Elektrobit

Eep release notes and documentation

product release 8.8.3

Elektrobit Automotive GmbH
Am Wolfsmantel 46
91058 Erlangen, Germany
Phone: +49 9131 7701 0
Fax: +49 9131 7701 6333
Email: info.automotive@elektrobit.com

Technical support

https://www.elektrobit.com/support

Legal disclaimer

Confidential information.

Table of Contents

Preface 5
1. Overview 5
1. Eep release notes 6
1.1. New features 6
1.2. Migrating the Eep module 6
1.3. Limitations and deviations 6
1.4. EB-specific enhancements 6
1.5. Change log 7
2. Eep user's guide 9
2.1. Overview 9
2.2. Configuring EEP Driver and related modules 9
2.2.1. Configuring EEP Driver 9
2.2.2. Configuring the SchM 9
2.2.3. Configuring the Dem 9
2.3. Writing application software 10
2.4. Compiling EEP Driver 10
2.5. Running EEP Driver 10
3. Eep references 11
3.1. Configuration parameters 11
3.1.1. CommonPublishedInformation 11
3.1.2. EepGeneral 14
3.1.3. EepInitConfiguration 17
3.1.4. EepDemEventParameterRefs 20
3.1.5. EepExternalDriver 22
3.1.6. EepPublishedInformation 22
3.1.7. PublishedInformation 26
3.2. Application programming interface (API) 26
3.2.1. Macro constants 26
3.2.1.1. EEP_E_BUSY 26
3.2.1.2. EEP_E_PARAM_ADDRESS 27
3.2.1.3. EEP_E_PARAM_CONFIG 27
3.2.1.4. EEP_E_PARAM_DATA 27
3.2.1.5. EEP_E_PARAM_LENGTH 27
3.2.1.6. EEP_E_UNINIT 27
3.2.1.7. EEP_INSTANCE_ID 28
3.2.1.8. EEP_INTERNAL_API_ID 28
3.2.1.9. EEP_SID_CANCEL 28
3.2.1.10. EEP_SID_COMPARE 28
3.2.1.11. EEP_SID_ERASE 28

3.2.1.12. EEP_SID_GET_JOB_RESULT 28
3.2.1.13. EEP_SID_GET_STATUS 29
3.2.1.14. EEP_SID_GET_VERSION_INFO 29
3.2.1.15. EEP_SID_INIT 29
3.2.1.16. EEP_SID_MAIN_FUNCTION 29
3.2.1.17. EEP_SID_READ 29
3.2.1.18. EEP_SID_SET_MODE 30
3.2.1.19. EEP_SID_WRITE 30
3.2.2. Functions 30
3.2.2.1. Eep_Cancel 30
3.2.2.2. Eep_Compare 30
3.2.2.3. Eep_Erase 31
3.2.2.4. Eep_GetJobResult 31
3.2.2.5. Eep_GetStatus 32
3.2.2.6. Eep_GetVersionInfo 32
3.2.2.7. Eep_Init 33
3.2.2.8. Eep_MainFunction 33
3.2.2.9. Eep_Read 34
3.2.2.10. Eep_SetMode 34
3.2.2.11. Eep_Write 34
4. Bibliography 36

Preface

1. Overview

Welcome to the EEP MCAL release notes and documentation. This document provides:

- Chapter 1, “Eep release notes“, which include
  - Section 1.1, “New features”
  - Section 1.2, “Migrating the Eep module”
  - Section 1.3, “Limitations and deviations ”
  - Section 1.4, “EB-specific enhancements”
  - Section 1.5, “Change log”
- Chapter 2, “Eep user's guide“: concept information and configuration instructions
- Chapter 3, “Eep references“: configuration parameters and the application programming

1. Eep release notes

- AUTOSAR version and revision: 4.0.0
- AUTOSAR SWS version and revision: 3.0.0
- Module version: 1.0.9
- Supplier: Elektrobit Automotive GmbH

1.1. New features

- No new features have been added since the last release.

1.2. Migrating the Eep module

This chapter describes how to migrate a module from one release to an other.

If you want to migrate from one release to another, follow the installation instructions according to EB tresos installation guide.

