USB OHCI Host Controller User Guide

Version 1.40

For use with USBH OHCI Host Controller versions 2.14 and above

Exported on 10/23/2018

All rights reserved. This document and the associated software are the sole property of HCC Embedded. Reproduction or duplication by any means of any portion of this document without the prior written consent of HCC Embedded is expressly forbidden.

HCC Embedded reserves the right to make changes to this document and to the related software at any time and without notice. The information in this document has been carefully checked for its accuracy; however, HCC Embedded makes no warranty relating to the correctness of this document.
# Table of Contents

1  System Overview .................................................................................................................... 3  
   1.1 Introduction .......................................................................................................................... 4  
   1.2 Feature Check ....................................................................................................................... 5  
   1.3 Packages and Documents ..................................................................................................... 6  
      Packages .................................................................................................................................. 6  
      Documents ............................................................................................................................... 6  
   1.4 Change History ..................................................................................................................... 7  

2  Source File List ....................................................................................................................... 8  
   2.1 API Header File ..................................................................................................................... 8  
   2.2 Configuration File .................................................................................................................. 8  
   2.3 Source Code ........................................................................................................................ 8  
   2.4 Version File .......................................................................................................................... 8  
   2.5 Platform Support Package (PSP) Files .................................................................................. 9  

3  Configuration Options ............................................................................................................ 10  

4  Starting the OHCI Controller .................................................................................................. 12  
   4.1 usbh_ohci hc ....................................................................................................................... 12  
   4.2 Host Controller Task .......................................................................................................... 12  
   4.3 Code Example ..................................................................................................................... 13  

5  Integration .............................................................................................................................. 14  
   5.1 OS Abstraction Layer ......................................................................................................... 14  
   5.2 PSP Porting ......................................................................................................................... 15  
      ohci_hw_init ........................................................................................................................... 16  
      ohci_hw_start ....................................................................................................................... 17  
      ohci_hw_stop ....................................................................................................................... 18  
      ohci_hw_delete ..................................................................................................................... 19
1 System Overview

This chapter contains the fundamental information for this module.

The component sections are as follows:

- **Introduction** – describes the main elements of the module.
- **Feature Check** – summarizes the main features of the module as bullet points.
- **Packages and Documents** – the *Packages* section lists the packages that you need in order to use this module. The *Documents* section lists the relevant user guides.
- **Change History** – lists the earlier versions of this manual, giving the software version that each manual describes.
1.1 Introduction

This guide is for those who want to configure and use HCC's Open Host Controller Interface (OHCI) module with HCC's USB host stack. The OHCI module provides a high speed USB 1.1 host controller which provides full speed and low speed USB functions. The controller can handle all USB transfer types and, in conjunction with the USB host stack, can be used with any USB class driver.

The OHCI Host Controller package provides a host controller for a USB stack, as shown below.
1.2 Feature Check

The main features of the host controller are the following:

- Conforms to the HCC Advanced Embedded Framework.
- Designed for integration with both RTOS and non-RTOS based systems.
- Integrated with HCC USB Host stack and all its class drivers.
- Can be used with any OHCI-compliant USB host controller.
- Supports multiple simultaneous OHCI controllers, each with multiple devices attached.
- Supports all USB transfer types: Control, Bulk, Interrupt and Isochronous.
1.3 Packages and Documents

Packages

The table below lists the packages that you need in order to use this module:

<table>
<thead>
<tr>
<th>Package</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hcc_base_doc</td>
<td>This contains the two guides that will help you get started.</td>
</tr>
<tr>
<td>usbh_base</td>
<td>The USB host base package. This is the framework used by USB class drivers to communicate over USB using a specific USB host controller package.</td>
</tr>
<tr>
<td>usbh_drv_ohci</td>
<td>The USB OHCI host controller package described by this document.</td>
</tr>
</tbody>
</table>

Documents

For an overview of HCC’s embedded USB stacks, see Product Information on the main HCC website.

Readers should note the points in the HCC Documentation Guidelines on the HCC documentation website.

**HCC Firmware Quick Start Guide**

This document describes how to install packages provided by HCC in the target development environment. Also follow the Quick Start Guide when HCC provides package updates.

**HCC Source Tree Guide**

This document describes the HCC source tree. It gives an overview of the system to make clear the logic behind its organization.

