**RESTRICTED ACCESS**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **DISTRIBUTION :** | **Firm** | **To** | **Ref** | **Copies** | **1st page** | **e-mail** |
|  | AEE-C | Xavier Christmann, Yannick Erb, Samuel Delcloy, Jerome Lacroix, Maxime Bourdon, Marion Rakotonoelina, Francine Chambin, Mamadou Diallo | 8 |  |  |  |
|  | AEE-C | Cedric Sauvage, Pierre-Olivier Pilot, Sabine Flechelle, Sebastien Claveau, Audrey Vaché, M. Pastor, N. Bianchi, C. Dibourg, A. Dorel, Khaled Mohrath | 10 |  |  |  |
|  | AEE-T | S. Papadineti, A. Mestereaga, S. Dragan | 3 |  |  |  |
|  |  |  |  |  |  |  |
|  | AEE-C | Secretary ship | 1 |  |  |  |

**SW Architecture Design & Interface Description :**

**ERH sw UNIT**

OBJECT: This document is the description of the design & interfaces for *ERH* SW unit.

SUMMARY: This document provides a high-level view of the *ERH* SW unit. The inputs of this document are provided by the software requirement. It is linked to the E1355904\_SWarchitectureDesignInterfaceDescription document.

CONCLUSION: Applicable from R3.0M/E SW release

**THIS DOCUMENT CONTAINS HIDDEN TEXT**

EVOLUTION OF THE DOCUMENT

|  |  |  |  |
| --- | --- | --- | --- |
| **Issue** | **Date** | **Author** | **Motive and nature of the modifications** |
| A00 | 31/08/2016 | C. Redon | First release (extract from the full PP4G architecture document) |
|  |  |  |  |

This document contains **14** pages.

Peer Review associated to this document: “*ERH -* Peer Review Workbook.xlsm” document located in \Quality\_Assurance\Peer\_Review

**CONTENTS**

1. Documentation 5

1.1. Upper Level Relevant Documents 5

1.2. Design Specification Documents 6

1.3. Freescale Documents 7

1.4. Tier2 Documents 8

1.5. HW Datasheet 8

1.6. Other Documents 8

1.7. Glossary And Definition 9

2. Description 10

3. Technical Functions 10

4. Runnables 10

4.1. ERH\_Init 10

4.1.1. Definition 11

4.1.2. Data flow / Parameters 11

4.1.3. Called functions 11

4.2. ERH\_StoreEverQualifiedAECsToNVP 11

4.2.1. Definition 11

4.2.2. Data flow / Parameters 11

4.2.3. Called functions 12

4.3. ERH\_runGetAecStatus 12

4.3.1. Definition 12

4.3.2. Data flow / Parameters 12

4.3.3. Called functions 12

4.4. ERH\_runSetAecEvent 12

4.4.1. Definition 12

4.4.2. Data flow / Parameters 13

4.4.3. Called functions 13

4.5. ERH\_runGetAecGroupsStatus 13

4.5.1. Definition 13

4.5.2. Data flow / Parameters 13

4.5.3. Called functions 13

4.6. ERH\_runGetAecNvpBlockRecordingStatus 13

4.6.1. Definition 14

4.6.2. Data flow / Parameters 14

4.6.3. Called functions 14

5. MCU resources 14

**TABLE OF FIGURES**

[**Figure 1: Erh - Static description** 10](#_Toc461189155)

# Documentation

## Upper Level Relevant Documents

This section presents all the documents needed to write the software architecture design document.

