters (space, tab, and control codes for carriage return (CR) and line feed (LF)) and phrases enclosed within parentheses will be ignored. To get more information, see sample "setting.wsb". The "setting.wsb" file is a text file that can be opened by using a text editor.

3.2. Command-Option Switches

To set program parameters, you can use command-option switches. These setting are loaded when launching the program. These setting can be defined in the "setting.wsb" file or using command-option switch as argument of WASABI-Hot! in the DOS environment.

<Detailed information can be found in the source code "src/main.c".>

Note 1: The switches with lowercase character take a parameter. The switches with uppercase character do not take any parameter. The switches with uppercase character have been developed to enable you to choose alternative internal default settings.

Note 2: The switches with uppercase character do not take a parameter. However, you can use the minus character (-) just after the switch identifier to disable that feature. For example, you can disable the DOS full-screen mode, by typing a minus after D (that is, -D-). This feature can also be used to turn-off a feature (using the command line argument) that has been enabled in the "setting.wsb" file.

3.2.1. Environmental Option Switches

Command-Option Switch	Description
-c SSSS	**Charge Pump Selector**: For the OTG feature, the V_{BUS} source can be selected. You can set it to internal or external. You can type a string following the option switch. WASABI-Hot! checks only the first character of the string. If the first character is "e" (lowercase) or "E" (uppercase), WASABI-Hot! uses the external charge pump for the OTG V_{BUS} source. Otherwise, it uses the internal charge pump. Default setting is "i" (internal)
-i SSSS	**Peer Name (ID)**: Defines peer name (ID) for the OTG feature. This ID will be used in the OTG feature demo. You can use different name for each peer. The string gets truncated to 10 characters, if you specify a name that is more than 10 characters.
-D	**DOS Mode (Operates in the Full-Screen DOS Environment)**: Use this option when you run WASABI-Hot! in the full-screen DOS mode. This adjusts the display color and disables character blinking. Also, it disables the file access in Interrupt Service Routine (ISR) and enables the peak-meter feature when

playing the audio file. By default, this mode is enabled. Therefore, if you wish to disable this mode, use the "-" (minus) character after the "-D" option switch when launching WASABI-Hot!.

-F **Fast Enumeration Mode**: In the default setting, WASABI-Hot! inserts pauses to show message for each device enumeration completion. This option has been developed to disable these pauses.

-S **Silent Enumeration Mode**: In the default setting, if USB speakers are connected and are enabled, WASABI-Hot! produces a beep sound at each enumeration completion. Set this option to disable the beep sound.

-V **Display Chip Version**: Enables you to display the chip version on the screen.

-I **Skip PCI Board Detects**: In the normal sequence, WASABI-Hot! tries to detect the PCI board first. If it fails, it tries to detect the ISA board with base address setting. Use this option switch to disable the PCI board detection process.

-W **Enable Warning Message**: Use this option switch to display warning messages for failed enumerations.

3.2.2. Audio-Related Option Switches

-v DD **Volume Level**: Modifies the default volume level for the audio play. WASABI-Hot! can control the audio volume level. The default value for this switch is set to a maximum level of "0 dB". This switch can be used to modify the volume level after the program has been launched. Decimal values follow this switch. The value can range from 0 to 60 dB. The volume level is set in 2 dB steps. Therefore, if you specify "-v 24" as the value for this switch, the audio volume level will be set to "-24 dB". However, if you specify an odd number as the value for this switch, it gets rounded to an even number. For example, if you set the value to "-v 31", the value gets rounded to "-30 dB."

- a FFFFF **Default Audio File**: (This option is not necessary if you use the "-l" option switch.). It defines the audio (music) file to play. The audio file path and the file name are defined by using this option switch. The file path must be written in the UNIX path format, that is, with path delimiter (/). A linear-PCM coded ".wav" file or raw data audio file (no ".wav" header data file) can be used.

- l FFFFF **Audio List File**: Defines the audio list file path and the file name. You can define audio files by using the "music list file". The "music list file" is a text file that contains a list of audio file paths and file names. Each item is placed in a new line. The list displays as a menu when you start to play audio enabling you to select a file to play. The path must be written in the UNIX path format, that is, with path delimiters. In the music list file, you can add comments on WAVE file items. A comment must have the "#" or "!" character in the line head.