**HCC USB Host Base System User Guide**

This document defines the USB host base system upon which the complete USB stack is built.

**HCC USB OHCI Host Controller User Guide**

This is this document.
1.4 Change History

This section describes past changes to this manual.

- To download this manual or a PDF describing an earlier software version, see USB Host PDFs.
- For the history of changes made to the package code itself, see History: usbh_drv_ohci.

The current version of this manual is 1.40. The full list of versions is as follows:

<table>
<thead>
<tr>
<th>Manual version</th>
<th>Date</th>
<th>Software version</th>
<th>Reason for change</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.30</td>
<td>2017-06-19</td>
<td>2.14</td>
<td>New Change History format.</td>
</tr>
<tr>
<td>1.20</td>
<td>2015-03-27</td>
<td>2.13</td>
<td>Removed Hardware-Specific Functions section.</td>
</tr>
<tr>
<td>1.10</td>
<td>2015-03-06</td>
<td>2.13</td>
<td>Added Change History.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Added Hardware-Specific Functions to PSP Porting.</td>
</tr>
<tr>
<td>1.00</td>
<td>2014-08-29</td>
<td>2.13</td>
<td>First release.</td>
</tr>
</tbody>
</table>
2 Source File List

This section describes all the source code files included in the system. These files follow the HCC Embedded standard source tree system, described in the HCC Source Tree Guide. All references to file pathnames refer to locations within this standard source tree, not within the package you initially receive.

Note: Do not modify any of these files except the configuration file and PSP files.

2.1 API Header File

The file src/api/api_usbh_ohci.h is the only file that should be included by an application using this module. It declares the usbh_ohci_hc() function.

2.2 Configuration File

The file src/config/config_usbh_ohci.h contains all the configurable parameters. Configure these as required. For details of these options, see Configuration Options.

2.3 Source Code

These files in src/usb-host/usb-driver/ohci are the source code files. These files should only be modified by HCC.

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ohci.c</td>
<td>Source file for OHCI code.</td>
</tr>
<tr>
<td>ohci.h</td>
<td>Header file for OHCI public functions.</td>
</tr>
<tr>
<td>ohci_hc.c</td>
<td>Source file for OHCI HC descriptor.</td>
</tr>
<tr>
<td>ohci_hc.h</td>
<td>OHCI-specific header file.</td>
</tr>
<tr>
<td>ohci_hub.c</td>
<td>Source file for OHCI hub.</td>
</tr>
<tr>
<td>ohci_hub.h</td>
<td>Header file for OHCI hub public functions.</td>
</tr>
<tr>
<td>ohci_reg.h</td>
<td>OHCI register file.</td>
</tr>
</tbody>
</table>

2.4 Version File

The file src/version/ver_usbh_ohci.h contains the version number of this module. This version number is checked by all modules that use this module to ensure system consistency over upgrades.
2.5 Platform Support Package (PSP) Files

These files are in the directory `src/psp/target/usb_host_ohci`. They provide functions and elements the core code may need to use, depending on the hardware.

**Note:**
- These are PSP implementations for the specific microcontroller and development board; you may need to modify these to work with a different microcontroller and/or board. See [PSP Porting](#) for details.
- In the package these files are offset to avoid overwriting an existing implementation. Copy them to the root `hcc` directory for use.

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>psp_usbh_ohci.c</td>
<td>Functions source code.</td>
</tr>
<tr>
<td>psp_usbh_ohci.h</td>
<td>Functions header file.</td>
</tr>
</tbody>
</table>

The PSP also has a version file, `ver_psp_usbh_ohci.h`. 
3 Configuration Options

Set the system configuration options in the file `src/config/config_usbh_ohci.h`. This section lists the available options and their default values.

**OHCI_TRANSFER_TASK_STACK_SIZE**

The stack size of the OHCI transfer task(s). The default is 1024.

**OHCI_MAX_DEVICE**

The maximum number of devices supported. The default is 4. This should correspond to the maximum number of physical devices that will be attached; each externally attached hub counts as a device.

**OHCI_MAX_EP**

The maximum number of Bulk, Isochronous, and Interrupt endpoints. The default is 8.

**OHCI_MAX_TRANSFERS**

The maximum number of simultaneous transfers. The default is 8.

