Elektrobit

EB tresos® AutoCore Generic 8

Time Sync documentation

product release 8.8.4

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

1. Overview of EB tresos AutoCore Generic 8 Time Sync documentation

Welcome to the EB tresos AutoCore Generic 8 Time Sync (ACG8 Time Sync) product documentation.

This document provides:

- Chapter 2, “Supported Features“: list of services supported by the ACG8 Time Sync
- Chapter 3, “ACG8 Time Sync release notes“: release notes for the ACG8 Time Sync modules
- Chapter 4, “ACG8 Time Sync user guide“: containing background information and instructions
- Chapter 5, “ACG8 Time Sync module references“: information about configuration parameters and the application programming interface

2. Supported Features

Time synchronization supports the following main features according to the AUTOSAR and IEEE 802.1AS specification:

- Clock synchronization between master and slave
- Clock rate correction (needs support from the Ethernet driver)
- Propagation delay measurement
- Hardware timestamping (needs support from the Ethernet driver)
- Time synchronization gateway functionality
- Announcement of time synchronization priority

3. ACG8 Time Sync release notes

3.1. Overview

This chapter provides the ACG8 Time Sync product specific release notes. General release notes that are applicable to all products are provided in the EB tresos AutoCore Generic documentation. Refer to the general release notes in addition to the product release notes documented here.

3.2. Scope of the release

3.2.1. Configuration tool

Your release of EB tresos AutoCore is compatible with the release of the EB tresos Studio configuration tool:

- EB tresos Studio: 28.2.0 b211016-0103

3.2.2. AUTOSAR modules

The following table lists the AUTOSAR modules that are part of this ACG8 Time Sync release.

Module name	AUTOSAR version and revision	SWS version and revision	Module version	Supplier
CanTSyn	4.6.0 []	4.6.0 [0000]	3.0.0	Elektrobit Automotive GmbH
EthTSyn	4.6.0 []	4.6.0 [0000]	3.0.0	Elektrobit Automotive GmbH
FrTSyn	4.6.0 []	4.6.0 [0000]	3.0.0	Elektrobit Automotive GmbH
StbM	4.6.0 []	4.6.0 [0000]	4.0.0	Elektrobit Automotive GmbH

Table 3.1. Hardware-Independent Modules specified by the AUTOSAR standard

3.2.3. EB (Elektrobit) modules

The following table lists all modules which are part of this release but are not specified by the AUTOSAR standard. These modules include tooling developed by EB or they may hold files shared by all other modules.

Module name	Module version	Supplier
No EB modules available		

Table 3.2. Modules not specified by the AUTOSAR standard

3.2.4. MCAL modules and EB tresos AutoCore OS

For information about MCAL modules and OS, refer to the respective documentation, which is available as PDF at `$TRESOS_BASE/doc/3.0_EB_tresos_AutoCore_OS` and `$TRESOS_BASE/doc/5.0_MCAL_modules`[1]. It is also available in the online help in EB tresos Studio. Browse to the folders `EB tresos AutoCore OS` and `MCAL modules`.

3.3. Module release notes

3.3.1. CanTSyn module release notes

- AUTOSAR R4.6 Rev 0
- AUTOSAR SWS document version: 4.6.0
- Module version: 3.0.0.B466224
- Supplier: Elektrobit Automotive GmbH

3.3.1.1. Change log

This chapter lists the changes between different versions.

Module version 3.0.0
2021-10-08

[1] `$TRESOS_BASE` is the location at which you installed EB tresos Studio.

- Upgrade the CanTSyn from ASR 4.4.0 to AR 20-11.

Module version 2.0.10
2021-08-20

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

Module version 2.0.9
2021-07-28

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

Module version 2.0.8
2021-06-25

- Network segmentd Id for physical channel identification through time validation notifications.

Module version 2.0.7
2021-05-28

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

Module version 2.0.6
2021-03-05

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

Module version 2.0.5
2021-02-12

- ASCCANTSYN-225 Fixed known issue: Reception gets stuck if invalid FUP/OFNS message is received and FollowUpTimeout is disabled.