WASABI-Hot loads the "music list file" during the program launch. To reload the file, you need to relaunch WASABI-Hot!.

As an option, you can use the "#%" or "!%" identifier in the line head to provide information on the contents of the file. When you start to play audio, the strings that are defined after "#%" or "!%" will be shown in the menu providing information on the file contents. File information can be defined after an audio file definition. The information line must follow the WAVE file path definition.

```
# Music list file '#' and '!' character in the
# line head can be used as comment line marker.
! Also the empty line will be ignored.
../music/allnight.wav
# Next line is the file information line
#% File information sample this comment

../music/allnight.wav
../music/morethan.wav
../music/dig_it.wav
e:/k_debug/testwave/sin.wav
#% Test signal sin wave
e:/k_debug/testwave/sq.wav
#% Test signal square wave
e:/k_debug/testwave/ramp.wav
e:/k_debug/testwave/ramp2.wav
../music/i_will.wav
../music/yozora.wav
```

Figure 3-2: Sample Music List File

3.2.3. Printer-Related Option Switches

-p FFFFF   **Print File Path**: Defines the default path of the print file. You can set a default path for print image files. The printer operation (key) command (see Section 4.3) prompts you to enter the file path and the file name for printing images. The "-p" option switch enables you to set the default path from the DOS command line.

3.2.4. Usage of Option Switches

Here are some sample scenarios:

The following expression sets the base address of the ISP1362 to 300H, and the IRQ number of the Host Controller and the Device Controller in the case of an ISA board to 3 and 5, respectively. The audio file is placed in the "../music" directory and the name of the file is "dig_it.wav".

WASABI-H -a ../music/dig_it.wav

The following expression places the music list file in the working directory and is named "song.lst". The default path of the print file is set to "c:/print/image/". The volume attenuation level is −32 dB, and the OTG peer name is "banana". The fast and silent enumeration modes are enabled. The DOS mode to launch WASABI-Hot! in the DOS box is turned off. This command does not have the base address and IRQ settings. The target could be a PCI board.

WASABI-H -l song.lst -p c:/print/image/ -v 32 -i banana -F –S –D-

These switches can be used in the DOS environment. However, if you are launching WASABI-Hot! in the Windows environment by double clicking on it, you will need to write a batch file or create a shortcut file to the application and edit its properties.

Rebuilding the source code might change the native default values. You can have more options by editing-rebuilding the code. For more information, refer to the WASABI-Hot! source code.

For more information on command-option switches, refer to src/main.c.

4. Running WASABI-Hot!

This section describes how to run the WASABI-Hot! demonstration.

4.1. Plugging In and Plugging Out USB Devices

Figure 4-1: Hub

After WASABI-Hot! is launched and running, you can connect devices to port 1 (OTG port) and port 2 (standard-A receptacle port). WASABI-Hot! is emulating a USB device.

In the idle state, the OTG port will not provide V_{BUS} on the port. You need to turn on V_{BUS} first by generating a "bus request". The bus request has two states: ON and OFF. Press the **z** key to toggle between these two states.

When the device enumeration is complete, string device descriptors appear on the screen with assigned USB device addresses. The USB device address of the first enumerated device is 2. The address of the second enumerated device is 3. If two or more devices are connected on the ports at the same time, enumeration will start from the Root Hub side port. If multiple devices are connected to downstream ports on the same hub, the device that is attached to the lowest numbered port will be enumerated first.

WASABI-Hot! allows you to connect up to 126 devices. This is because WASABI-Hot! reserves USB address 1 for internal representation of the Root Hub (that is, to drive the Root Hub as one of the device hubs for the internal mechanism). USB addresses from 2 to 127 are available for USB devices.

Addresses are recycled. At enumeration, the smallest available address number is used.

Status of USB speakers and printers is shown in the blue area of the screen.

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **16 of 37**

Figure 4-2: Screen Design

4.2.   Playing Audio

Figure 4-3: Playing Audio

Using the audio play feature in WASABI-Hot!, you can produce the beep sound or play any audio (music) file. The data source of the beep sound is an internally calculated waveform. The audio (music) file is a WAVE file data stored on the hard disk drive.

The beep sound is produced by WASABI-Hot! when there is an internal event or the user enters a command. When WASABI-Hot! runs in the non-silent mode (without using the "-S" option switch), the beep sound will be generated when enumeration is complete. A manual beep does not affect the silent mode settings.

