Document Title	Specification and Integration of Hardware Test Management at start up and shutdown
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	804

Document Status	published
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	R24-11

Document Change History

Date	Release	Changed by	Description
2024-11-27	R24-11	AUTOSAR Release Management	• No content changes
2023-11-23	R23-11	AUTOSAR Release Management	• No content changes
2022-11-24	R22-11	AUTOSAR Release Management	• No content changes
2021-11-25	R21-11	AUTOSAR Release Management	• No content changes
2020-11-30	R20-11	AUTOSAR Release Management	• No content changes
2019-11-28	R19-11	AUTOSAR Release Management	• No content changes • Changed Document Status from Final to published
2018-10-31	4.4.0	AUTOSAR Release Management	• Minor corrections
2017-12-08	4.3.1	AUTOSAR Release Management	• Editorial changes

▽

△

2016-11-30	4.3.0	AUTOSAR Release Management	• Inital Release

Disclaimer

This work (specification and/or software implementation) and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the work.

The material contained in this work is protected by copyright and other types of intellectual property rights. The commercial exploitation of the material contained in this work requires a license to such intellectual property rights.

This work may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only. For any other purpose, no part of the work may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The work has been developed for automotive applications only. It has neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Contents

1 Scope of this document

This document gives a brief description about the Hardware test management concept integration in standard AUTOSAR software platform.

It shall serve as a user guide for those who wants to implement the hardware test management start up and shutdown module as an AUTOSAR BSW module following the AUTOSAR methods and process.

The requirements of HTMSS module itself are described in HTMSS RS and SWS documents.

The contents of this document mainly describes the below aspects:

- Description of main feature in integrating hardware specific tests in AUTOSAR based ECU's
- The impact in AUTOSAR architecture and solution
- Definition of requirements of the impacted modules in AUTOSAR in order to integrate the HTMSS within the std. AUTOSAR software sequence and its corresponding behaviour in the ECU software

1.1 Limitations

None

2 Definition of terms and acronyms

Abbreviation / Acronym:	Description:
HTMSS	Hardware Tests Management Start up and Shutdown
DEM	Diagnostic Event Manager
ECU	Electronic Control Unit
BIST	Built-In Self Tests
CDD	Complex Device Driver
MSTP	Microcontroller Specific Test Package

Table 2.1: Acronyms and abbreviations used in the scope of this Document

3 Related Documentation

[1] ISO 26262-5:2018 – Road vehicles – Functional Safety – Part 5: Product development at the hardware level
https://www.iso.org

[2] Specification of ECU State Manager
AUTOSAR_CP_SWS_ECUStateManager

[3] Specification of Basic Software Mode Manager
AUTOSAR_CP_SWS_BSWModeManager

[4] Specification of MCU Driver
AUTOSAR_CP_SWS_MCUDriver

[5] Specification of Hardware Test Manager on start up and shutdown
AUTOSAR_CP_SWS_HWTestManager

Additional AUTOSAR general specifications shall be considered while implementing HTMSS and associated extension requirements in impacted modules for its compatibilities in AUTOSAR software platform.

4 Objective

Each ECU is designed to provide predefined functionality in the context of a given system architecture. Then it's of great importance that this ECU operates without failures, which in turn can be avoided or detected before they appear, by simple monitoring of expected faults. One strategy to monitor operability of ECU is to execute tests that check given logic and conditions, and keep the results for further analysis.

The HTMSS concept depicts the need to address results from such tests on the ECU, and to provide their status on request.

That set of testing and monitoring activities shall provide the required diagnostic coverage, on potential faults, in order to satisfy the ISO26262 part 5 [1] chapter 8 requirements for hardware architecture metrics.