Module version 2.0.4
2021-01-22

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

Module version 2.0.3
2020-10-23

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

Module version 2.0.2
2020-08-28

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

Module version 2.0.1
2020-06-19

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

Module version 2.0.0
2020-05-22

- Update CanTSyn module to be compliant with the AUTOSAR 4.4 specification.

Module version 1.6.12
2020-04-24

- ASCCANTSYN-191 Fixed known issue: Mismatch of sequence counter value that appeared on bus and reported to StbM for Time Validation.
- ASCCANTSYN-193 Fixed known issue: StbM_CanTSyn.h header file is not included in CanTSyn module.

Module version 1.6.11
2020-03-25

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

Module version 1.6.10
2020-02-21

- Implement time validation.

Module version 1.6.9
2019-09-06

- ASCCANTSYN-178 Fixed known issue: Consecutive SYNC messages with the same SC are incorrectly accepted.

Module version 1.6.8
2019-08-09

- HandleIDs are not automatically calculated by HandleID Wizard.

Module version 1.6.7
2019-07-12

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

Module version 1.6.6
2019-06-14

- Enhance CanTSyn exclusive areas to support multicore environment

Module version 1.6.5
2019-04-18

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

Module version 1.6.4
2019-03-22

- ASCCANTSYN-171 Fixed known issue: CRC check on SYNC, FUP, OFS and OFNS is performed independently of CanTSynRxCrcValidated.

Module version 1.6.3
2019-02-15

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

Module version 1.6.2
2018-12-21

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

Module version 1.6.1
2018-10-26

- ASCCANTSYN-157 Fixed known issue: SC check is omitted after CanTSynGlobalTimeFollowUpTimeout time-out at reception.
- ASCCANTSYN-155 Fixed known issue: CanTSyn accepts messages with a stuck sequence counter.

Module version 1.6.0
2018-09-20

- Provide CAN FD support according to AUTOSAR 4.3.1.
- Enhance precision of Global Time.

Module version 1.5.2
2018-06-22

- ASCCANTSYN-129 Fixed known issue: The OVS field is incorrectly processed when a FUP frame is received.
- Ignore received SC when Time Base timeout detected in StbM.
- Enable transmission mode per Controller.

Module version 1.5.1
2018-05-25

- ASCCANTSYN-110 Fixed known issue: Debounce counter is not considered after FUP or OFNS message transmission.

Module version 1.5.0
2018-04-20

- ASCCANTSYN-104 Fixed known issue: Immediate synchronization request is ignored if cyclicMsgResumeCounter is running.

- Implement Global Time Measurement Support.
- Add support for uint32 PduLengthType.

Module version 1.4.5
2018-02-16

- ASCCANTSYN-120 Fixed known issue: Processing of received SYNC frames interprets timestamp according to the wrong byte order.
- ASCCANTSYN-95 Fixed known issue: Invalid configuration if CanTSynGlobalTimeDebounceTime is smaller than CanTSynMasterConfirmationTimeout.

Module version 1.4.4
2017-12-15

- Align const-ness in CanTSyn_RxIndication() Signature with CanIf.
- Disabling of periodic transmission by setting CanTSynGlobalTimeTxPeriod to 0.
- ASCCANTSYN-100 Fixed known issue: CanTSyn sends messages without having the GLOBAL_TIME_BASE bit set.
- ASCCANTSYN-86 Fixed known issue: Follow-up messages are dropped erroneously.
- Provide rate correction in addition to pure offset correction.

Module version 1.4.3
2017-10-20

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

Module version 1.4.2
2017-09-22

- Time domain specific data ID lists for CRC.

Module version 1.4.1
2017-08-25

- Remove unused structure elements.

Module version 1.4.0
2017-07-28

- Provide actual precision of individual time bases.