Keys comma (**,**) and period (**.**) can be used to increase or decrease the volume level.

4.2.1.   Producing the Beep Sound

Connect USB audio speakers and then press the **b** key on the keyboard. The **b** key is a command to produce a beep. This command can be use for checking the logical connection to USB speakers.

The **Z**, **X**, **C**, **V**, **B**, **N**, **M** and **<** keys (these key must be pressed in combination with the Shift key) are also for producing the beep sound. These keys are assigned to tonal scale (no chromatic scale is available). It may be possible to play simple music using this feature.

4.2.2. Playing an Audio File

To play an audio file, press the **a** key. If you have multiple audio files defined in the "music list file" using the "-l" command-option switch, WASABI-Hot! will display the list of available files, see Figure 4-4. It starts to play audio (music) after you have selected the file number.

After any audio play command, WASABI-Hot! searches for a USB audio device on USB. An audio device is identified by the interface class code (0x01) that can be found in the interface descriptor. If there are multiple audio devices, WASABI-Hot! will choose the device with the smallest USB address.

Figure 4-4: Audio List Displayed after the Play Command

While the audio file is playing, its status will be shown at the audio status line in the blue area of the screen, see Figure 4-5.

Figure 4-5: Play Started

4.3.   Printing Pictures

Figure 4-6: Printing Pictures

To start printing, press the **p** key on the keyboard. WASABI-Hot! searches for USB printer devices on USB. A printer device is identified by the interface class code (0x07) that can be found in the interface descriptor. If there are two printer devices, WASABI-Hot! will choose both the devices for operation.

WASABI-Hot! prompts you to point to a printer image file. (If there are two printers, WASABI-Hot! will prompt you to choose a printer as the printing target before this step.)

Type the full path name, and then press the **Enter** key (see Figure 4-7).

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **19 of 37**

Figure 4-7: Type the File to Print

The printing process can be canceled by pressing the **p** key.

Multiple printers can run at the same time, see Figure 4-8. The ATL scheduler in WASABI-Hot! supports simultaneous transfers.

Figure 4-8: Simultaneously Running Two Printers

UM10017-01

User's Guide   **Rev. 1.0—September 2002**   **20 of 37**

4.4. HID Operations

Figure 4-9: HID Operations

WASABI-Hot! can operate keyboard and mouse devices. The keyboard just functions as a normal keyboard to operate the application. The mouse just displays information about that state on the screen.

To enter the keyboard test mode, press the **k** key (see Figure 4-10). Once the **k** key is pressed; the key input will be echoed to the screen. Press the **Return** or **Enter** key to exit the keyboard test mode.

Figure 4-10: Testing the Keyboard

To operate a mouse, connect the mouse (see Figure 4-11). Move, click and drag. Dragging the mouse with the right button held down enables you to control the volume when USB speakers are connected.

Figure 4-11: Testing the Mouse

WASABI-Hot! supports Griffin Technologies PowerMate (USB volume knob) to control its audio features. It is a sample to implement custom class drivers for vendor-specific devices. This device can be operated as a mute and volume control device. Rotate the knob to the left or right to decrease or increase the volume, respectively. Click (to perform the click action, press the knob) to switch the mute state ON and OFF.

4.5.   Playing with OTG

The features described in the earlier sections demonstrate host capabilities. The features in the following sections demonstrate OTG features.

4.5.1. Bus Request
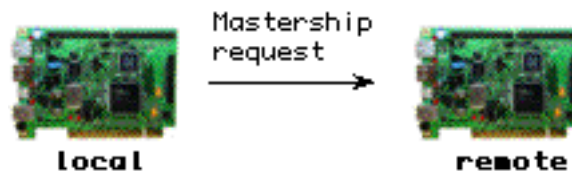
Figure 4-12: Bus Request

The **z** key command is for the OTG feature. It toggles the "bus request" state between ON and OFF. With this action, the internal OTG Finite State Machine (OTG-FSM) will start to switch to the next state.

Figure 4-13: Bus Requested and Obtained Mastership

Changing the bus request state from OFF to ON is an action to acquire the role of host. The "OTG A-device" (which means the OTG dual-role device with mini-A plug connected) will start to provide V_{BUS} to the OTG port and try to acquire the "A-host" state.