- The module does not need to be migrated. Reason: the AUTOSAR release has not changed since the last EB tresos AutoCore release.

1.3. Limitations and deviations

This chapter lists the limitations of the module and its deviations from the AUTOSAR standard.

- Configuration parameter EepUseInterrupts is not supported.
- Configuration parameter EepWriteCycleReduction is not supported.
- Configuration parameter EepBaseAddress is not supported.
- Configuration parameter EepJobCallCycle is not supported.
- Internal buffering is performed during compare jobs.

1.4. EB-specific enhancements

This chapter list the enhancements provided by the module.

- No enhancements in this release.

1.5. Change log

This chapter lists the changes between different versions.

Module version 1.0.9

2021-06-25

- Internal module improvement. This module version update does not affect module functionality.

Module version 1.0.8

2021-03-05

- Internal module improvement. This module version update does not affect module functionality.

Module version 1.0.7

2020-10-23

- Internal module improvement. This module version update does not affect module functionality.

Module version 1.0.6

2020-06-19

- Internal module improvement. This module version update does not affect module functionality.

Module version 1.0.5

2017-10-18

- EBAMCALWINCORE-165 Changed DEM error IDs to Aurosar 4.x format.

Module version 1.0.4

2013-10-15

- EBAMCALWINCORE-130 Added support for Debug and Trace.
- EBAMCALWINCORE-135 Corrected paths in BSWMD.

Module version 1.0.3

2013-06-21

- EBAMCALWINCORE-107 Added support for MemMap generator.
- EBAMCALWINCORE-111 Unified syntax of include directives.
- EBAMCALWINCORE-94 Fixed online documentation.

Module version 1.0.2

2012-07-16

- EBAMCALWINCORE-74 Eep file is written at unpredictable points in time

Module version 1.0.1

2012-06-15

- Initial AUTOSAR 4.0 version
- EBAMCALWINCORE-55 Added defines for exported symbols according to new naming scheme.
- EBAMCALWINCORE-75 Corrected module classification.
- EBAMCALWINCORE-78 Added default values to configuration scheme.

Module version 1.0.0

2012-03-02

2. Eep user's guide

2.1. Overview

This users guide provides information which is specific to EB tresos WinCore `EEP Driver`. After following the instructions given in this document, you will be able to use EB tresos WinCore `EEP Driver` to read, write, erase, compare into EEPROM address space. The EEPROM address space is emulated in MS Windows binary file. The location and name of this file is configurable. EB tresos WinCore `EEP Driver` behaves as a standard AUTOSAR EEP Driver and acts as MCAL for the upper layer modules of the memory stack. It thus adds full EEP capabilities to EB tresos WinCore.

2.2. Configuring EEP Driver and related modules

2.2.1. Configuring EEP Driver

The module implementation adds two vendor specific parameters that specify location and filename of the binary file where EEPROM address space is emulated, those parameters are: EepFilePath, EepFileName. The appropriate default values are provided in module schema.

The module schema also provides minimal recommended configuration called "EepRecMinimal" that can be directly used by user. The module is implemented in post-build configuration variant.

2.2.2. Configuring the SchM

There are no critical sections required by the module implementation.

2.2.3. Configuring the Dem

EB tresos WinCore `EEP Driver` reports certain run-time errors to the `Diagnostic Event Manager` (`Dem`). Thus, add a new Dem event called `EEP_E_WRITE_FAILED`, `EEP_E_READ_FAILED`, `EEP_E_ERASE_FAILED`, `EEP_E_COMPARE_FAILED` to the Dem configuration.

2.3. Writing application software

There are no special requirements. The modules behaves as specified in SWS document.

2.4. Compiling EEP Driver

There are no special requirements. Additionaly to SWS document that module requires functionality contained in <stdio.h> library (file operations).

2.5. Running EEP Driver

To successfully run EB tresos WinCore `EEP Driver`, follow this hints:

1. The user of Eep module has to ensure that after call of Eep_Init() the module's status is set to MEMIF_-IDLE and job's result to MEMIF_JOB_OK, this indicates that Eep module were able to either create or re-open the binary file where EEPROM address space is emulated. If this step failed after call of Eep_Init() the module's status is set to MEMIF_BUSY and job's result to MEMIF_JOB_OK this serves as indication that Eep module can not accept any job.