Module version 1.3.0
2017-06-02

- On-request transmission of TimeSync messages.

Module version 1.2.4
2017-03-31

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

Module version 1.2.3
2017-03-10

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

Module version 1.2.2
2017-03-03

- ASCCANTSYN-35 Fixed known issue: Removed consecutive zero-based CanTSynGlobalTimeDomainIds constrain.
- Align function declaration with function definition.

Module version 1.2.1
2017-02-03

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

Module version 1.2.0
2016-11-04

- Support of Time Slave functionality

Module version 1.1.0
2016-09-23

- Full support of time master functionality (normal and offset time bases, CRC, and user data)

Module version 1.0.0
2016-05-31

- Initial AUTOSAR 4.2 version

3.3.1.2. New features

- Update CanTSyn module to be compliant with the AUTOSAR 4.4.

3.3.1.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- Zero disables CanTSynGlobalTimeFollowUpTimeout

  Description:

  If CanTSynGlobalTimeFollowUpTimeout parameter is configured to zero, CanTSynGlobalTimeFollowUp-Timeout will not be taken into consideration while waiting for FUP or OFNS.

  After reception of a SYNC frame, if `CanTSynGlobalTimeFollowUpTimeout` is configured to 0, the awaited FUP frame shall be accepted whenever it comes.

  After reception of an OFS frame, if `CanTSynGlobalTimeFollowUpTimeout` is configured to 0, the awaited OFNS frame shall be accepted whenever it comes.
- Time Validation

  Description:

  CanTSyn supports Time Validation as per AUTOSAR R19-11.

3.3.1.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- The CanTSyn PDUs should be unique across all masters and slaves

Description:

The CanTSyn module doesn't support the PDUs across different masters and slaves to be not unique.

Requirements:

SWS_CanTSyn_00029, SWS_CanTSyn_00039

- The CanTSyn does not support Post-Build

  Description:

  The CanTSyn module doesn't support Post-Build loadable or Post-Build variant handling.

- Value zero disables CanTSynGlobalTimeFollowUpTimeout timeout

  Description:

  When CanTSyn parameter CanTSynGlobalTimeFollowUpTimeout is configured to zero, it will be considered disabled.

3.3.1.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- Confirmation Timeout of Master Time Domain

  Description:

  Up to four seconds supported.

  Rationale:

  This restriction allows to use a timeout smaller than four seconds because of the internal virtual local time calculation function which is limited to differences of four seconds.

- FollowUp Timeout and CanTSynCyclicMsgResumeTime

  Description:

  Up to 65535 supported.

  Rationale:

  CanTSyn does not use values greater than UINT32 for CanTSynGlobalTimeFollowUpTimeout and CanTSynCyclicMsgResumeTime.

- CanTSyn is not able to detect a wrap around

Description:

CanTSyn is not able to detect a wrapped around Sequence Counter, when a SYNC/OFS message is received.

Rationale:

When two consecutive SYNC/OFS messages with the same Sequence Counter are received, the second SYNC/OFS message will always be rejected, considering its SC to be a "stuck" one.

- CanTSyn is not able to multiplex OFFSET messages with SYNC messages (using the same PDU)

  Description:

  CanTSyn does not support multiplexing.

  Rationale:

  When 2 CanTSynGlobalTimeMasterPdu are referring the same EcuC Pdu, CanIf module will not be able to map the received Id with the right CanTSynGlobalTimeMasterConfirmationHandleId. Even if the same handle id would be used, CanTSyn will not differentiate between consecutive TxConfirmations

3.3.1.6. Open-source software

CanTSyn does not use open-source software.

3.3.2. EthTSyn module release notes

- AUTOSAR R4.6 Rev 0
- AUTOSAR SWS document version: 4.6.0
- Module version: 3.0.0.B466224
- Supplier: Elektrobit Automotive GmbH

3.3.2.1. Change log

This chapter lists the changes between different versions.