The "OTG B-device" will generate Session Request Protocol (SRP) as the first step to acquire the role of host from the A-device.

Swapping of host roles will not occur if another device is already acting as host during the period the session is established. The device merely maintains the states of the bus request. It starts to get the bus after the session is free. Changing the bus request from ON to OFF allows control of the connection to be transferred. This mechanism will work with any OTG device.

4.5.2. V_{BUS} Drop

Key command **/** (slash) is for V_{BUS} drop. It forces a session to be cleared by dropping the V_{BUS} voltage. This command clears the unexpected error state of OTG.

This mechanism will work with any OTG device.

4.5.3. OTG Demo Applications

Some additional OTG demo features are available to show the host and device capabilities other than just transferring control of the host role. The commands described in this section are used for communicating with another WASABI-Hot! device.

These demos can start from any bus request state (OFF or ON). If you start it from the OFF state, it will return to the OFF state after the demo is completed. If you start from the ON state, it returns to the ON state.

<All these features are implemented in "src/_otg/otg_demo.c">

4.5.3.1. Receiving String Messages

Figure 4-14: Receiving String Messages

The other OTG device introduces itself by using the vendor-specific command. The other OTG device returns its name given by the "-i' command-option switch as data of control-IN transfers.

Press key **7** to execute the "Receiving String Message" feature.

Figure 4-15: String Received from Remote

4.5.3.2. Transferring Dummy Data

Figure 4-16: Transferring Dummy Data

This feature will show synchronized bar-graph activities on both OTG devices. The bar graph shows the progress of the dummy data transfer.

Control transfer using vendor-specific command retrieves data. Press key **8** to execute this feature.

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **24 of 37**

Figure 4-17: Receiving Data from the Peripheral Device

Figure 4-18: Serving Data to the Host Device

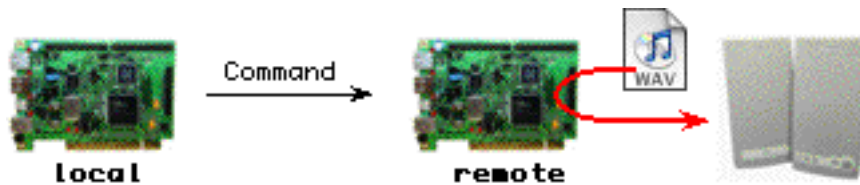4.5.3.3. Playing Music on the Other OTG Device

Figure 4-19: Playing Music on the Remote Peer

To play audio on the other OTG device by using the remote control feature, press key **9**. On pressing the key, a list of audio files located on the other USB device will be displayed. Select a file to play. The other USB device will start to play the selected audio file.

Press key **9** again to stop playing the audio file located on the other OTG device. This function is similar to the **a** key press when playing audio files located on the current OTG device (see Section 4.2.2).

The control transfer performs all transfers.

4.5.3.4. Updating Application. Copying from the Other OTG device

This command might be beneficial for software developers. The WASABI-Hot! executable file can be updated through the OTG cable. The program overwrites the application executable file "WASABI-H.exe" located on the current OTG device with the file data located on the other OTG device.

Press the **u** key and follow the command. All data transfers are done by control-IN transfer data size 64 bytes. It can demonstrate the high transfer speed of the ISP1362's control transfer (This can be varied by the performance of the PC. For example, a Pentium III 550 MHz CPU PC takes 1.5 s to transfer170 kbytes of data).