3. Eep references

3.1. Configuration parameters

Containers included		
Container name	**Multiplicity**	**Description**
CommonPublishedInformation	1..1	**Label:** Common Published Information Common container, aggregated by all modules. It contains published information about vendor and versions.
EepGeneral	1..1	Container for general configuration parameters of the EEP-ROM driver. These parameters are always pre-compile.
EepInitConfiguration	1..n	Container for runtime configuration parameters of the EEP-ROM driver.
EepPublishedInformation	1..1	Additional published parameters not covered by Common-PublishedInformation container.
PublishedInformation	1..1	**Label:** EB Published Information Additional published parameters not covered by Common-PublishedInformation container.

Parameters included	
Parameter name	**Multiplicity**
IMPLEMENTATION_CONFIG_VARIANT	1..1

Parameter Name	**IMPLEMENTATION_CONFIG_VARIANT**
Label	Config Variant
Multiplicity	1..1
Type	ENUMERATION
Default value	VariantPostBuild
Range	VariantPostBuild

3.1.1. CommonPublishedInformation

Parameters included	
Parameter name	**Multiplicity**

Parameters included	
ArMajorVersion	1..1
ArMinorVersion	1..1
ArPatchVersion	1..1
SwMajorVersion	1..1
SwMinorVersion	1..1
SwPatchVersion	1..1
ModuleId	1..1
VendorId	1..1
Release	1..1

Parameter Name	ArMajorVersion
Label	AUTOSAR Major Version
Description	Major version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	3
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ArMinorVersion
Label	AUTOSAR Minor Version
Description	Minor version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	0
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ArPatchVersion
Label	AUTOSAR Patch Version
Description	Patch level version number of AUTOSAR specification on which the appropriate implementation is based on.

Multiplicity	1..1
Type	INTEGER_LABEL
Default value	0
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

Parameter Name	**SwMajorVersion**
Label	Software Major Version
Description	Major version number of the vendor specific implementation of the module.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	1
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

Parameter Name	**SwMinorVersion**
Label	Software Minor Version
Description	Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	0
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

Parameter Name	**SwPatchVersion**
Label	Software Patch Version
Description	Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	9
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

Parameter Name	ModuleId
Label	Numeric Module ID
Description	Module ID of this module from Module List
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	90
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

Parameter Name	VendorId
Label	Vendor ID
Description	Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	1
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

Parameter Name	Release
Label	Release Information
Multiplicity	1..1
Type	STRING_LABEL
Default value	
Configuration class	**PublishedInformation:**
Origin	Elektrobit Automotive GmbH

3.1.2. EepGeneral

Parameters included	
Parameter name	**Multiplicity**
EepDevErrorDetect	1..1
EepDriverIndex	1..1

Parameters included	
EepUseInterrupts	1..1
EepVersionInfoApi	1..1
EepWriteCycleReduction	1..1
EepFilePath	0..1
EepFileName	0..1

Parameter Name	EepDevErrorDetect	
Description	Pre-processor switch to enable and disable development error detection.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EepDriverIndex	
Description	Index of the driver, used by EA.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=254 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EepUseInterrupts	
Description	Switches to activate or deactivate interrupt controlled job processing. Not supported by the module implementation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EepVersionInfoApi	
Description	Pre-processor switch to enable / disable the API to read out the modules version information.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EepWriteCycleReduction	
Description	Switches to activate or deactivate write cycle reduction (EEPROM value is read and compared before being overwritten). Not supported by the module implementation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EepFilePath	
Description	This parameter specifies the destination path where EEPROM file is located. If empty, the file will be created at the location of executable. Otherwise e.g.: C:/temp	
Multiplicity	0..1	
Type	STRING	
Default value		
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	EepFileName
Description	This parameter specifies the EEPROM file name. This is the binary file where the EEPROM address space is emulated.
Multiplicity	0..1
Type	STRING
Default value	Eep_memory.bin

Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

3.1.3. EepInitConfiguration

Containers included		
Container name	**Multiplicity**	**Description**
EepDemEventParameterRefs	0..1	Container for the references to DemEventParameter elements which shall be invoked using the API Dem_ReportErrorStatus API in case the corresponding error occurs. The EventId is taken from the referenced DemEventParameter's DemEventId value. The standardized errors are provided in the container and can be extended by vendor specific error references.
EepExternalDriver	0..1	This container is present for external EEPROM drivers only. Internal EEPROM drivers do not use the parameter listed in this container, hence its multiplicity is 0 for internal drivers.

Parameters included	
Parameter name	**Multiplicity**
EepBaseAddress	1..1
EepDefaultMode	1..1
EepFastReadBlockSize	1..1
EepFastWriteBlockSize	1..1
EepJobCallCycle	0..1
EepJobEndNotification	0..1
EepJobErrorNotification	0..1
EepNormalReadBlockSize	1..1
EepNormalWriteBlockSize	1..1
EepSize	1..1

Parameter Name	**EepBaseAddress**
Description	This parameter is the EEPROM device base address. Not supported by the module implementation (assumed to be always 0).
Multiplicity	1..1

Type	INTEGER
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepDefaultMode**
Description	This parameter is the default EEPROM device mode after initialization.
Multiplicity	1..1
Type	ENUMERATION
Default value	MEMIF_MODE_SLOW
Range	MEMIF_MODE_FAST MEMIF_MODE_SLOW
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepFastReadBlockSize**
Description	Number of bytes read within one job processing cycle in fast mode. If the hardware does not support burst mode this parameter shall be set to the same value as EepNormalReadBlockSize.
Multiplicity	1..1
Type	INTEGER
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepFastWriteBlockSize**
Description	Number of bytes written within one job processing cycle in fast mode. If the hardware does not support burst mode this parameter shall be set to the same value as EepNormalWriteBlockSize.
Multiplicity	1..1
Type	INTEGER
Range	<=4294967295

	>=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepJobCallCycle**	
Description	Call cycle time of the EEPROM driver's main function. Unit: [s]	
Multiplicity	0..1	
Type	FLOAT	
Default value	0.01	
Range	<=Infinity	
	>=0.0	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepJobEndNotification**	
Description	This parameter is a reference to a callback function for positive job result (see EEP045).	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepJobErrorNotification**	
Description	This parameter is a reference to a callback function for negative job result (see EEP046).	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepNormalReadBlockSize**
Description	Number of bytes read within one job processing cycle in normal mode.
Multiplicity	1..1

Type	INTEGER
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepNormalWriteBlockSize**
Description	Number of bytes written within one job processing cycle in normal mode.
Multiplicity	1..1
Type	INTEGER
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepSize**
Description	This parameter is the used size of EEPROM device in bytes.
Multiplicity	1..1
Type	INTEGER
Default value	1024
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

3.1.4. EepDemEventParameterRefs

Parameters included	
Parameter name	**Multiplicity**
EEP_E_COMPARE_FAILED	0..1
EEP_E_ERASE_FAILED	0..1
EEP_E_READ_FAILED	0..1

Parameters included	
EEP_E_WRITE_FAILED	0..1

Parameter Name	EEP_E_COMPARE_FAILED	
Description	Reference to the DemEventParameter which shall be issued when the error "EEPROM compare failed (HW)" has occurred.	
Multiplicity	0..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EEP_E_ERASE_FAILED	
Description	Reference to the DemEventParameter which shall be issued when the error "EEPROM erase failed (HW)" has occurred.	
Multiplicity	0..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EEP_E_READ_FAILED	
Description	Reference to the DemEventParameter which shall be issued when the error "EEPROM read failed (HW)" has occurred.	
Multiplicity	0..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EEP_E_WRITE_FAILED	
Description	Reference to the DemEventParameter which shall be issued when the error "EEPROM write failed (HW)" has occurred.	
Multiplicity	0..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