Module version 3.0.0
2021-10-08

- Upgrade the EthTSyn from ASR 4.4.0 to AR 20-11.
- ASCETHTSYN-813 Fixed known issue: DevAuth TLV out of bounds access in case of incomplete Authentification TLV.

Module version 2.2.7
2021-09-17

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

Module version 2.2.6
2021-08-20

- ASCETHTSYN-788 Fixed known issue: EthTSyn drops a valid sync frame if the length parameter is greater than 44 bytes.
- Support of EthTSyn_TxConfirmation() according to AUTOSAR R20-11.
- ASCETHTSYN-793 Fixed known issue: EthTSyn wrongly reports generation error for a valid configuration.

Module version 2.2.5
2021-07-28

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

Module version 2.2.4
2021-06-25

- ASCETHTSYN-768 Fixed known issue: Wrong character in the EthTSyn_defs.mak file.
- Implemented switch delay compensation for Time Aware Bridge with GTM not as Management CPU
- Adapt Time Validation as specified in EthTSyn SWS R20-11

Module version 2.2.3
2021-05-28

- ASCETHTSYN-761 Fixed known issue: EthTSyn is missing in the Create ECU Configuration wizard.
- ASCETHTSYN-759 Fixed known issue: EthTSyn does not detect parallel Masters when EthTSynMasterSlaveConflictDetection is used.

Module version 2.2.2
2021-04-09

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

Module version 2.2.1
2021-03-05

- ASCETHTSYN-741 Fixed known issue: Wrong PREFAILED status reported to Dem.

Module version 2.2.0
2021-02-12

- Relocated parameters as per AR-3116.

Module version 2.1.0
2021-01-22

- Implemented Post-Build Selectable support.

Module version 2.0.6
2020-12-18

- Implemented SequenceCounterJumpWidth functionality.

Module version 2.0.5
2020-10-23

- Implemented EthTSynPdelayRespAndRespFollowUpTimeout functionality.
- Implemented behavior: If MessageCompliance is set to TRUE only one timedomain can be configured.

Module version 2.0.4
2020-09-25

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

Module version 2.0.3
2020-08-28

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

Module version 2.0.2
2020-07-31

- ASCETHTSYN-671 Fixed known issue: EthTSyn does not compile when TimeValidationSupport is used on one slave configuration.

Module version 2.0.1
2020-06-19

- EthTSyn calculates propagation delay independent of execution order of the mainfunctions in a task.
- ASCETHTSYN-656 Fixed known issue: Incomplete reporting of EthTSynPdelayFailedReportError event.

Module version 2.0.0
2020-05-22

- Updated EthTSyn module to AUTOSAR 4.4
- ASCETHTSYN-627 Fixed known issue: Wrong timeout supervision on EthTSyn slave.
- Implement EthTSynGlobalTimeFollowUpTimeout.
- ASCETHTSYN-638 Fixed known issue: PDelay can be calculated with wrong time stamps
- ASCETHTSYN-651 Fixed known issue: Incorrect seconds value in case of nanoseconds field bigger than one second

Module version 1.4.15
2020-04-24

- ASCETHTSYN-603 Fixed known issue: StbM_EthTSyn.h header file is not included in EthTSyn module.
- ASCETHTSYN-613 Fixed known issue: Wrong calculation of responseOriginTimestamp.

Module version 1.4.14
2020-03-25

- ASCETHTSYN-587 Fixed known issue: Time domain IDs cannot be correctly configured

- ASCETHTSYN-588 Fixed known issue: EthTSyn does not validate the Time Domain ID when receiving a message

Module version 1.4.13
2020-03-11

- Implemented support of Authentication Challenge TLV for Device Authentication supplicant.

Module version 1.4.12
2020-02-21

- ASCETHTSYN-562 Fixed known issue: No AUTOSAR Sub-TLV in SyncFUp frame if only CRC_NOT_SUPORTED is used.
- Support for interfaces which allow time validation of synchronized time.
- Implemented EthTSynSendSyncFrameOnlyOnHostPort functionality.
- ASCETHTSYN-572 Fixed known issue: SYNC FUP messages are sent with wrong data if immediate transmission is used.
- Implemented the pdelay functionality for the Time Aware Bridge with GTM as Management CPU.

