# 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 Host Controller .....................................................................................................12
  4.1 usbh_renesas_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
    renesas_hw_init ..........................................................................................................................16
    renesas_hw_start .......................................................................................................................17
    renesas_hw_stop .......................................................................................................................18
    renesas_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 implement HCC Embedded's Renesas USB Host Controller with the HCC USB host stack.

The Renesas module provides a high speed USB 2.0 host controller which provides both full 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. Renesas MCUs supported include those in the SuperH SH-2A and SH7260 series, and the S5D9 MCU group.

The position of the host controller within the USB stack is 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 the HCC USB Host stack and all its class drivers.
• Supports multiple simultaneous Renesas 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_renesas</td>
<td>The USB Renesas 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 Renesas 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_renesas.

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>1.40</td>
<td>2018-10-24</td>
<td>2.21</td>
<td>Added PSP version files to Source Files.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Added USBH_PORT configuration option.</td>
</tr>
<tr>
<td>1.30</td>
<td>2018-06-25</td>
<td>2.21</td>
<td>Added references to S5D9 group to Introduction, Source Files and PSP Porting.</td>
</tr>
<tr>
<td>1.20</td>
<td>2017-06-19</td>
<td>2.20</td>
<td>New Change History format.</td>
</tr>
<tr>
<td>1.10</td>
<td>2015-03-30</td>
<td>2.14</td>
<td>Added Change History.</td>
</tr>
<tr>
<td>1.00</td>
<td>2015-03-05</td>
<td>2.14</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_renesas.h` is the only file that should be included by an application using this module. It declares the `usbh_renesas_hc()` function. For details, see Starting the Host Controller.

### 2.2 Configuration File

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

### 2.3 Source Code

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

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>renesas.c</td>
<td>Source file for Renesas code.</td>
</tr>
<tr>
<td>renesas.h</td>
<td>Header file for Renesas public functions.</td>
</tr>
<tr>
<td>renesas_hc.c</td>
<td>Source file for the Renesas HC descriptor.</td>
</tr>
<tr>
<td>renesas_hc.h</td>
<td>HC descriptor header file.</td>
</tr>
<tr>
<td>renesas_hub.c</td>
<td>Source file for Renesas hub.</td>
</tr>
<tr>
<td>renesas_hub.h</td>
<td>Header file for Renesas hub public functions.</td>
</tr>
<tr>
<td>renesas_hw.h</td>
<td>Header file for Renesas hardware-specific functions.</td>
</tr>
</tbody>
</table>

### 2.4 Version File

The file `src/version/ver_usbh_renesas.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

There are three sets of files, in directories named `src/psp_rx63/target`, `src/psp_s5d9/target` and `src/psp_sh7269/target`. These 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 board; you may need to modify these to work with a different microcontroller and/or development 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.

The files are as follows:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>config/config_usbh_renesas.h</code></td>
<td>Configuration file with appropriate values for the device (absent from <code>psp_s5d9</code>).</td>
</tr>
<tr>
<td><code>target/include/hcc_xxxx_regs.h</code></td>
<td>Register definitions.</td>
</tr>
<tr>
<td><code>target/usb-host-renesas/psp_usbh_renesas.c</code></td>
<td>Functions source code.</td>
</tr>
<tr>
<td><code>target/usb-host-renesas/psp_usbh_renesas.h</code></td>
<td>Header file for functions.</td>
</tr>
</tbody>
</table>

The following version files are used:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ver_psp_reg.h</code></td>
<td>PSP version.</td>
</tr>
<tr>
<td><code>ver_psp_usbh_renesas_hw.h</code></td>
<td>Hardware version.</td>
</tr>
</tbody>
</table>
# 3 Configuration Options

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

**USBH_PORT_USED**

The port to use. The default is 0. This controls the value of the last option in the file, `USBH_PORT`.

