

Message Digest Algorithm 5 (MD5) User's Guide

Version 1.00 BETA

For use with MD5 versions 1.3 and above

Date: 11-Feb-2015 11:32

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

| | |
|---|----|
| System Overview | 4 |
| Introduction | 4 |
| Feature Check | 5 |
| Packages and Documents | 5 |
| Packages | 5 |
| Documents | 5 |
| Source File List | 6 |
| API Header File | 6 |
| Configuration File | 6 |
| System File | 6 |
| Version File | 6 |
| Configuration Option | 7 |
| Application Programming Interface (API) | 8 |
| md5_init_fn | 8 |
| Hash Output Size | 9 |
| Error Codes | 10 |
| Integration | 11 |
| PSP Porting | 11 |

Version 1.00 BETA

For use with MD5 versions 1.3 and above

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.

 [Encryption Documents Home](#)

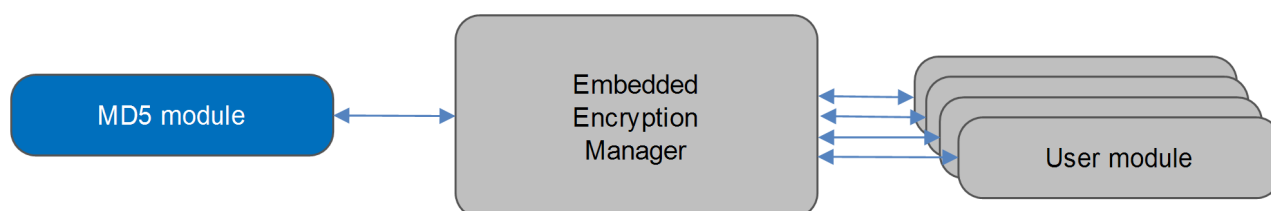
1 System Overview

1.1 Introduction

This guide is for those who want to generate hash codes for data by using the MD5 Message Digest Algorithm. The MD5 module implements the MD5 hash algorithm. The MD5 method, which produces a 16 byte hash value, has been superseded by the SHA-1 and SHA-256 hash algorithms.

You register the MD5 module with HCC's Embedded Encryption Manager (EEM), making it usable by other applications (for example, HCC's TLS/SSL) through a standard interface. The EEM is the core component of HCC's encryption system.

The system structure is shown below:

**Note:**

- Although every attempt has been made to simplify the system's use, to get the best results you must understand clearly the requirements of the systems you design.
- HCC Embedded offers hardware and firmware development consultancy to help you implement your system; contact sales@hcc-embedded.com.

1.2 Feature Check

The main features of the MD5 module are the following:

- It conforms to the HCC Advanced Embedded Framework.
- It conforms to the HCC Coding Standard including full MISRA compliance.
- It conforms to the HCC Embedded Encryption Manager (EEM) standard and is compatible with the EEM.
- It can be verified using the HCC Encryption Test Suite.

1.3 Packages and Documents

Packages

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

| Package | Description |
|----------------------------|--|
| <code>hcc_base_docs</code> | This contains the two guides that will help you get started. |
| <code>enc_base</code> | The EEM base package. |
| <code>enc_md5</code> | The MD5 package described in this document. |

Documents

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 Embedded Encryption Manager User's Guide

This document describes the EEM.

HCC Message Digest Algorithm 5 (MD5) User's Guide

This is this document.

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 files except the configuration file.

2.1 API Header File

The file `src/api/api_enc_sw_md5.h` is the only file that should be included by an application using this module. It defines the `md5_init_fn()` function.

2.2 Configuration File

The file `src/config/config_enc_sw_md5.h` contains the single [configurable parameter](#) of the system. Configure this as required. This is the only file in the module that you should modify.

2.3 System File

The file `src/enc/software/md5/md5.c` is the source code file. **This file should only be modified by HCC.**

2.4 Version File

The file `src/version/ver_enc_sw_md5.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 Configuration Option

Set the single system configuration option in the file `src/config/config_enc_sw_md5.h`.

MD5_INSTANCE_NR

The number of allowed MD5 driver instances. The default is 2.

4 Application Programming Interface (API)

This section describes the single API function, the hash output size, and the error codes.

4.1 md5_init_fn

Call this function from the EEM to forward the structure containing MD5 functions to it.

Format

```
t_enc_ret md5_init_fn ( t_enc_driver_fn const * * const pp_encdriver )
```

Arguments

| Parameter | Description | Type |
|--------------|--|--------------------|
| pp_encdriver | A pointer to a structure containing MD5 functions. | t_enc_driver_fn ** |

Return Values

| Return value | Description |
|-----------------|--|
| ENC_SUCCESS | Successful execution. |
| ENC_INVALID_ERR | The module has already been initialized. |

4.2 Hash Output Size

The hash output size is defined in the file `src/api/api_enc_sw_md5.h`.

| Name | Value | Description |
|-------------|-------|---------------------------------------|
| MD5_OUT_LEN | 16U | The size of the hash output in bytes. |

4.3 Error Codes

The table below lists the error codes that may be generated by the API calls.

| Error code | Value | Meaning |
|-----------------|-------|--|
| ENC_SUCCESS | 0U | Successful execution. |
| ENC_INVALID_ERR | 1U | The module has already been initialized. |

5 Integration

The MD5 module is designed to be as open and as portable as possible. No assumptions are made about the functionality, the behavior, or even the existence, of the underlying operating system. For the system to work at its best, perform the porting outlined below. This is a straightforward task for an experienced engineer.

5.1 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 module makes use of the following standard PSP functions:

| Function | Package | Element | Description |
|---------------------|----------|------------|---|
| psp_memcpy() | psp_base | psp_string | Copies a block of memory. The result is a binary copy of the data. |
| psp_memset() | psp_base | psp_string | Sets the specified area of memory to the defined value. |

The module makes use of the following standard PSP macros:

| Macro | Package | Element | Description |
|-------------|----------|----------------|---|
| PSP_RD_LE32 | psp_base | psp_endianness | Reads a 32 bit value stored as little-endian from a memory location. |
| PSP_WR_LE32 | psp_base | psp_endianness | Writes a 32 bit value to be stored as little-endian to a memory location. |