Media Driver for USB Host Mass Storage User Guide

Version 1.40

For use with Media Driver for USB Host Mass Storage versions 1.02 and above

Exported on 02/22/2019

Table of Contents

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.

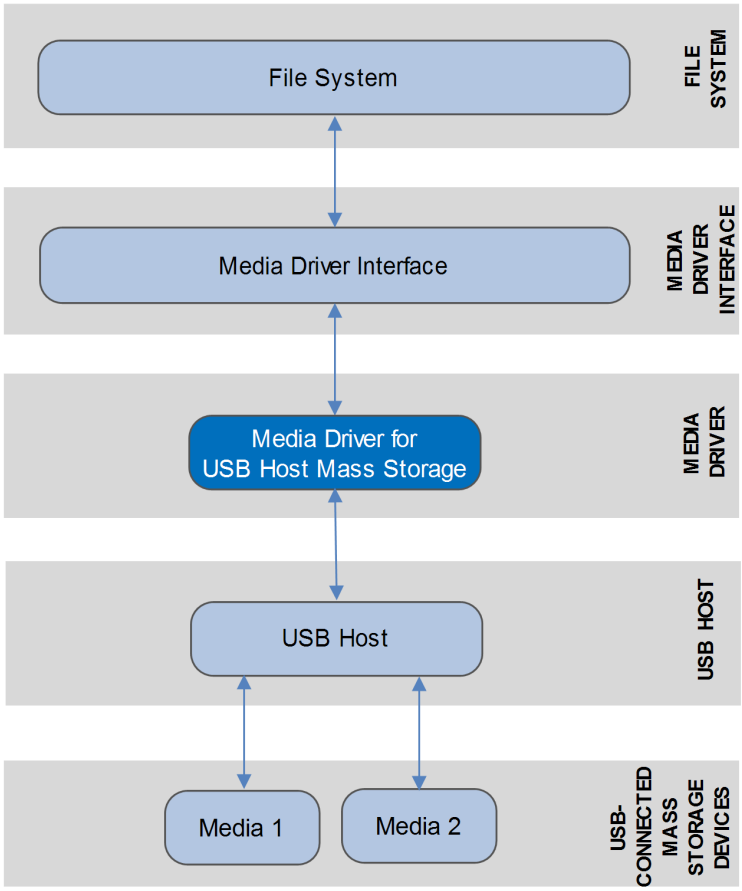
Note: To download this manual as a PDF, see File System Media Driver PDFs.

1.1 Introduction

This guide is for those who want to create a media driver for the USB Host Mass Storage (MST) class driver. It covers all aspects of configuration and use. This media driver conforms to the *HCC Media Driver Interface Specification*.

The media driver provides an interface for a file system to read from and write to a USB mass storage device. Examples of devices include pen drives/thumb drives, USB disk drives, and other USB media connected to an embedded system. A single media driver can support one or more physical media, each of these being represented as a different drive at the media driver interface. The file system handles all drives identically, regardless of their internal design features, though these may vary widely.

This diagram shows a typical system architecture including a file system, media driver, USB host, and media.



1.2 Feature Check

The main features of the media driver are the following:

- Conforms to the HCC Advanced Embedded Framework.
- Conforms to the *HCC Media Driver Interface Specification*.
- Designed for integration with both RTOS and non-RTOS based systems.
- Supports multiple USB Mass Storage drives.

1.3 Packages and Documents

Packages

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

Package	Description
hcc_base_doc	This contains the two guides that will help you get started.
media_drv_base	The base media driver package that includes the framework all media drivers use.
media_drv_mst	The media driver package described in this document.
usbh_cd_mst	The USB Host Mass Storage class driver.

Documents

For an overview of HCC file systems and data storage, 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 Media Driver Interface Guide

This document describes the HCC Media Driver Interface Specification.

HCC Embedded USB Host Mass Storage Class Driver User Guide

This document describes the USB Host Mass Storage class driver.

HCC Media Driver for USB Host Mass Storage 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 File System Media Driver PDFs.
- For the history of changes made to the package code itself, see History: media_drv_mst.

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

Manual version	Date	Software version	Reason for change
1.40	2019-02-22	1.02	New template.
1.30	2017-08-18	1.02	Updated *Packages* list.
1.20	2017-06-22	1.02	New *Change History* format.
1.10	2015-05-08	1.02	Various small changes.
1.00	2014-08-13	1.02	First online version.

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.

2.1 API Header File

The file **src/api/api_mdriver_mst.h** is the only file that should be included by an application using this module. For details of the single API function, see Application Programming Interface.

2.2 System File

The file **src/media-driver/mst/mst.c** is the source code for the media driver. **This file should only be modified by HCC**.

2.3 Version File

The file **src/version/ver_mdriver_mst.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.

3 Application Programming Interface

This section describes the single function and the structure it uses.

When the media driver is used:

1. The file system calls the media driver's **mst_initfunc()** function.
2. **mst_initfunc()** returns a pointer to an F_DRIVER structure containing a set of functions for accessing the media driver.

3.1 mst_initfunc

Use this function to initialize the interface with the driver.

The caller passes a parameter to the initialization function of a conforming driver. The driver returns a pointer to an F_DRIVER structure defining the interface to that driver.

Note: The call must allocate or use a static structure for the F_DRIVER structure. It must return a pointer to this structure, which must contain all the driver entry points, and also other data as required.

Format

```
F_DRIVER * ( mst_initfunc )( unsigned long driver_param )
```

Arguments

Argument	Description	Type
driver_param	The drive to use. The first drive is 0. This cannot be greater than (SCSI_MAX_UNITS*SCSI_MAX_LUN - 1). These values are taken from the SCSI interface of the HCC USB Host Mass Storage class driver.	unsigned long

Return values

Return value	Description
F_DRIVER *	A pointer to the driver structure, or NULL if the request failed.

3.2 F_DRIVER

This is the format of the *F_DRIVER* structure. This structure is defined in the HCC Media Driver Interface Specification.

Element	Type	Description
separated	int	Non-zero if the driver is separated.
user_data	unsigned long	User-defined data.
user_ptr	void *	User-defined pointer.
writesector	F_WRITESECTOR	Write a sector to the drive. This is mandatory if format or any write access is required.
writemultiplesector	F_WRITEMULTIPLESECTOR	Write a series of sectors to the drive. If this is unavailable F_WRITESECTOR may be used.
readsector	F_READSECTOR	Read a sector from the drive.
readmultiplesector	F_READMULTIPLESECTOR	Read a series of sectors from the drive. If this is unavailable F_READSECTOR may be used.
getphy	F_GETPHY	Used to get the physical properties of the drive, such as the number of sectors.
getstatus	F_GETSTATUS	(Only for removable drives) Used to test whether a drive has been removed or changed.
release	F_RELEASE	Release any resources associated with a drive when it is freed by the host (file) system.
ioctl	F_IOCTL	Used to send user-defined messages to the driver and get a response.

4 Integration

This section specifies the single element of this package that needs porting, depending on the target environment.

4.1 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 media driver makes use of the following standard PSP function:

Function	Package	Component	Description
psp_memset()	psp_base	psp_string	Sets the specified area of memory to the defined value.