**RENESAS_TRANSFER_TASK_STACK_SIZE**

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

**RENESAS_ISR_ID**

The ISR ID of the host controller. The default is `HCC_VECT_USB0_USB10`.

**RENESAS_INT_PRIO**

The ISR priority of the host controller. The default is 1.

**RENESAS_SUPPORT_HS**

Set this to 1 to support high speed transfers (for high speed SH7264 devices). The default is 0.

**RENESAS_BIGEND_FIFO_ACCESS**

Keep this at the default of 1 if big-endian FIFO port control is needed (for high speed SH7264 devices).

**RENESAS_32BIT_FIFO_ACCESS**

Keep this at the default of 1 for high speed SH7264 devices. (SH7264 devices allow 32 bit access to FIFO.)

**RENESAS_MAX_EP**

The maximum number of bulk and interrupt endpoints, including EP0s. The default is 15.

**RENESAS_ISOCHRONOUS_SUPPORT**

Set this to 1 to support isochronous transfers. The default is 0.

**RENESAS_NUM_BULKPIPE_RX**

The pipe to use for bulk RX transfers. The default is 2.

**RENESAS_NUM_BULKPIPE_TX**

The pipe to use for bulk TX transfers. The default is 1.
RENESAS_NUM_INTPIPE_RX

The pipe to use for interrupt RX transfers. The default is 3.

RENESAS_NUM_INTPIPE_TX

The pipe to use for interrupt TX transfers. The default is 1.

USBH_PORT

If USBH_PORT_USED is set to 0, this is set to HCC_USB0_BASE. Otherwise it is set to HCC_USB1_BASE.
4 Starting the Host Controller

This section shows how to start the host controller and describes the task created. It includes a code example.

4.1 usbh_renesas_hc

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

**Format**

```
extern void * const usbh_renesas_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>renesas_transfer_task</code></td>
</tr>
<tr>
<td>Priority</td>
<td>USBH_TRANSFER_TASK_PRIORITY</td>
</tr>
<tr>
<td>Stack size</td>
<td><code>RENESAS_TRANSFER_TASK_STACK_SIZE</code>. The default is 1024.</td>
</tr>
</tbody>
</table>
4.3 Code Example

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

- There is only one external interface function, `usbh_renesas_hc()`. To link this host controller to the system, you call the `usbh_hc_init()` function with this function as a parameter.
- The last parameter in the `usbh_hc_init()` call is the number of the host controller.

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

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

    if ( rc == 0 )
    {
        /* Attach Renesas host controller */
        rc = usbh_hc_init( 0, usbh_renesas_hc, 0 );
    }

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

    if ( rc == 0 )
    {
        rc = usbh_hc_start( 0 ); /* Start Renesas 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>2</td>
</tr>
<tr>
<td>Events</td>
<td>2</td>
</tr>
<tr>
<td>ISRs</td>
<td>1</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 macro:

<table>
<thead>
<tr>
<th>Macro</th>
<th>Package</th>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSP_RD_LE16</td>
<td>psp_base</td>
<td>psp_endianness</td>
<td>Reads a 16 bit value stored as little-endian from a memory location.</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 for the RX63, SH7269 and S5D9 devices in the respective psp_usbh_renesas.c files.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>renesas_hw_init()</td>
<td>Initializes the device.</td>
</tr>
<tr>
<td>renesas_hw_start()</td>
<td>Starts the device.</td>
</tr>
<tr>
<td>renesas_hw_stop()</td>
<td>Stops the device.</td>
</tr>
<tr>
<td>renesas_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.
renesas_hw_init

This function is provided by the PSP to initialize the device.

**Format**

```c
int renesas_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>
reensas_hw_start

This function is provided by the PSP to start the device.

**Format**

```c
int reensas_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>
renesas_hw_stop

This function is provided by the PSP to stop the device.

**Format**

```c
int renesas_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>
renesas_hw_delete

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

**Format**

```c
int renesas_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>