



HASH internal peripheral



## Contents

1. HASH internal peripheral .....	3
2. STM32MP15 resources .....	6
3. ETZPC internal peripheral .....	6
4. OP-TEE overview .....	7
5. Crypto API overview .....	7
6. STM32CubeMP1 architecture .....	7
7. STM32CubeMX .....	7
8. STM32MPU Embedded Software architecture overview .....	7
9. How to assign an internal peripheral to a runtime context .....	7



# HASH internal peripheral

Stable: 12.02.2020 - 16:46 / Revision: 12.02.2020 - 16:44

## Contents

1 Article purpose .....	3
2 Peripheral overview .....	3
<b>2.1 Features</b> .....	<b>4</b>
<b>2.2 Security support</b> .....	<b>4</b>
3 Peripheral usage and associated software .....	4
<b>3.1 Boot time</b> .....	<b>4</b>
<b>3.2 Runtime</b> .....	<b>4</b>
3.2.1 Overview .....	4
3.2.2 Software frameworks .....	4
3.2.3 Peripheral configuration .....	5
3.2.4 Peripheral assignment .....	5
4 How to go further .....	6
5 References .....	6

## 1 Article purpose

The purpose of this article is to:

- briefly introduce the HASH peripheral and its main features
- indicate the level of security supported by this hardware block
- explain how each instance can be allocated to the three runtime contexts and linked to the corresponding software components
- explain, when necessary, how to configure the HASH peripheral.

## 2 Peripheral overview

The **HASH** peripheral is used to compute a message digest.

Digest algorithms could be:

- MD5<sup>[1]</sup>
- SHA<sup>[2]</sup> (1, 224, 256)

The **HASH** peripheral is also able to give the HMAC<sup>[3]</sup> used for authentication using the same algorithm support.



## 2.1 Features

Refer to the [STM32MP15 reference manuals](#) for the complete list of features, and to the software components, introduced below, to see which features are implemented.

## 2.2 Security support

HASH1 is a **secure** peripheral (under ETZPC control)

HASH2 is a **non secure** peripheral .

# 3 Peripheral usage and associated software

## 3.1 Boot time

HASH1 instance is used as boot device to support binary authentication.

HASH2 is not used at boot time.

## 3.2 Runtime

### 3.2.1 Overview

HASH1 instance can be allocated to:

- the Arm<sup>®</sup> Cortex<sup>®</sup>-A7 secure core to be controlled in OP-TEE by the [OP-TEE HASH driver](#)

or

- the Arm<sup>®</sup> Cortex<sup>®</sup>-A7 non-secure core to be controlled in Linux<sup>®</sup> by the [Linux Crypto framework](#)

HASH2 instance can be allocated to:

- the Arm<sup>®</sup> Cortex<sup>®</sup>-M4 to be controlled in STM32Cube MPU Package by [STM32Cube HASH driver](#)

Chapter [Peripheral assignment](#) describes which peripheral instance can be assigned to which context.

### 3.2.2 Software frameworks

Do	Peri	Software frameworks	Comment
Cor tex -A7	Cor tex -A7 no	Cortex-M4	

Do	Peri	Software frameworks			Comment
main (OP-TEE)	non-secure (Linux)	(STM32Cube)			
		OP-TEE HASH driver	Linux Crypto framework	STM32Cube HASH driver	

### 3.2.3 Peripheral configuration

The configuration is applied by the firmware running in the context to which the peripheral is assigned. The configuration can be done alone via the **STM32CubeMX** tool for all internal peripherals, and then manually completed (particularly for external peripherals), according to the information given in the corresponding software framework article.

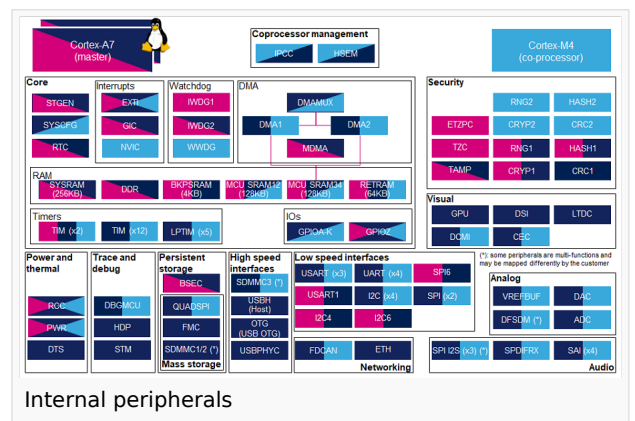
### 3.2.4 Peripheral assignment

**Check boxes** illustrate the possible peripheral allocations supported by **STM32 MPU Embedded Software**:

- means that the peripheral can be assigned () to the given runtime context.
- is used for system peripherals that cannot be unchecked because they are statically connected in the device.

Refer to [How to assign an internal peripheral to a runtime context](#) for more information on how to assign peripherals manually or via **STM32CubeMX**.

The present chapter describes **STMicroelectronics** recommendations or choice of implementation. Additional possibilities might be described in **STM32MP15** reference manuals.



Internal peripherals

Do	Peri	Runtime allocation		Comment
main (Insecure)	Cortex-A7 (Linux)	Cortex-A7 non-secure (Linux)	Cortex-M4 (STM32Cube)	



Do ma in	Per m i t t e r e n c e	Runtime allocation				Comme nt
S e c u r i t y	H A S H	HASH1				Assig nment (singl e choic e )
		HASH2				

## 4 How to go further

Not applicable.

## 5 References

- <https://en.wikipedia.org/wiki/MD5>
- [https://en.wikipedia.org/wiki/Secure\\_Hash\\_Algorithms](https://en.wikipedia.org/wiki/Secure_Hash_Algorithms)
- <https://en.wikipedia.org/wiki/HMAC>

Message Digest 5

Secure Hash Algorithm

Hash-based Message Authentication Code

Open Portable Trusted Execution Environment

Microprocessor Unit

## Permission error

Stable: 24.06.2020 - 17:52 / Revision: 24.06.2020 - 12:32

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer



## Permission error

---

*Stable: 20.05.2020 - 10:36 / Revision: 18.05.2020 - 12:05*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer

## Permission error

---

*Stable: 13.05.2020 - 08:56 / Revision: 13.05.2020 - 08:54*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer

## Permission error

---

*Stable: 19.06.2020 - 08:24 / Revision: 19.06.2020 - 08:18*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer

## Permission error

---

*Stable: 21.02.2020 - 08:39 / Revision: 04.02.2020 - 15:22*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer

## Permission error

---

*Stable: 31.01.2020 - 13:04 / Revision: 31.01.2020 - 13:02*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer

## Permission error

---

*Stable: 15.10.2019 - 11:55 / Revision: 15.10.2019 - 11:55*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer



## Permission error

---

*Stable: 22.06.2020 - 09:50 / Revision: 22.06.2020 - 09:49*

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST\_editors, ST\_readers, Selected\_editors, sysop, reviewer