Figure 4-20: Updating Local Executables

4.5.3.5. Playing a File from Another OTG Device on the Current OTG Device
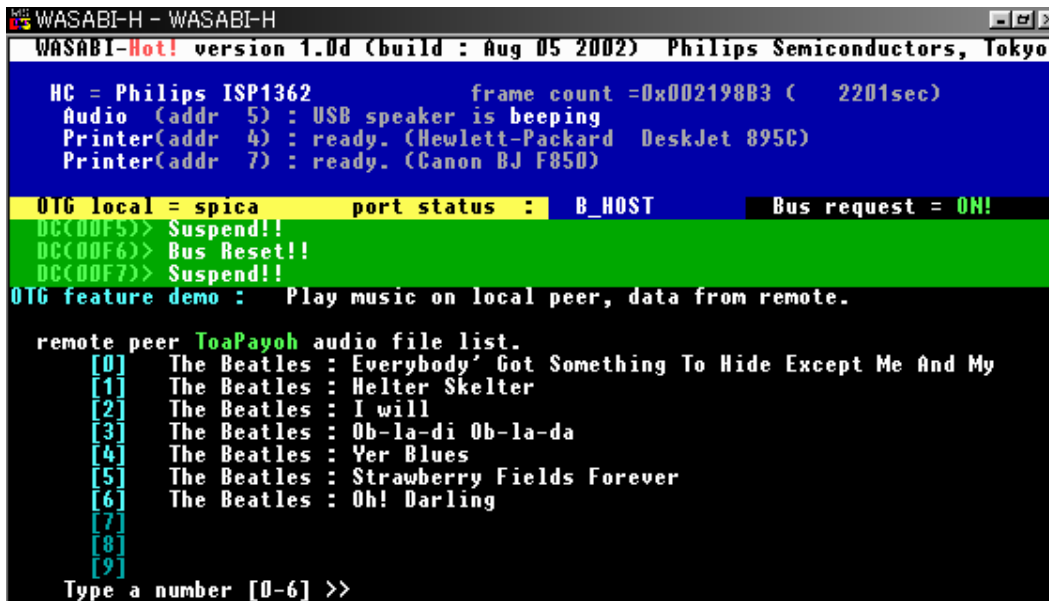

Figure 4-21: Playing a File from Another OTG Device on the Current OTG device

This is the most complex OTG demo on the WASABI-Hot! version 1.0. The local peer performs a bulk-IN transfer to retrieve WAVE data from the remote peer and feeds this data to USB speakers by using the isochronous-OUT transfer.

UM10017-01

User's Guide **Rev. 1.0—September 2002** **26 of 37**

This demo opens WAVE data file located on the other OTG device and reads it. The audio data is then streamed out to USB speakers.

Press the **a** key to play audio files located on the current OTG device. To play audio files located on the other OTG device, press the **a** key again or press the **o** key. The "music file list" located on the other OTG device will be displayed on the local host, see Figure 4-22. To start playing an audio file, type the file number. The local host will then open the file on the other OTG device and start to read, see Figure 4-23.

Figure 4-22: WAV File List on the Remote Peer

The audio play will continue till the end of file.

Other USB traffic might affect the latency of the remote file read. If the volume becomes too big, there will be audio buffer underrun. In such a situation, the host will terminate the audio play.

Figure 4-23: Playing Audio File Located on a Remote Peer

At the peripheral side in this demo, the other OTG device works as a data server and it will show the status of the data transfer. Progress of the data transfer and the remaining time will be shown in the green area of the screen, see Figure 4-24.

Figure 4-24: Peripheral Side Screen at the Remote File Local Play

4.6.   Other Features

WASABI-Hot! has some other utility features, described in this section.

4.6.1. Topology View

The **Spacebar** key command displays topology view of USB. It shows the tree structure of the current USB connection and address availability. See Figure 4-25.

Figure 4-25: Topology View

4.6.2. Forcing Root Hub Ports Reconnect

This feature has been implemented for debugging purposes.

The **r** key can force reconnect the device on the Root Hub. It first disconnects devices on port 1 and port 2, and then reconnects those devices. This reconnect action starts to enumerate all devices on the USB topology.

Forcing reconnect does not affect OTG status on port 1. If the OTG port is in the idle state, it does not change the state of OTG.

4.6.3. Environmental Information

To display current environmental information (that could be set by using command-option switches) of the application, press the **e** key, see Figure 4-26.

UM10017-01
User's Guide
Rev. 1.0—September 2002
29 of 37

Figure 4-26: Environmental Information

4.6.4. Version Information

The **v** key can be used to display information on the application. It will provide the application name, version, build date and time, and information on authors. See Figure 4-27.

Figure 4-27: Version Information

4.6.5. Help Information

The **?** or **h** key will display a list of key command references, see Figure 4-28.

Figure 4-28: Help Screen

4.6.6. Host Controller Registers Dump

The **d** key shows the Host Controller registers dumped values. The screen contains two pages. When the first page is displayed, press any key to go to the next page; see Figure 4-29.

Figure 4-29: Register Dump

4.6.7. Quitting the Application Execution

Press the **q** or **Esc** key to quit the application. The application will prompt you to confirm to continue the action. To quit, press **y**. Press any other key to cancel the action.