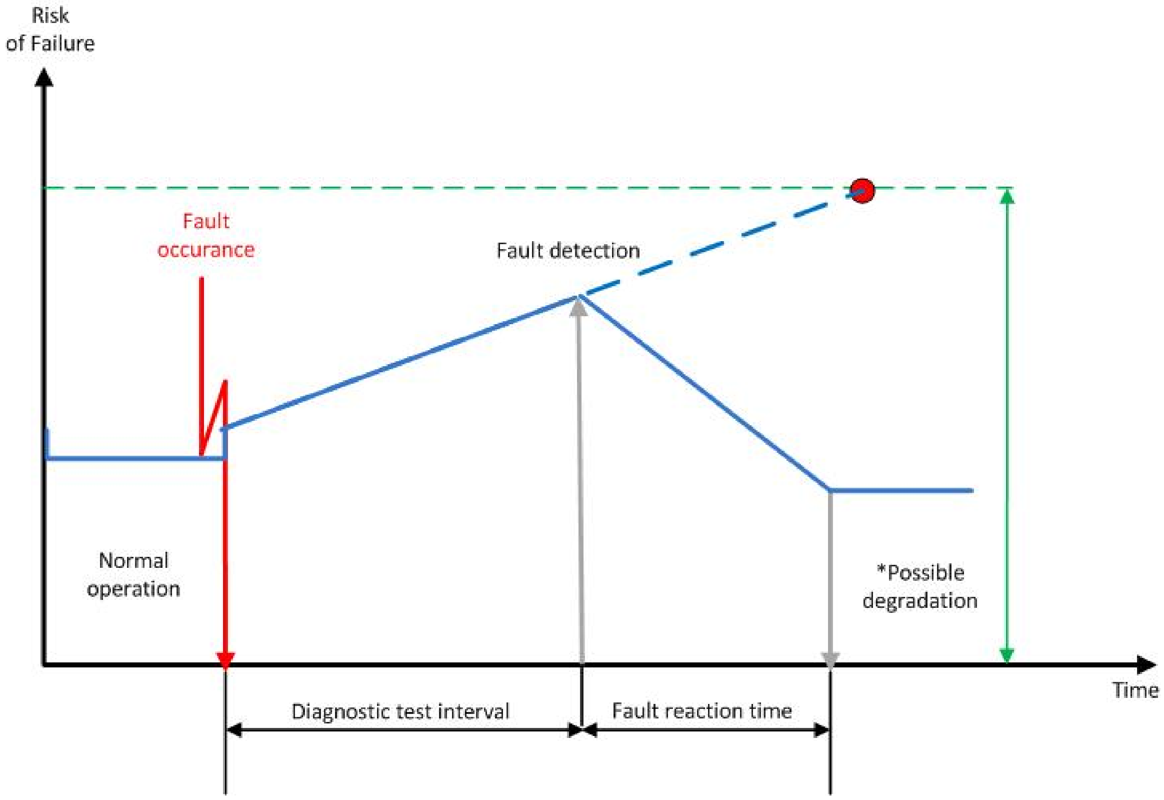


Figure 4.1: Maintenance of safe state

The goal of tests and monitoring facility is to guarantee the fault detection is performed within the predefined time interval (Figure 4.1). After the fault is detected, the responsible software component is informed. This software component shall take the necessary actions; apply a counter-fault reaction in order to maintain the safe state - Figure 4.1.

This document provides the general use cases and requirements to integrate start up and shut down tests in a standard AUTOSAR environment.

5 Goal

During the development for AUTOSAR based systems, a technical approach for integrating the semiconductor-manufacturer specific tests in standard AUTOSAR shall be considered.

The goal is to standardize the accessible interfaces with a microcontroller specific test package (which could be a non-AUTOSAR software module) and to integrate within the AUTOSAR system, which configures the tests, trigger tests execution and collect the test results. In consequence, this concept introduces a BSW module called HTMSS to achieve its functionalities.