|  |  |  |  |
| --- | --- | --- | --- |
| Nb | **Document** | **Reference** | **Company** |
|  | TF-A : To Manage the power supply | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_A\_To\_Manage\_The\_Power\_Supply | AEE-C / ALE  PP Platform |
|  | TF-B : To Manage the communication | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_B\_To\_Manage\_The\_Communication | AEE-C / ALE  PP Platform |
|  | TF-C : To Secure PP ECU functioning using Pictus MCU | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_C\_To\_Secure\_PP\_ECU\_Functioning\_Pictus | AEE-C / ALE  PP Platform |
|  | TF-E : To Manage Diagnostic Requests | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_E\_To\_Manage\_Diagnostic\_Requests | AEE-C / ALE  PP Platform |
|  | TF-F : To Perform Measurements | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_F\_To\_Perform\_Measurements | AEE-C / ALE  PP Platform |
|  | TF-G : To Drive the Motor | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_G\_To\_Drive\_the\_Motor | AEE-C / ALE  PP Platform |
|  | TF-H : To Perform Autotests | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_H\_To\_Perform\_Autotests | AEE-C / ALE  PP Platform |
|  | TF-I : To Manage the Failure | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_I\_To\_Manage\_The\_Failure | AEE-C / ALE  PP Platform |
|  | TF-J : To Manage NVM - NVP (Non Volatile Parameters) | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_J\_To\_Manage\_NVM | AEE-C / ALE  PP Platform |
|  | TF-N : To evaluate belt data | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_N\_To\_Evaluate\_Belt\_Data | AEE-C / ALE  PP Platform |
|  | TF-O : To schedule the SW | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_O\_To\_Run\_SW | AEE-C / ALE  PP Platform |
|  | TF-Q : To Provide Data For Expertise | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_Q\_To\_Provide\_Data\_For\_Expertise | AEE-C / ALE  PP Platform |
|  | TF-R : To Decide Belt Function Execution | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_R\_To\_Decide\_Belt\_Function\_Execution | AEE-C / ALE  PP Platform |
|  | TF-D : To Program MCU | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_D\_To\_Program\_MCU | AEE-C / ALE  PP Platform |
|  | TF-P : To handle network management | /Platform\_PP\_ECU\_PP\_4G/30\_DES\_Requirements/Technical Functions  DES\_TF\_P\_To Handle\_Network\_Management | AEE-C / ALE  PP Platform |

## Design Specification Documents

This section presents all the documents that complete this software architecture design document.

|  |  |  |  |
| --- | --- | --- | --- |
| Nb | **Document** | **Reference** | **Company** |
|  | SW Architecture Design Interface Description | [E1355904\_SWarchitectureDesignInterfaceDescription.docx](../../../Architecture/Document/E1355904_SWarchitectureDesignInterfaceDescription.docx) | AEE-C / ALE |
|  | Design document of Analog Digital Conversion | ADC - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Auto Tests Manager | ATM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Belt Function Execution | BFE - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Belt Function Selection | BFS - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Belt Movement Monitoring | BMM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Belt Parking Algorithm | BPA - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Belt Slack Reduction | BSR - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of CAN | CAN - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Communication Interaction Layer | CIL - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of DiagOnCAN services management | DIA - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Digital Input Output | DIO - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of End of life | EOL - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Error Handler | ERH - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of ECU State Manager | ESM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Flash EEPROM Emulation | FEE - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Flash Driver | FLS - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Haptic Warning | HWA - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Logical Supply Monitoring | LSM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Micro Controler Unit | MCU - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Memory Integrity Control | MIC - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Mode Management | MMG - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Non-Volatile Parameters | NVP - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Power Abstraction Layer | PAL - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Pre-Crash Master | PCM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Physical Measures Provider | PMP - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Pre Pre-Tensioning | PRE - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Production cycle function | PRO - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Port Driver | PRT - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Pulse Width Modulation | PWM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Reset Cause Management | RCM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Scheduler | SCH - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of System Context Management | SCM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Standard Function Recovery (releasing function) | SFR - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of System Time Management | STM- Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Timer | TIM - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Transceiver | TRV - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Vehicle Dynamics algorithm | VDA - Detailed Design Document.docx | AEE-C / ALE |
|  | Design document of Watchdog | WDG - Detailed Design Document.docx | AEE-C / ALE |
|  | EEPROM parameters implementation | SBE\_4G\_NVP\_layout.xls | AEE-C / ALE |

## Freescale Documents

This section presents all the documents that complete this software architecture design document.

|  |  |  |  |
| --- | --- | --- | --- |
| Nb | **Document** | **Reference** | **Company** |
|  | MC9S12ZVC-Family Reference Manual Preliminary  Confidential | MC9S12ZVCRM\_Rev0.06.pdf | Freescale |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Tier2 Documents

This section presents all the documents that complete this software architecture design document.

|  |  |  |  |
| --- | --- | --- | --- |
| Nb | **Document** | **Reference** | **Company** |
|  | User manual FlashBootloader | UserManual\_FlashBootloader.pdf | Vector |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## HW Datasheet