Module version 1.4.11
2020-01-24

- ASCETHTSYN-563 Fixed known issue: Field gmTimeBaseIndicator is initialized with wrong values.

Module version 1.4.10
2019-11-08

- ASCETHTSYN-549 Fixed known issue: Wrong value used for Pdelay calculation.

Module version 1.4.9
2019-10-11

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

Module version 1.4.8
2019-09-06

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

Module version 1.4.7
2019-08-09

- Simple Bridge in relation with EthSwt.

Module version 1.4.6
2019-06-14

- Enhance EthTSyn exclusive areas to support multicore environment.

Module version 1.4.5
2019-04-18

- Introduce support for Offset Timebases.

Module version 1.4.4
2019-03-22

- ASCETHTSYN-459 Fixed known issue: Fields gmTimeBaseIndicator, lastGmPhaseChange and scaledLastGmFreqChange are initialized with wrong values.
- Internal module improvement. This module version update does not affect module functionality.

Module version 1.4.3
2019-02-15

- ASCETHTSYN-455 Fixed known issue: EthIf_GetEgressTimeStamp() called with wrong BufIdx after a pdelay_req frame was sent.
- ASCETHTSYN-458 Fixed known issue: EthTSyn_SwitchPorts variable is wrongly mapped in CONST memory area.

Module version 1.4.2
2019-01-25

- Introduce support for Diagnostic event reporting for Ethernet TimeSync.
- Implemented EthTSynGlobalTimePdelayRespEnable functionality.

Module version 1.4.1
2018-10-26

- ASCETHTSYN-407 Fixed known issue: Pdelay response time stamp is not retrieved in context of EthTSyn_TxConfirmation().
- ASCETHTSYN-412 Fixed known issue: Variable is not defined in an appropriate MemMap section.
- ASCETHTSYN-418 Fixed known issue: EthTSyn rejects Pdelay response and Pdelay response follow-up frames.

Module version 1.4.0
2018-09-20

- ASCETHTSYN-391 Fixed known issue: EthTSyn corrects the time in the Eth driver before StbM can check for time leap.
- ASCETHTSYN-386 Fixed known issue: EthTSyn master: SYNC periodicity is wrong.
- Implemented enhancement for precision of global time.

Module version 1.3.2
2018-06-22

- Added support for debounce transmissions.
- Updated signature of EthTSyn_SetTransmissionMode() to AR 4.2.2.

Module version 1.3.1
2018-05-18

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

Module version 1.3.0
2018-04-20

- ASCETHTSYN-339 Fixed known issue: organizationId incorrectly set in AUTOSAR TLV header
- ASCETHTSYN-334 Fixed known issue: EthTSyn performs a write access on NULL_PTR if EthSwt_-ProvideTxBuffer() returns an NULL_PTR for parameter BufPtr.
- ASCETHTSYN-388 Fixed known issue: organizationId incorrectly set in IEEE TLV header.
- ASCETHTSYN-353 Fixed known issue: The Time Gateway synchronization status is incorrectly used in Status Sub-TLV.

- ASCETHTSYN-358 Fixed known issue: EthTSyn generates incorrect values for switch host port

Module version 1.2.1
2018-02-16

- ASCETHTSYN-285 Fixed known issue: Length value of FUP frame is not correctly set for certain Sub-TLVs
- ASCETHTSYN-273 Fixed known issue: Only Modulo 16 part of Sequence Counter shall be used for Data ID selection
- Internal module improvement. This module version update does not affect module functionality.