6 Motivation

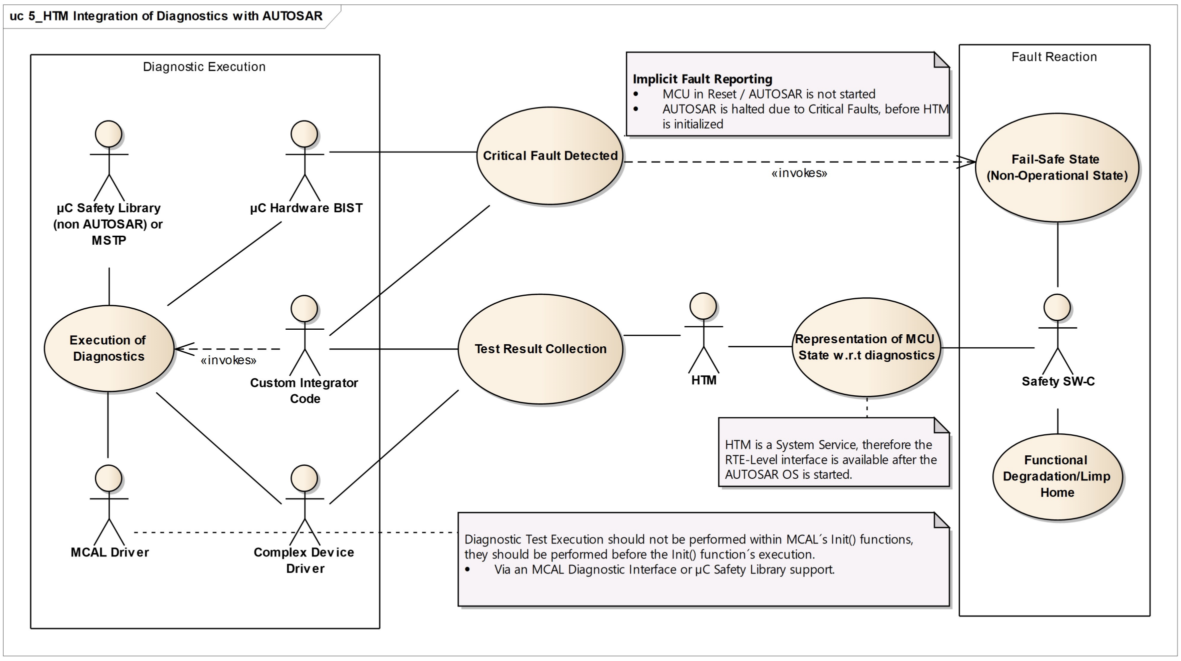
The motivation of HTMSS concept is to support awareness of the system that given functionality is trusted. E.g. resources are available, under operational conditions and no faults were detected. Even if certain faults are detected, still the system may be capable to perform certain activities with trust of their results. The determinism of how the system shall react on these faults has to be implemented based on system analysis and recommendations. The HTMSS concept shall provide support to achieve this awareness. Collecting results from a number of tests can provide facts on conditions and aspects of the resources in the system.

The concept shall meet the following requirements:

- It shall be possible to run tests before Autosar.
- It shall be possible to run tests at Autosar start up and shutdown, and to propagate their results.
- Test results shall be propagated for later analysis

7 Use Case

Diagnostic of potential faults, in the context of a safety critical system, is essential for functional safety goal achievement. It requires integration of tests, their execution and results propagation in parallel to the intended functionality. This implies the necessity to introduce a new component in system service layer, a basic software module called Hardware Test Manager (HTMSS). It implements the tests and monitoring orchestration, and propagates results to stakeholder software components - Figure 7.1.



Implicit Fault Reporting
- MCU in Reset / AUTOSAR is not started
- AUTOSAR is halted due to Critical Faults, before HTM is initialized

HTM is a System Service, therefore the RTE-Level interface is available after the AUTOSAR OS is started.

Diagnostic Test Execution should not be performed within MCAL´s Init() functions, they should be performed before the Init() function´s execution.
- Via an MCAL Diagnostic Interface or µC Safety Library support.

Figure 7.1: Hardware Test Manager - use case concept

The hardware tests shall be executed in harmony with the overall system behaviour. They shall not influence the intended functionality negatively. This is the reason a precise selection of the points they need to be executed should be done. Usually the tests are separate in two basic groups:

- **Non destructive test** - after execution of such test, the item can be transferred back to its previous state (before execution of the test) without a necessity of complete item or system reinitialization.
- **Destructive test** - after execution of such test, the item cannot go back to a known operational state without applying severe initialization procedures for the item or for the complete system.

The impact of detected faults in the system needs to be considered. Some faults (general core fault, RAM test fault), called here critical fault (Figure 7.1), make the system start meaningless. Instead, the MCU can be kept in continuous reset, or another silent mode, where its starting is prohibited. All other detected faults are cognizable by respective application software component, responsible for maintenance of the safe state.