5. Inside of WASABI-Hot!

The structure of the source code is briefly discussed in this section.

5.1.   Rebuilding the Code

The WASABI-Hot! code is free. No restriction is applied on its distribution and modification. Most parts are written from scratch. Some portions of the code are also referenced to free codes. For code building, use Borland Turbo C++ 3.0 because there are some development environment dependencies in the source code (WASABI-Hot! is written in the C language). Definitions of interrupt service routines (ISRs) are in the Turbo C style, and the memory model and file access use Turbo C's library calls.

For rebuilding the code, use the project file that is included in the distribution package. It includes settings recommended parameters for building (for example, memory model, debug options, accessing directories, warning levels, and so on).

Before the code building, edit the appropriate parameters, such as the directory in which Turbo C is installed, and include source file directories, library directory, and so on (see Section 5.3.1).

5.2.   Units and Modules

In this document, "unit" refers to a big functional block that consists of some modules. WASABI-Hot! is a monolithic application. However, internally, the application is divided into some units or modules depending on the functionality. The source code files are placed in directories.
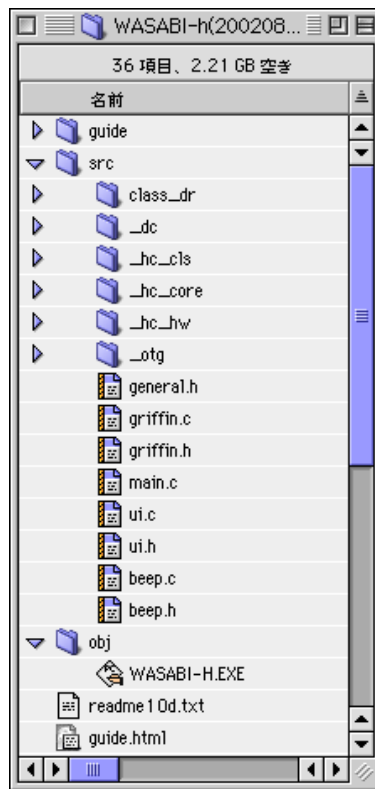
Figure 5-1: Source Directories

Table 5-1: Application-Level Unit

Directory	Description
src/	main function, sample drivers and user interfaces

Table 5-2: Host Controller Unit

Directory	Description
src/_hc_core/	USB management and ISP1362 HC control
src/_hc_hw/	Hardware access abstraction for the ISP1362 HC
src/_hc_cls/	Class driver interface
src/class_dr/	Class drivers and applets

Table 5-3: Device Controller Unit

Directory	Description
src/_dc/	Device control routine

Table 5-4: OTG Controller Unit

Directory	Description
src/_otg/	OTG modules (FSM, HC and DC control)
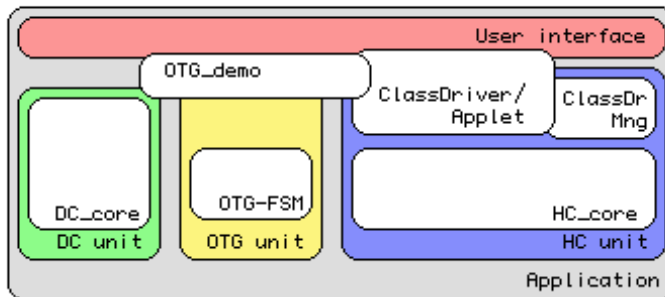
Figure 5-2: Software Functional Blocks

5.2.1. Application-Level Unit Modules

This unit contains the entry point of the application. It manages initializations, main loop and service routine call mechanisms. Some driver samples and user interface code files are also placed in this directory.

5.2.2. Host Controller Unit Modules

The entire Host Controller related processes are defined in the files in these directories. The HC unit has the following sub units:

- Hardware abstractions

- Bus management

- Class driver management

- Class driver/applets.

5.2.3. Device Controller Unit Modules

The important role of the Device Controller is to receive requests from the host and transmit or receive data according to the USB specification. The "dc_chap9.c" file includes control transfer handler, and all standard requests and descriptors. The "dc_spsfc.c" file defines vendor-specific requests for communicating with another OTG device. Using this request, remote device messages are displayed on the demo screen. In addition, endpoint 4 configured as bulk IN is used for receiving audio format data. The "dc_isr.c" file is used for transmitting or receiving data to or from each endpoint FIFO packet by packet.