This section presents all the documents related to the HW components that complete this software architecture design document.

|  |  |  |  |
| --- | --- | --- | --- |
| Nb | **Document** | **Reference** | **Company** |
|  | BTN8984TA datasheet | BTN8984TA\_TDS\_051 | Infineon |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Other Documents

This section presents all the documents that also have been needed to write this software architecture design document.

|  |  |  |  |
| --- | --- | --- | --- |
| Nb | **Document** | **Reference** | **Company** |
|  | Unified Modelling Language | 2.1.1 | OMG |
|  | MCU RFQ | [E1244926](http://plm.autoliv.int/linkto/latestreleased/ProductDescription/E1244926/*) | ALE |
|  |  |  |  |

## Glossary And Definition

This section presents all the definitions and/or abbreviations used in this document.

*List of terms in alphabetical order:*

|  |  |
| --- | --- |
| ***Term*** | ***Meaning*** |
| ADC | Analog Digital Converter |
| AEC | Autoliv Error Code |
| API | Application Programming Interface |
| ASDM | Active Safety Domain Master |
| ASIC | Application Specific Integrated Circuit |
| ASY | Active SafetY |
| BSW | Basic SW modules |
| CAN | Controller Area Network |
| C/S | Chip Select |
| COP | Computer Operating Properly |
| eCPL | Electronic Crash Pole Locking |
| DART | Ditch - Airborne - Rough Terrain |
| DFLASH | Data FLASH |
| ECC | Error Code Correction |
| ECU | Electronic Control Unit |
| EOL | End Of Life |
| EEPROM | Electric Erasable and Programmable Read only Memory |
| HFPP | High Force Pre-Pre-Tensioning belt function |
| HF-PRE | High Force PRE pre-tensioning |
| HR | Hard Releasing |
| I/O | Input/Output |
| IMU | Inartial Measurements Unit |
| ISS | Integrated Safing System |
| LFPP | Low Force Pre-Pre-Tensioning belt function |
| MSA | Motor Start/Stop Automatic |
| MCAL | Micro-Controller Abstraction Layer |
| MCU | Micro-controller Unit |
| NMG | Mode ManaGement |
| NVM | Non Volatile Memory |
| OS | Operating System |
| PCM | Pre-Crash Master |
| PFLASH | Program FLASH |
| PIT | Periodic Interrupt Timer |
| PLL | Phase-locked loop |
| RAM | Random Access Memory |
| RCWM | Rear Collision Warning and Mitigation |
| RML | Left PP ECU |
| RMR | Right PP ECU |
| RMx | Both PP ECU |
| ROM | Read Only Memory |
| RSU | Remote Sensor Unit |
| RTE | Real Time Environment |
| RTOS | Real Time Operating System |
| SFR | Standard Function Recovery |
| SODL | Side Obstacle Detection Left |
| SPI | Serial Peripheral Interface |
| SRS | Supplementary Restraint System |
| TBC | To be confirmed |
| TBD | To be defined |
| TF | Technical Function |
| TFLASH | Test FLASH of the Pictus MCU (“one time programmable” memory) |
| W/D | Watchdog |