8 Constraints and assumptions

HTMSS goal is to provide necessary environment and infrastructure for collecting and reporting operational status of particular hardware module or periphery. Operational status is result of tests evaluation on these hardware modules and periphery. Basically, there are two types of tests executed in these phases, in respect to their impact on the system/microcontroller - destructive and non-destructive. The concept requires that the used microcontroller has the capability to maintain tests results integrity in a dedicated memory address/register, during the execution of destructive tests. The tests results are accessible for HTMSS. In case of severe failure (core or RAM/ROM failure) of hardware module, detected with, MSTP may take a decision not to continue with further software execution. In that case, the system must go in a safe state. Continuous reset is considered as a safe state. MSTP is responsible for maintaining the safe state. MSTP design specification and implementation are provided by the microcontroller supplier. Tests planned to be executed in AUTOSAR Initialization can be provided either by microcontroller supplier (microcontroller specific tests) or designed and implemented by the system integrator (ECU functional specific tests). They are orchestrated and evaluated by HTMSS itself.

9 HTMSS AUTOSAR integration approach

The hardware test management start up & shutdown proposes the integration of Microcontroller Specific Test Package (MSTP) in the AUTOSAR software environment as follows. The interaction between the standard AUTOSAR modules and MSTP shall be managed by introducing a new module called HTMSS in the BSW service layer.The basic functionalities of HTMSS are:

- Initialization of HTMSS module (including the MSTP module, if needed)
- Interface to configure MSTP tests based on HTMSS module configuration
- Interface for starting the MSTP tests execution
- Collect and provide the MSTP tests results to the required modules and application SWC's for evaluating the results and take relevant decisions.

To fulfill the functional integration certain AUTOSAR std. modules needs to be extended especially with ECU State manager `UP` phase and `DOWN` phase.

Below sections will describe the requirements that need to be considered in the AUTOSAR development process to achieve the functionalities proposed for HTMSS integration.

10 HTMSS Feature Description

This section describes the main feature of hardware test management start up and shutdown, integration in AUTOSAR

[FS_HTMSS_00001] [DRAFT] AUTOSAR shall provide a standardized safety mechanism to integrate microcontroller specific hardware test

Description:	AUTOSAR shall provide a mechanism for collecting the microcontroller specific tests executed during start up and shutdown phases, evaluate the test status and provide it to the stakeholder SW-C.
Rationale:	A failure in the hardware test can lead to a safe state.
Use Case:	e.g. Critical hardware resource test determines the health of the MCU.

11 AUTOSAR Architecture solution

The integration of hardware test management for startup and shutdown requires functional extension of several BSW modules, as well as introduction of a new module "HTMSS" within the BSW layer.

The new module HTMSS shall fulfill the following the functional requirements: it shall interact with the Microcontroller Specific Test Package (here in forth called as MSTP), collect MSTP test results and provide the results to the relevant BSW modules and application SWCs.

The interfaces between HTMSS and MSTP shall be vendor-specific and can be handled via MSTP wrapper implemented within the AUTOSAR development process.