5.2.4. OTG Controller Modules

The OTG Controller module includes OTG-FSM that has been ported from OTGC ver. 1.0. OTGC is a USB-OTG reference code provided by Philips Semiconductors. The "HAL4OTG.c", "OTGSVC.c" and "OTGISR.c" files are ported from OTGC ver. 1.0.

WASABI-Hot! has the interface layer for these imported modules ("otg.c").

This directory also includes an OTG demo program ("otg_demo.c"). It has several demo codes that work with user interactions. For more details, see Section 4.5 "Playing with OTG".

5.3.   Directory Structure of the Distribution Package
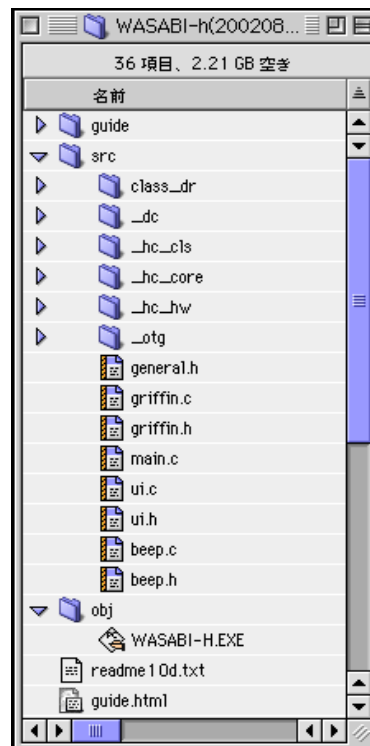
Figure 5-3: Source Directories

5.3.1. Top-Level Directory

The WASABI-Hot! distribution package has a directory structure for managing several file types. At the top level are the "src" and "obj" directories, whose contents are given in Table 5-5.

Table 5-5: File Content of the Top-Level Directory

File Name	Description
wasabi-h.prj	Project file for WASABI-Hot! Can be used in Turbo C
readme*.txt	Release notes
setting.wsb	Setting file example (See Section 3.2)
*.lst	Music file list sample

You can use the project file in this package to rebuild the code. This file includes some recommended setting for the development environment. You can use these files without making any changes. However, before using this file, edit the directory setting fields for your PC. The project file in this package has the settings listed in Table 5-6.

Table 5-6: Project File Settings in the Package

File Name	Description
d:\tc	For Turbo C environment
d:\prj\src	For source code files
d:\prj\obj	For file output

If you have different directory structure on your PC, you must edit those parameters.

5.3.2. "src" and "obj" Directories

The "src" directory is the source code directory. The "obj" directory has been created for compiler and linker output. All output files will be stored in the "obj" directory, including the "wasabi-h.exe" executable file. The "src" directory has substructures to sort files according to their functionality. Details will be described in the following section.

5.4. "src" Directory: List of Source Code Files

Table 5-7: Application-Level Modules

File Name	Description
src/main.c	Application main
src/general.h	General setting definitions
src/ui.c	User interface routines
src/ui.h	User interface routines header
src/beep.c	Sample implementation of custom audio driver
src/beep.h	Header for beep, sample implementation custom audio driver
src/griffin.c	Sample implementation of vendor-specific class driver
src/griffin.h	Header for griffin.c

Table 5-8: Host Controller Modules (Core of USB Management)

File Name	Description
src/_hc_core/atl_mix.c	Transfer controls/ISP1362 internal RAM manager for non-isochronous transfers
src/_hc_core/atl_mix.h	Header file for atl_mix.c
src/_hc_core/chap9.h	Chapter 9 related data definitions for the HC
src/_hc_core/dev_ep.c	Device and endpoint manager
src/_hc_core/dev_ep.h	Header for device and endpoint manager
src/_hc_core/init.c	Initialization routines
src/_hc_core/init.h	Initialization value definitions
src/_hc_core/isr.c	ISR management
src/_hc_core/isr.h	Header for the ISR management
src/_hc_core/port_ctl.c	Port control routines for the Root Hub and device hubs
src/_hc_core/port_ctl.h	Header for port control processes
src/_hc_core/transfer.c	Control transfer manager
src/_hc_core/transfer.h	Header for transfer handling definitions