**OHCI_MAX_ISO_TRANSFERS**

The maximum number of simultaneous isochronous transfers. The default is 32.

**OHCI_ISO_LATEST_START**

The latest bit in a frame at which an ISO transfer can be started. The default is 400.

**OHCI_HC_COUNT**

The number of OHCI controllers supported. (Some microcontrollers have more than one OHCI controller.) The maximum is 4 and the default is 1.

**OHCI_RENDIAN**

The reverse endianness between OHCI registers and the system. Set this to 1 if the OHCI controller is operating in reverse endianness relative to the processor. The default is 0.

**OHCI_DBUFFER_ADDRESS**

The dedicated buffer start address (0 = none). The default is 0.

**OHCI_DBUFFER_SIZE**

The dedicated buffer size. The default is 0.
OHCI_READ_REG_32, OHCI_WRITE_REG_32

These specify the read/write routines to use when accessing an OHCI register. Currently these are mapped to `psp_rreg32()` and `psp_wreg32()`.

These macros are defined in `psp/include/psp_reg.h` and the parameters are (base, offset) for read and (base, offset, value) for write.

**Note:** For the following options, \( n \) is 0 to 3. Only set values for the host controllers which are used.

**OHCI_BASE_n**

The base address of the OHCI register space, required if processor-specific registers are available for additional settings. The default is 0.

**OHCI_ISR_ID_n**

The ISR identifier of the OHCI controller. The default is 0. This is always platform/RTOS-specific.

**OHCI_ISR_PRIORITY_n**

The ISR priority of the OHCI controller. The default is 0. This is always platform/RTOS-specific.
4 Starting the OHCI Controller

This section shows how to start the host controller and describes the task created.

4.1 usbh_ohci_hc

This is the only external interface function. This is the host controller descriptor required by the `usbh_hc_init()` function.

**Format**

```c
extern void * const usbh_ohci_hc
```

4.2 Host Controller Task

The host controller task handles all completed transfers. Callback requested for the transfer is executed from this task.

The task has the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry point</td>
<td><code>ohci_transfer_task_n</code> (n=0/1/2/3)</td>
</tr>
<tr>
<td>Priority</td>
<td>OAL_HIGHEST_PRIORITY (USBH_TRANSFER_TASK_PRIORITY)</td>
</tr>
<tr>
<td>Stack size</td>
<td>This depends on the RTOS. 1024 is the default.</td>
</tr>
</tbody>
</table>
4.3 Code Example

This example shows how to initialize the OHCI host controller. Note the following:

- There is only one external interface function, `usbh_ohci_hc()`. You call the `usbh_hc_init()` function with this function as a parameter to link this host controller to the system.
- The last parameter in the `usbh_hc_init()` call is the number of the host controller. In this example OHCI has two controller units so the first call uses 0 and the second call uses 1.

```c
void start_usb_host_stack ( void )
{
    int rc;
    rc = hcc_mem_init();

    if ( rc == 0 )
    {
        rc = usbh_init(); /* Initialize the USB host stack */
    }

    if ( rc == 0 )
    {
        /* Attach first OHCI host controller */
        rc = usbh_hc_init( 0, usbh_ohci_hc, 0 );
    }

    if ( rc == 0 )
    {
        /* Attach second OHCI host controller */
        rc = usbh_hc_init( 0, usbh_ohci_hc, 1 );
    }

    if ( rc == 0 )
    {
        rc = usbh_start(); /* Start the USB host stack */
    }

    if ( rc == 0 )
    {
        rc = usbh_hc_start( 0 ); /* Start first OHCI Host controller */
    }

    if ( rc == 0 )
    {
        rc = usbh_hc_start( 1 ); /* Start second OHCI Host controller */
    }

    .....  
}
```
5 Integration

This section specifies the elements of this package that need porting, depending on the target environment.

5.1 OS Abstraction Layer

All HCC modules use the OS Abstraction Layer (OAL) that allows the module to run seamlessly with a wide variety of RTOSes, or without an RTOS.