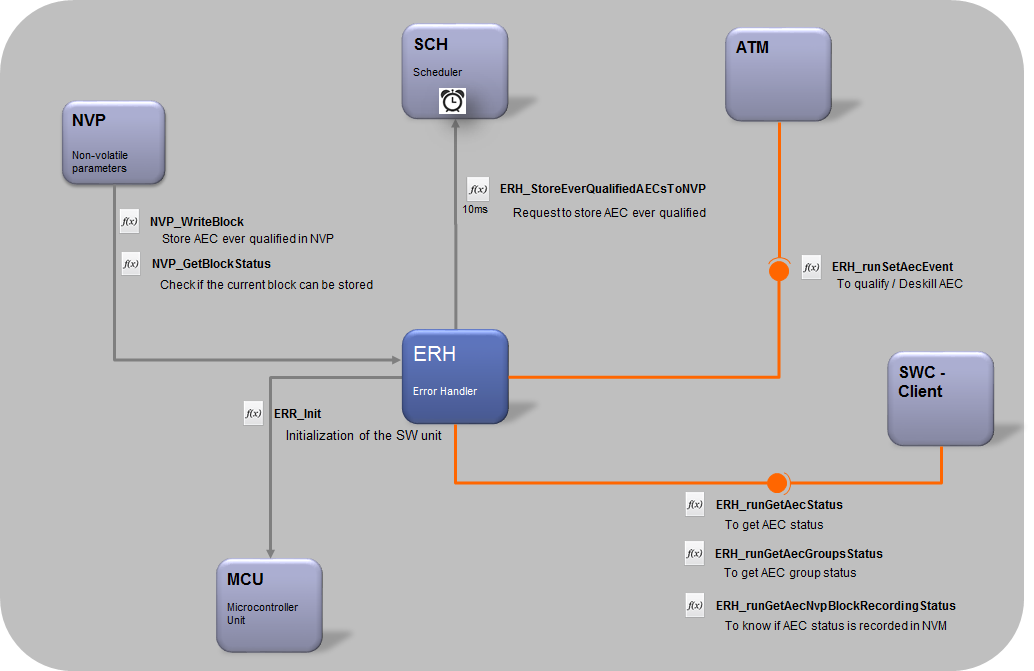
# Description

ERH is a generic error handler developed by AUTOLIV. The objective is to handle error notification transmitted by ATM in filtering error qualification / deskilling (based on thresholds recorded in NVM).

“Error” will be so called AEC: Autolic Error Code.

AEC will be handled in 2 different manners:

* Time-based: An internal timer is managed for each AEC (Autoliv Error Code). A timer will be triggered on when an event is received and triggered off when an AEC qualification time is reached.
* Counting-based: An internal counter is managed for each AEC. An associated counter is incremented / decremented depending on the state of the received events which can respectively failed or passed.



**Figure 1: Erh - Static description**

# Technical Functions

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirements** | **Criteria** | **Levels/Tolerances** | **Source** |
| ARCH\_SW\_ERH\_0000 | This component shall implement the TF-I technical function |  |  |

# Runnables

## ERH\_Init

### Definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Prototype** | | | |
| void **ERH\_Init** (void) | | | |
| **Object** | | | |
| This function aim at initializing all the AEC status and associated counters. | | | |
| **Parameters** | | | |
| Name | Type | Direction | Description |
| NA | NA | NA | NA |
| **Returned value** | | | |
| Name | Description | | |
| NA | NA | | |
| **Dynamic aspect** | | | |
| Synchronous server operation  Reentrant | | | |
| **Requirements** | | | |
| ARCH\_SW\_ERH\_0020 | | | |
| **Covered requirements** | | | |
|  | | | |

### Data flow / Parameters

NA

### Called functions

NA

## ERH\_StoreEverQualifiedAECsToNVP

### Definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Prototype** | | | |
| void **ERH\_StoreEverQualifiedAECsToNVP** (void) | | | |
| **Object** | | | |
| This function shall manage the recording of newest qualified AEC (Autoliv Error Code). | | | |
| **Parameters** | | | |
| Name | Type | Direction | Description |
| NA | NA | NA | NA |
| **Returned value** | | | |
| Name | Description | | |
| NA | NA | | |
| **Dynamic aspect** | | | |
| Synchronous server operation  Reentrant  Called every 10ms | | | |
| **Requirements** | | | |
| ARCH\_SW\_ERH\_0025 | | | |
| **Covered requirements** | | | |
|  | | | |

### Data flow / Parameters

NA

### Called functions

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirements** | **Criteria** | **Levels/Tolerances** | **Source** |
| ARCH\_SW\_ERH\_0026 | [NVP\_WriteBlock](#_Hlk448222329) shall be called to store the AEC ever qualified on request. | Failed writing in NVP must be checked for a next retry. |  |
| ARCH\_SW\_ERH\_0027 | [NVP\_GetBlockStatus](#_Hlk448222379) shall be called to check the status of the writing request. | Pending state must be returned if a writing process is already being processed.s |  |