3.1.5. EepExternalDriver

Parameters included	
Parameter name	**Multiplicity**
EepSpiReference	1..n

Parameter Name	EepSpiReference	
Description	Reference to SPI sequence (required for external EEPROM drivers).	
Multiplicity	1..n	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

3.1.6. EepPublishedInformation

Parameters included	
Parameter name	**Multiplicity**
EepAllowedWriteCycles	1..1
EepEraseTime	1..1
EepEraseUnitSize	1..1
EepEraseValue	1..1
EepMinimumAddressType	1..1
EepMinimumLengthType	1..1
EepReadUnitSize	1..1
EepSpecifiedEraseCycles	1..1
EepTotalSize	1..1
EepWriteTime	1..1
EepWriteUnitSize	1..1

Parameter Name	EepAllowedWriteCycles
Description	Specified maximum number of write cycles under worst case conditions of specific EEPROM hardware (e.g. +90?C)
Multiplicity	1..1

Type	INTEGER_LABEL	
Default value	4294967295	
Range	<=4294967295 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepEraseTime**	
Description	Maximum time for erasing one EEPROM data unit.	
Multiplicity	1..1	
Type	FLOAT_LABEL	
Default value	1	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepEraseUnitSize**	
Description	Size of smallest erasable EEPROM data unit in bytes.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	1	
Range	<=4294967295 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepEraseValue**	
Description	Value of an erased EEPROM cell.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	255	
Range	<=255 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild

Origin	AUTOSAR_ECUC

Parameter Name	**EepMinimumAddressType**	
Description	Minimum expected size of Eep_AddressType.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	4294967295	
Range	<=4294967295 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepMinimumLengthType**	
Description	Minimum expected size of Eep_LengthType.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	4294967295	
Range	<=4294967295 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepReadUnitSize**	
Description	Size of smallest readable EEPROM data unit in bytes.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	1	
Range	<=4294967295 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EepSpecifiedEraseCycles**

Description	Number of erase cycles specified for the EEP device (usually given in the device data sheet).
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	4294967295
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepTotalSize**
Description	Total size of EEPROM in bytes.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	4294967295
Range	<=4294967295 >=0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepWriteTime**
Description	Maximum time for writing one EEPROM data unit.
Multiplicity	1..1
Type	FLOAT_LABEL
Default value	0
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EepWriteUnitSize**
Description	Size of smallest writeable EEPROM data unit in bytes.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	1

Range	<=4294967295 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

3.1.7. PublishedInformation

Parameters included	
Parameter name	**Multiplicity**
PbcfgMSupport	1..1

Parameter Name	**PbcfgMSupport**	
Label	PbcfgM support	
Description	Specifies whether or not the Eep can use the PbcfgM module for post-build support.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**PublishedInformation:**	
Origin	Elektrobit Automotive GmbH	

3.2. Application programming interface (API)

3.2.1. Macro constants

3.2.1.1. EEP_E_BUSY

Purpose	Development error code used if Eep_Read(), Eep_Write(), Eep_Compare(), Eep_Erase(), Eep_SetMode() is called and EEPROM module is busy.

Value	33U

3.2.1.2. EEP_E_PARAM_ADDRESS

Purpose	Development error code used if Eep_Read(), Eep_Write(), Eep_Compare(), Eep_Erase() is called and parameter EepromAddress is not in within the valid EEPROM address range.
Value	17U

3.2.1.3. EEP_E_PARAM_CONFIG

Purpose	Development error code used if Eep_Init() is called with null parameter pointer.
Value	16U

3.2.1.4. EEP_E_PARAM_DATA

Purpose	Development error code used if Eep_Read(), Eep_Write(), Eep_Compare(), Eep_Erase() is called and parameter DataBufferPtr is null.
Value	18U

3.2.1.5. EEP_E_PARAM_LENGTH

Purpose	Development error code used if Eep_Read(), Eep_Write(), Eep_Compare(), Eep_Erase() is called and parameter Length is not within the specified minimum and maximum values.
Value	19U

3.2.1.6. EEP_E_UNINIT

Purpose	Development error code used if Eep_Read(), Eep_Write(), Eep_Compare(), Eep_Erase(), Eep_SetMode() is called and EEPROM module is uninitialized.
Value	32U

3.2.1.7. EEP_INSTANCE_ID

Purpose	AUTOSAR module instance identification.
Value	0U

3.2.1.8. EEP_INTERNAL_API_ID

Purpose	Service id for internal functions.
Value	0xFFU

3.2.1.9. EEP_SID_CANCEL

Purpose	AUTOSAR API service ID.
Value	6U
Description	Definition of service ID for Eep_Cancel().