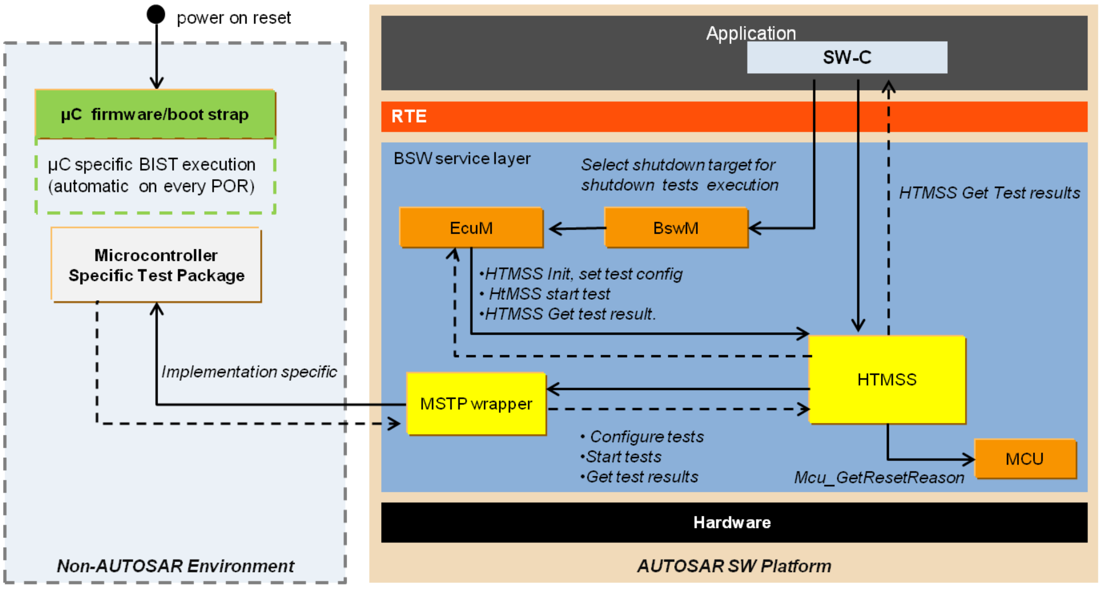


Figure 11.1: HTMSS Overview in AUTOSAR architecture

12 Integration requirements in AUTOSAR SW architecture

This section describes the fundamental requirements that need to be implemented in the affected standard AUTOSAR modules, in order to integrate the HTMSS module and the corresponding Microcontroller Specific Test Package in the AUTOSAR architecture.

The following AUTOSAR modules are affected in this context:

- ECU State Manager [2] - Extension of EcuM UP and DOWN phase
- BSW Mode Manager [3] - Extension of shutdown target
- MCU driver [4] - Extension of reset reason

12.1 ECU state manager

The EcuM modules needs to be extended as follows to incorporate HTMSS in the AUTOSAR software environment:

The extension is needed to fulfill below proposed functional integration approach

- The EcuM START UP phase shall prepare the HTMSS & MSTP module and perform the start up test execution.
- The ECUM DOWN phase shall integrate the HTMSS shutdown tests within the shutdown target sequence flow triggered from BswM.

The detailed requirements for EcuM are described in the below sections.

12.1.1 General requirements

[SWS_EcuM_04136_EXTENSION]
The EcuM_ShutdownTargetType shall be extended with ECUM_SHUTDOWN_HWTEST_RESET and ECUM_SHUTDOWN_HWTEST_OFF to handle the reset caused by shutdown test execution.

Name	EcuM_ShutdownTargetType		
Kind	Type		
Derived from	uint8		
Range	ECUM_SHUTDOWN_TARGET_SLEEP	0x0	–
	ECUM_SHUTDOWN_TARGET_RESET	0x1	–
	ECUM_SHUTDOWN_TARGET_OFF	0x2	–
	ECUM_SHUTDOWN_HWTEST_RESET	**0x3**	–
	ECUM_SHUTDOWN_HWTEST_OFF	**0x4**	**–**

▽

△

Description	–
Variation	–
Available via	Rte_EcuM_Type.h

12.1.2 During EcuM START UP PHASE:

[SWS_EcuM_HTMSS_00001]
In the Init block 1, EcuM shall call HTMSS_Init to initialize the HTMSS module. (Please refer to [5] Section 9.1.1)

[SWS_EcuM_HTMSS_00002]
The ECU manager module shall call HTMSS_StartTest to trigger the MSTP start up test execution based on return value of Mcu_GetResetReason API. (Please refer to [5] Section 9.1.2)

[SWS_EcuM_HTMSS_00003]
The ECU manager module shall call HTMSS_GetTestStatus to collect the MSTP start up test results or shutdown test results based on return value of Mcu_GetResetReason API. (Please refer to [5] Section 9.1.2 and 9.1.5)

[SWS_EcuM_HTMSS_00004]
The ECU manager module shall call HTMSS_StartupTestErrorHook in case the function HTMSS_GetTestStatus returns HTMSS_STATUS_NOK. (Please refer to [5] Section 9.1.2)

12.1.3 During EcuM SHUTDOWN PHASE:

[SWS_EcuM_HTMSS_00005]
The ECU manager module shall call the HTMSS_StartTest service function to trigger the MSTP shutdown test execution based on EcuM_ShutdownTarget. (Please refer to [5] Section 9.1.3)

[SWS_EcuM_HTMSS_00006]
The ECU manager module shall call HTMSS_GetTestStatus based on the Mcu_ResetType. (Please refer to [5] Section 9.1.4 and 9.1.5)

Hint: Normally shutdown test execution causes a hardware reset. After this reset and in the EcuM_Init the Mcu_GetResetReason will be called by EcuM. If the reset reason is MCU_HWTEST_RESET then EcuM shall call HTMSS_GetTestStatus to collect the shutdown test results.

