# 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 ...........................................................................9
   2.5 Platform Support Package (PSP) Files ....................................9

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

4  Starting the Host Controller ................................................11
   4.1 usbh_isp_hc .......................................................................11
   4.2 Host Controller Task .........................................................11
   4.3 Code Example .....................................................................12

5  Integration ...............................................................................13
   5.1 OS Abstraction Layer .........................................................13
   5.2 PSP Porting .........................................................................14
      isp_hwinit ............................................................................15
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.

**Note:** To download this manual as a PDF, see USB Host PDFs.
1.1 Introduction

This guide is for those who want to implement HCC Embedded’s USB host stack with NXP Semiconductors’ ISP1362 USB host controller.

The ISP1362 includes a USB host controller and a USB device controller. The USB device controller is actually an ISP1181, for which we provide a separate low level driver. These two drivers can be used together as required. This manual only covers the USB host functionality of the ISP1362. The ISP1181 is covered in the USB Device Low Level Driver for ISP1181 User Guide.

The ISP1362 module provides a high speed USB 2.0 host controller that 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.

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 NXP Semiconductor ISP1362 USB host controllers.
- Supports multiple simultaneous ISP1362 controllers, each with multiple devices attached.
- Can work together with HCC’s USB device controller for the ISP1181.
- 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_isp1362</td>
<td>The USB 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 ISP1362 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_isp1362.

The current version of this manual is 1.20. 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>1.20</td>
<td>2019-03-08</td>
<td>1.02</td>
<td>New template.</td>
</tr>
<tr>
<td>1.10</td>
<td>2017-06-19</td>
<td>1.02</td>
<td>New Change History format.</td>
</tr>
<tr>
<td>1.00</td>
<td>2015-12-21</td>
<td>1.01</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_isp1362.h is the only file that should be included by an application using this module. It declares the usbh_isp_hc() function. For details, see Starting the Host Controller.

2.2 Configuration File

The file src/config/config_usbh_isp1362.h contains all the configurable parameters. Configure these as required. For details, see Configuration Options.

2.3 Source Code

The source code files are in the directory src/usb-host/usb-driver/isp1362. These files should only be modified by HCC.

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>isp1362.c</td>
<td>Source file for ISP1362 code.</td>
</tr>
<tr>
<td>isp1362.h</td>
<td>Header file for ISP1362 public functions.</td>
</tr>
<tr>
<td>isp1362_hc.c</td>
<td>Source file for the ISP1362 HC descriptor.</td>
</tr>
<tr>
<td>isp1362_hc.h</td>
<td>HC descriptor header file.</td>
</tr>
<tr>
<td>isp1362_hub.c</td>
<td>Source file for public hub functions.</td>
</tr>
<tr>
<td>isp1362_hub.h</td>
<td>Header file for public hub functions.</td>
</tr>
<tr>
<td>isp1362_reg.h</td>
<td>Header file for ISP1362 registers.</td>
</tr>
<tr>
<td>isp1362_rw.c</td>
<td>Source code for functions including read/write.</td>
</tr>
<tr>
<td>isp1362_rw.h</td>
<td>Header file for functions including read/write.</td>
</tr>
</tbody>
</table>
2.4 Version File

The file src/version/ver_usbh_isp1362.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 named src/psp/target/usbh_isp1362. These provide functions and elements the core code may need to use.

---

**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>psp_usbh_isp1362.c</td>
<td>Source code of functions.</td>
</tr>
<tr>
<td>psp_usbh_isp1362.h</td>
<td>Header file for functions.</td>
</tr>
</tbody>
</table>
3 Configuration Options

Set the following system configuration options in the file `src/config/config_usbh_isp1362.h`. This section lists the available options and their default values.

**Note:** For full details of these options, refer to the manufacturer’s manual for the device.

**MAX_DEVICE**

The maximum number of devices supported. The default is 4.

**MAX_EP**

The maximum number of endpoints; the total of bulk, isochronous, and interrupt endpoints. The default is 20.

**ISP_HOST_ISR**

The ISR. The default is 0x0001176F.

**ISP_HOST_IT_PRIO**

The priority of the ISP device interrupt. The default is 0.

**USBH_ISP_TRANSFER_STACK_SIZE**

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

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

**Format**

```
extern void * const usbh_isp_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>usbh_isp_transfer_task</code></td>
</tr>
<tr>
<td>Priority</td>
<td>USBH_TRANSFER_TASK_PRIORITY</td>
</tr>
<tr>
<td>Stack size</td>
<td><code>USBH_ISP_TRANSFER_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_isp_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 ISP1362 host controller */
        rc = usbh_hc_init( 0, usbh_isp_hc, 0 );
    }
    if ( rc == 0 )
    {
        rc = usbh_start(); /* Start USB host stack */
    }
    if ( rc == 0 )
    {
        rc = usbh_hc_start( 0 ); /* Start ISP1362 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>0</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 function:

<table>
<thead>
<tr>
<th>Function</th>
<th>Package</th>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>
</tbody>
</table>

The host controller makes use of the following function provided by the PSP template file psp_usbh_isp1362.h. This is target-specific.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>isp_hwinit()</td>
<td>Initializes the device.</td>
</tr>
</tbody>
</table>

This function is described in the following section.

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

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

This enables the clocks, GPIO pin, external memory interface, and so on.

**Format**

```c
int isp_hwinit ( void )
```

**Arguments**

None.

**Return Values**

<table>
<thead>
<tr>
<th>Return value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful execution.</td>
</tr>
<tr>
<td>Else</td>
<td>Operation failed.</td>
</tr>
</tbody>
</table>