3.2.1.10. EEP_SID_COMPARE

Purpose	AUTOSAR API service ID.
Value	5U
Description	Definition of service ID for Eep_Compare().

3.2.1.11. EEP_SID_ERASE

Purpose	AUTOSAR API service ID.
Value	4U
Description	Definition of service ID for Eep_Erase().

3.2.1.12. EEP_SID_GET_JOB_RESULT

Purpose	AUTOSAR API service ID.

Value	8U
Description	Definition of service ID for Eep_GetJobResult().

3.2.1.13. EEP_SID_GET_STATUS

Purpose	AUTOSAR API service ID.
Value	7U
Description	Definition of service ID for Eep_GetStatus().

3.2.1.14. EEP_SID_GET_VERSION_INFO

Purpose	AUTOSAR API service ID.
Value	10U
Description	Definition of service ID for Eep_GetVersionInfo().

3.2.1.15. EEP_SID_INIT

Purpose	AUTOSAR API service ID.
Value	0U
Description	Definition of service ID for Eep_Init().

3.2.1.16. EEP_SID_MAIN_FUNCTION

Purpose	AUTOSAR API service ID.
Value	9U
Description	Definition of service ID for Eep_MainFunction().

3.2.1.17. EEP_SID_READ

Purpose	AUTOSAR API service ID.

Value	2U
Description	Definition of service ID for Eep_Read().

3.2.1.18. EEP_SID_SET_MODE

Purpose	AUTOSAR API service ID.
Value	1U
Description	Definition of service ID for Eep_SetMode().

3.2.1.19. EEP_SID_WRITE

Purpose	AUTOSAR API service ID.
Value	3U
Description	Definition of service ID for Eep_Write().

3.2.2. Functions

3.2.2.1. Eep_Cancel

Purpose	Cancel EEPROM Driver ongoing job synchronously.
Synopsis	`void Eep_Cancel ( void );`
Service ID	EEP_SID_CANCEL
Sync/Async	Synchronous
Reentrancy	Non-Reentrant
Description	This function shall abort a running job synchronously

3.2.2.2. Eep_Compare

Purpose	Initiate EEPROM Driver compare operation.

Synopsis	`Std_ReturnType` **`Eep_Compare`** `( Eep_AddressType EepromAddress , const uint8 * DataBufferPtr , Eep_LengthType Length );`	
Service ID	EEP_SID_COMPARE	
Sync/Async	Asynchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`EepromAddress`	Start job EEPROM address
	`DataBufferPtr`	User buffer data pointer
	`Length`	Job length in bytes
Return Value	Status of the job acceptance	
	`E_OK`	job accepted
	`E_NOT_OK`	job not accepted, failure
Description	This function shall copy given parameters, and initiate compare job	

3.2.2.3. Eep_Erase

Purpose	Initiate EEPROM Driver erase operation.	
Synopsis	`Std_ReturnType` **`Eep_Erase`** `( Eep_AddressType EepromAddress , Eep_LengthType Length );`	
Service ID	EEP_SID_ERASE	
Sync/Async	Asynchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`EepromAddress`	Start job EEPROM address
	`Length`	Job length in bytes
Return Value	Status of the job acceptance	
	`E_OK`	job accepted
	`E_NOT_OK`	job not accepted, failure
Description	This function shall copy given parameters, and initiate erase job	

3.2.2.4. Eep_GetJobResult

Purpose	Get result of last job of EEPROM Driver.
Synopsis	`MemIf_JobResultType` **`Eep_GetJobResult`** `( void );`