[SWS_EcuM_HTMSS_00007]
The ECU manager module shall call HTMSS_ShutdownTestErrorHook in case the function HTMSS_GetTestStatus returns HTMSS_STATUS_NOK. (Please refer to [5] Section 9.1.5)

12.1.4 Example sequence diagrams for HTMSS integration in ECUM:

The below sequence diagrams shall be referred to integrate HTMSS in EcuM UP and DOWN phase.

[SWS_EcuM_HTMSS_00008]
Please refer to [5] Chapter 9.1.1 for HTMSS init function integration in EcuM.

[SWS_EcuM_HTMSS_00009]
Please refer to [5] Chapter 9.1.2 for HTMSS start up test integration in EcuM.

[SWS_EcuM_HTMSS_00010]
Please refer to [5] Chapter 9.1.3 for HTMSS shutdown test execution integration in EcuM.

[SWS_EcuM_HTMSS_00011]
Please refer to [5] Chapter 9.1.4 and 9.1.5 to collect the last shutdown test results for application usage.

[SWS_EcuM_HTMSS_00012]
Please refer to [5] Chapter 9.1.6 to integrate the shutdown tests execution in the EcuM shutdown phase.

12.2 BSW mode manager

The BswMEcuMSelectShutdownTarget shall be extended with HWTEST_OFF and HWTES_RESET to handle the reset caused by shutdown test execution.

[ECUC_BswM_00993_EXTENSION] Definition of EcucEnumerationParamDef BswMEcuMShutdownTarget

Parameter Name	BswMEcuMShutdownTarget	
Parent Container	BswMEcuMSelectShutdownTarget	
Description	This parameter contains the shutdown target that the BswM selects at the EcuM.	
Multiplicity	1	
Type	EcucEnumerationParamDef	
Range	BSWM_ECUM_SHUTDOWN_TARGET_OFF	–
	BSWM_ECUM_SHUTDOWN_TARGET_RESET	In case the configuration parameter BswMEcuMShutdownTarget is set to BSWM_ECUM_SHUTDOWN_TARGET_RESET the configuration parameter BswMEcuMResetModeRef shall exist and contain a valid reference to a EcuM reset mode.

▽

△

	BSWM_ECUM_SHUTDOWN_TARGET_SLEEP	In case the configuration parameter BswMEcuMShutdownTarget is set to BSWM_ECUM_SHUTDOWN_TARGET_SLEEP the configuration parameter BswMEcuMSleepModeRef shall exist and contain a valid reference to a EcuM sleep mode.	
	HWTEST_OFF	**In case the configuration parameter BswMEcuMShutdownTarget is set to HWTEST_OFF the configuration parameter BswMEcuMResetModeRef shall exist and contain a valid reference to a EcuM shutdown hardware test OFF mode.**	
	HWTEST_RESET	**In case the configuration parameter BswMEcuMShutdownTarget is set to HWTEST_RESET the configuration parameter BswMEcuMResetModeRef shall exist and contain a valid reference to a EcuM shutdown hardware test RESET mode.**	
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME, VARIANT-POST-BUILD
	Post-build time	–	
Scope / Dependency	scope: local		

12.3 MCU driver

[SWS_Mcu_00252_EXTENSION] The `Mcu_ResetType` shall be extended with `MCU_HWTEST_RESET` to handle the reset caused by shutdown test execution.

Name	Mcu_ResetType		
Kind	Enumeration		
Range	MCU_POWER_ON_RESET	0x00	Power On Reset (default)
	MCU_WATCHDOG_RESET	0x01	Internal Watchdog Timer Reset
	MCU_SW_RESET	0x02	Software Reset
	MCU_HWTEST_RESET		**Reset caused by shutdown tests**
	MCU_RESET_UNDEFINED	0x03	Reset is undefined
Description	This is the type of the reset enumerator containing the subset of reset types. It is not required that all reset types are supported by hardware.		
Available via	Mcu.h		

13 Impact on performance and software behaviour in AUTOSAR

The integration of HTMSS in AUTOSAR has an impact on the ECUM Start-up and Shutdown behaviour in ECUs. The consequence of this would be

- Additional time required to complete the HTMSS functionalities during the respective EcuM phases (e.g. longer EcuM initialization phase, longer EcuM shutdown phase)
- The ECU startup sequence may be aborted in case critical faults (i.e. when MSTP test status is judged as a critical fault) are detected.

Thus, the integrator is free to decide on the HTMSS integration needs in AUTOSAR, which is proposed as an optional feature that complies with the AUTOSAR software environment.