## ERH\_runGetAecStatus

### Definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Prototype** | | | |
| void **ERH\_runGetAecStatus** (u8AecIdentifierType u8AecIdentifier, u8AecStatusType \* pu8AecStatus) | | | |
| **Object** | | | |
| This function shall provide the AEC status if the AEC identifier is valid. | | | |
| **Parameters** | | | |
| Name | Type | Direction | Description |
| u8AecIdentifier | u8AecIdentifierType | Input | AEC identifier |
| pu8AecStatus | u8AecStatusType | Output | AEC status type :  - ERH\_KU8\_AEC\_QUALIFIED\_STATUS\_MASK  - ERH\_KU8\_AEC\_EVER\_QUALIFIED\_STATUS\_MASK  - ERH\_KU8\_AEC\_EVER\_TESTED\_STATUS\_MASK  - ERH\_KU8\_AEC\_IN\_PROGRESS\_STATUS\_MASK  - ERH\_KU8\_AEC\_FAILED\_AT\_LEAST\_ONCE\_MASK  - ERH\_KU8\_AEC\_UNPLAUSIBLE\_STATUS |
| **Returned value** | | | |
| Name | Description | | |
| NA | NA | | |
| **Dynamic aspect** | | | |
| Synchronous server operation  Reentrant | | | |
| **Requirements** | | | |
| ARCH\_SW\_ERH\_0035 | | | |
| **Covered requirements** | | | |
|  | | | |

### Data flow / Parameters

NA

### Called functions

NA

## ERH\_runSetAecEvent

### Definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Prototype** | | | |
| void **ERH\_runSetAecEvent** (u8AecIdentifierType u8AecIdentifier, u8AecCommandType u8Command) | | | |
| **Object** | | | |
| This function shall update the AEC status based on a failed / passed test result. | | | |
| **Parameters** | | | |
| Name | Type | Direction | Description |
| u8AecIdentifier | u8AecIdentifierType | Input | AEC Identifier value |
| u8Command | u8AecCommandType | Input | Type of command :  - KU8\_AEC\_FAILED  - KU8\_AEC\_PASSED  - KU8\_AEC\_RST\_COUNTER  - KU8\_AEC\_RST\_ALL |
| **Returned value** | | | |
| Name | Description | | |
| NA | NA | | |
| **Dynamic aspect** | | | |
| Synchronous server operation  Reentrant | | | |
| **Requirements** | | | |
| ARCH\_SW\_ERH\_0040 | | | |
| **Covered requirements** | | | |
|  | | | |

### Data flow / Parameters

NA

### Called functions

NA

## ERH\_runGetAecGroupsStatus

### Definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Prototype** | | | |
| void **ERH\_runGetAecGroupsStatus** (u16AecGroupsStatusType \* pu16AecGroupsStatus) | | | |
| **Object** | | | |
| This function shall provide the AEC group status. | | | |
| **Parameters** | | | |
| Name | Type | Direction | Description |
| pu16AecGroupsStatus | u16AecGroupsStatusType | Out | AEC group status |
| **Returned value** | | | |
| Name | Description | | |
| NA | NA | | |
| **Dynamic aspect** | | | |
| Synchronous server operation  Reentrant | | | |
| **Requirements** | | | |
| ARCH\_SW\_ERH\_0060 | | | |
| **Covered requirements** | | | |
|  | | | |

### Data flow / Parameters

NA

### Called functions

NA

## ERH\_runGetAecNvpBlockRecordingStatus

### Definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Prototype** | | | |
| void **ERH\_runGetAecNvpBlockRecordingStatus** (b8BooleanType \* b8AecNvpBlockRecordingStatus) | | | |
| **Object** | | | |
| This function shall provide the status of the current NVP recording process. | | | |
| **Parameters** | | | |
| Name | Type | Direction | Description |
| pu16AecGroupsStatus | u16AecGroupsStatusType | Out | AEC group status |
| **Returned value** | | | |
| Name | Description | | |
| NA | NA | | |
| **Dynamic aspect** | | | |
| Synchronous server operation  Reentrant | | | |
| **Requirements** | | | |
| ARCH\_SW\_ERH\_0080 | | | |
| **Covered requirements** | | | |
|  | | | |

### Data flow / Parameters

NA

### Called functions

NA

# MCU resources

The following requirements on resource consumption objectives apply to the module/package:

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirements** | **Criteria** | **Levels/Tolerances** | **Source** |
| ARCH\_SW\_ERH\_9997 | The ROM size consumed by this component shall not exceed 1.5K bytes. |  |  |
| ARCH\_SW\_ERH\_9998 | The heap size consumed by this component shall be 500 bytes. |  |  |