Service ID	EEP_SID_GET_JOB_RESULT	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Return Value	Job result	
	`MEMIF_JOB_OK`	The job has been finished successfully.
	`MEMIF_JOB_PENDING`	The job has not yet been finished.
	`MEMIF_JOB_FAILED`	The job has not been finished successfully.
	`MEMIF_BLOCK_INCONSISTENT`	Compared data areas are not equal.
	`MEMIF_JOB_CANCELED`	The job has been canceled.
Description	This function shall synchronously return result of the last job accepted by EEPROM Driver	

3.2.2.5. Eep_GetStatus

Purpose	Get status of EEPROM Driver.
Synopsis	`MemIf_StatusType` **`Eep_GetStatus`** `( void );`
Service ID	EEP_SID_GET_STATUS
Sync/Async	Synchronous
Reentrancy	Reentrant
Return Value	Module status
	`MEMIF_UNINIT` The module has not been initialized (yet).
	`MEMIF_BUSY` The module is currently busy.
	`MEMIF_IDLE` The module is currently idle.
Description	This function shall synchronously return status of EEPROM Driver

3.2.2.6. Eep_GetVersionInfo

Purpose	Get version information of the EEPROM Driver.
Synopsis	`void` **`Eep_GetVersionInfo`** `( Std_VersionInfoType *const VersionInfoPtr );`
Service ID	EEP_SID_GET_VERSION_INFO

Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (out)	`VersionInfoPtr`	Pointer to where to store the version information of this module.
Description	This service returns the version information of this module. The version information includes: ► Vendor Id ► Module Id ► Instance Id ► Vendor specific version numbers Precondition: Function is only available if parameter EepVersionInfoApi is set to true.	

3.2.2.7. Eep_Init

Purpose	Initialize the EEPROM Driver.	
Synopsis	`void` **`Eep_Init`** `( const Eep_ConfigType * ConfigPtr );`	
Service ID	EEP_SID_INIT	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`ConfigPtr`	Config Pointer to driver configuration.
Description	This function initializes the EEPROM Driver.	

3.2.2.8. Eep_MainFunction

Purpose	Service to perform the processing of the EEPROM jobs.
Synopsis	`void` **`Eep_MainFunction`** `( void );`
Service ID	EEP_SID_MAIN_FUNCTION
Sync/Async	Synchronous
Reentrancy	Non-Reentrant
Description	This function shall perform the processing of the EEPROM read, write, erase and compare jobs.

3.2.2.9. Eep_Read

Purpose	Initiate EEPROM Driver read operation.	
Synopsis	`Std_ReturnType` **`Eep_Read`** `( Eep_AddressType EepromAddress , uint8 * DataBufferPtr , Eep_LengthType Length );`	
Service ID	EEP_SID_READ	
Sync/Async	Asynchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`EepromAddress`	Start job EEPROM address
	`Length`	Job length in bytes
Parameters (out)	`DataBufferPtr`	User buffer data pointer
Return Value	Status of the job acceptance	
	`E_OK`	job accepted
	`E_NOT_OK`	job not accepted, failure
Description	This function shall copy given parameters, and initiate read job	

3.2.2.10. Eep_SetMode

Purpose	Set the EEPROM Driver operation mode.	
Synopsis	`void` **`Eep_SetMode`** `( MemIf_ModeType Mode );`	
Service ID	EEP_SID_SET_MODE	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`Mode`	Required module mode to be set
Description	This function sets the EEPROM Driver operation mode into MEMIF_MODE_SLOW or MEMIF_MODE_FAST.	

3.2.2.11. Eep_Write

Purpose	Initiate EEPROM Driver write operation.

Synopsis	`Std_ReturnType` **`Eep_Write`** `( Eep_AddressType EepromAddress , const uint8 * DataBufferPtr , Eep_LengthType Length );`	
Service ID	EEP_SID_WRITE	
Sync/Async	Asynchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`EepromAddress`	Start job EEPROM address
	`DataBufferPtr`	User buffer data pointer
	`Length`	Job length in bytes
Return Value	Status of the job acceptance	
	`E_OK`	job accepted
	`E_NOT_OK`	job not accepted, failure
Description	This function shall copy given parameters, and initiate write job	

4. Bibliography

Bibliography