Table 5-9: Host Controller Modules (Interface for Hardware)

File Name	Description
src/_hc_hw/hc_comm.c	ISP1362 HC register command list
src/_hc_hw/hw_acces.c	ISP1362 HC Hardware access/control routines
src/_hc_hw/hw_acces.h	Header for hw_acces.c
src/_hc_hw/pci.c	PCI card Plug and Play (PnP) handler
src/_hc_hw/pci.c	Header for pci.c

Table 5-10: Host Controller Modules (Device Class Related)

File Name	Description
src/_hc_cls/cls_hndl.c	HC class driver management routines
src/_hc_cls/cls_hndl.h	Header HC class driver management
src/class_dr/audio/sing.c	Class driving emulator/applet for audio play
src/class_dr/audio/sing.h	Class driving emulator/applet for audio play header
src/class_dr/audio/hid.c	HID class driver
src/class_dr/audio/hid.h	HID class driver header
src/class_dr/audio/hub.c	Hub class driver
src/class_dr/audio/hub.h	Hub class driver header
src/class_dr/audio/printer.c	Class driving emulator/applet for printer
src/class_dr/audio/printer.h	Class driving emulator/applet for printer header

UM10017-01
User's Guide   **Rev. 1.0—September 2002**   **36 of 37**

Table 5-11: Device Controller Modules

File Name	Description
src/_dc/dc_chap9.c	Control transfer handler, Chapter 9 handler
src/_dc/dc_chap9.h	Header file for dc_chap9.c
src/_dc/dc_comm.c	DC register access definition
src/_dc/dc_comm.h	Header file for dc_comm.c
src/_dc/dc_flags.c	Flag definitions
src/_dc/dc_flags.h	Header file for dc_flags.c
src/_dc/dc_hal.c	Simple HAL
src/_dc/dc_hal.h	Header file for dc_hal.c
src/_dc/dc_isr.c	DC ISR management
src/_dc/dc_isr.h	Header file for dc_isr.c
src/_dc/dc_main.c	Main routine when only DC is used
src/_dc/dc_main.h	Header file for dc_main.c
src/_dc/dc_spcfc.c	Vendor-specific request definition
src/_dc/dc_spcfc.h	Header file for dc_spcfc.c

Table 5-12: OTG Controller Modules

File Name	Description
src/_otg/hal4otg.c	Routine from OTGC, HAL for the OTG module
src/_otg/hal4otg.h	Header from OTGC, HAL for the OTG module header
src/_otg/otgisr.c	Routine from OTGC, ISR for the OTG module
src/_otg/otgsvc.c	Routine from OTGC, OTG state machine definition
src/_otg/otgsvc.h	Header from OTGC, OTG state machine definition
src/_otg/otg.c	Glue (bridging) code between OTGC and WASABI-Hot!
src/_otg/otg.h	Header for otg.c
src/_otg/otg_demo.c	OTG demo applet definitions
src/_otg/otg_demo.h	Header for otg_demo.c

5.5. "obj" Directory

The "obj" directory contains an executable file in the distribution package. After the code is rebuilt, all object files are saved in this directory. Turbo C uses this directory as an output directory.

Note: You can monitor the size of the binary code in each object files. The "objxref.com" tool in the "bin" directory of Turbo C works just like the "size" command in the UNIX tool.

Following is a sample of executing this command in the "obj" directory using the "/RC" option:

```
objxref *.obj /RC > result.txt
```

The "result.txt" file is the output target. This file contains the required memory size for the executable code and static variables after unresolved symbol information. This will be a good estimate for code size, if you want to port code.

6. References

- *ISP1362 Single-chip Universal Serial Bus On-The-Go controller* datasheet

- *Universal Serial Bus Specification Rev. 2.0*

- *On-The-Go Supplement to the USB 2.0 Specification Revision 1.0*

- *ISP1362 PCI Evaluation Board User's Guide*

Philips Semiconductors

Philips Semiconductors is a worldwide company with over 100 sales
offices in more than 50 countries. For a complete up-to-date list of
our sales offices please e-mail
sales.addresses@www.semiconductors.philips.com.
A complete list will be sent to you automatically.
You can also visit our website
http://www.semiconductors.philips.com/sales/

www.semiconductors.philips.com

PHILIPS