This module requires the following OAL elements:

<table>
<thead>
<tr>
<th>OAL Resource</th>
<th>Number Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks</td>
<td>1</td>
</tr>
<tr>
<td>Mutexes</td>
<td>1</td>
</tr>
<tr>
<td>Events</td>
<td>1</td>
</tr>
<tr>
<td>ISRs</td>
<td>One for each supported OHCI host controller (OHCI_HC_COUNT).</td>
</tr>
</tbody>
</table>
5.2 PSP Porting

The Platform Support Package (PSP) is designed to hold all platform-specific functionality, either because it relies on specific features of a target system, or because this provides the most efficient or flexible solution for the developer.

The module makes use of the following standard PSP functions:

<table>
<thead>
<tr>
<th>Function</th>
<th>Package</th>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>psp_memcpy()</td>
<td>psp_base</td>
<td>psp_string</td>
<td>Copies a block of memory. The result is a binary copy of the data.</td>
</tr>
<tr>
<td>psp_memset()</td>
<td>psp_base</td>
<td>psp_string</td>
<td>Sets the specified area of memory to the defined value.</td>
</tr>
<tr>
<td>psp_rreg32()</td>
<td>psp_base</td>
<td>psp_reg</td>
<td>Read routine used to access an OHCI register.</td>
</tr>
<tr>
<td>psp_wreg32()</td>
<td>psp_base</td>
<td>psp_reg</td>
<td>Write routine used to access an OHCI register.</td>
</tr>
</tbody>
</table>

The host controller makes use of the following functions that must be provided by the PSP. These are designed for you to port them easily to work with your hardware solution. The package includes samples in the src/psp/target/usb-host-ohci/psp_usbh_ohci.c file.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ohci_hw_init()</td>
<td>Initializes the device.</td>
</tr>
<tr>
<td>ohci_hw_start()</td>
<td>Starts the device.</td>
</tr>
<tr>
<td>ohci_hw_stop()</td>
<td>Stops the device.</td>
</tr>
<tr>
<td>ohci_hw_delete()</td>
<td>Deletes the device, releasing the associated resources.</td>
</tr>
</tbody>
</table>

These functions are described in the following sections.

**Note:** HCC can provide samples for different configurations; contact support@hcc-embedded.com.
ohci_hw_init

This function must be provided by the PSP to initialize the device.

**Format**

```c
int ohci_hw_init ( t_usbh_unit_id unit )
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>unit</td>
<td>The unit ID.</td>
<td>t_usbh_unit_id</td>
</tr>
</tbody>
</table>

**Return Values**

<table>
<thead>
<tr>
<th>Return value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USBH_SUCCESS</td>
<td>Successful execution.</td>
</tr>
<tr>
<td>USBH_ERROR</td>
<td>Operation failed.</td>
</tr>
</tbody>
</table>
ohci_hw_start

This function must be provided by the PSP to start the device.

Format

```c
int ohci_hw_start ( t_usbh_unit_id unit )
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>unit</td>
<td>The unit ID.</td>
<td>t_usbh_unit_id</td>
</tr>
</tbody>
</table>

Return Values

<table>
<thead>
<tr>
<th>Return value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USBH_SUCCESS</td>
<td>Successful execution.</td>
</tr>
<tr>
<td>USBH_ERROR</td>
<td>Operation failed.</td>
</tr>
</tbody>
</table>
ohci_hw_stop

This function must be provided by the PSP to stop the device.

**Format**

```c
int ohci_hw_stop ( t_usbh_unit_id unit )
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>unit</td>
<td>The unit ID.</td>
<td>t_usbh_unit_id</td>
</tr>
</tbody>
</table>

**Return Values**

<table>
<thead>
<tr>
<th>Return value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USBH_SUCCESS</td>
<td>Successful execution.</td>
</tr>
<tr>
<td>USBH_ERROR</td>
<td>Operation failed.</td>
</tr>
</tbody>
</table>
ohci_hw_delete

This function must be provided by the PSP to delete the device, releasing the associated resources.

Format

```c
int ohci_hw_delete ( t_usbh_unit_id unit )
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>unit</td>
<td>The unit ID.</td>
<td>t_usbh_unit_id</td>
</tr>
</tbody>
</table>

Return Values

<table>
<thead>
<tr>
<th>Return value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USBH_SUCCESS</td>
<td>Successful execution.</td>
</tr>
<tr>
<td>USBH_ERROR</td>
<td>Operation failed.</td>
</tr>
</tbody>
</table>