Module version 1.2.0
2017-12-15

- Added switch delay compensation for acting as global time master.
- Updated API calls of StbM_GetCurrentTimeRaw() and StbM_GetCurrentTimeDiff() according to AUTOSAR RfC 77248.
- ASCETHTSYN-280 Fixed known issue: A compiler error occurs if the CRC support is disabled for Tx PDUs and enabled for Rx PDUs.
- Added on-request transmission of TimeSync messages.

Module version 1.1.7
2017-11-17

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

Module version 1.1.6
2017-10-24

- Updated design

Module version 1.1.5
2017-09-22

- ASCETHTSYN-228 Fixed known issue: If the SyncToGateway bit is not set in the status Sub-TLV, the received frame shall be wrongfully discarded.
- ASCETHTSYN-223 Fixed known issue: DET errors reported incorrectly during EthTSyn_TrcvLinkStateChg() and EthTSyn_TxConfirmation().

- Updated to MISRA 2012

Module version 1.1.4
2017-08-25

- Added UserData Sub-TLV support

Module version 1.1.3
2017-07-28

- Updated receiving unexpected SubTLVs according to RfC 77619

Module version 1.1.2
2017-06-30

- ASCETHTSYN-206 Fixed known issue: Valid received sync follow-up frame gets discarded due to incorrect check.

Module version 1.1.1
2017-04-25

- ASCETHTSYN-197 Fixed known issue: Sync follow up frame TLVs get corrupted
- ASCETHTSYN-198 Fixed known issue: If EthTSynMessageCompliance is set to false, received sync follow-up frames are discarded
- ASCETHTSYN-176 Fixed known issue: EthTSyn compilation reports an error if no slave is configured.

Module version 1.1.0
2017-03-31

- Added configurable link delay handling
- Added CRC calculation for TimeSync message

Module version 1.0.7
2017-03-03

- Updated the configuration schema to AUTOSAR 4.3.0
- ASCETHTSYN-154 Fixed known issue: Temporary synchronization inaccuracy of 1 second

Module version 1.0.6
2017-02-03

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

Module version 1.0.5
2017-01-05

- Added support of full range correction field handling
- Added VLAN support

Module version 1.0.4
2016-12-03

- Updated `EthTSyn_Init()` to also allow a call with `NULL_PTR`

Module version 1.0.3
2016-11-04

- Added usage of `EthIf_UpdatePhysAddrFilter()`
- Added support for `Eth_BufIdxType`

Module version 1.0.2
2015-11-06

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

Module version 1.0.1
2015-06-19

- Added DBC/LDF/Fibex importer support
- Added task auto assign of EthTSyn_MainFunction() for RTE

Module version 1.0.0
2015-02-20

- Initial release for AUTOSAR 4.2 rev. 1

3.3.2.2. New features

- Implemented switch delay compensation for Time Aware Bridge with GTM not as Management CPU.
- Time measurement with Switches (Time Aware Bridges) are supported for the Time Validation use case.

3.3.2.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- Periodic sending of announce frames

  Description:

  EthTSyn masters can send announce frames periodically. This can be configured through The EB configuration parameter `EthTSynGeneral/EthTSynAnnounceFrameSupport`.

Value	Description
TRUE	An EthTSyn master sends announce frames in the same interval as sync frames.
FALSE	No announce frames are sent.

- Filtering of measured peer delay values

  Description:

  It is possible to filter measured peer delay values. The filter reduces a negative impact of incorrect peer delay measurements on the synchronization. The filtering is controlled through the EB configuration parameter `EthTSynGlobalTimeDomain/EthTSynGlobalTimeSlave/EthTSynPdelayFilter`. Setting the configuration parameter to zero disables the filtering.

- Adding and removing MAC address

  Description:

  `EthTSyn` uses `EthIf_UpdatePhysAddrFilter()` in the context of `EthTSyn_TrcvLinkStateChg()` to add and remove the MAC address 01-80-C2-00-00-0E to/from the `Eth` filter.

- Correction field range

  Description:

  `EthTSyn` supports the full ns range of the correction field from a sync follow up frame. It allows time aware bridges to use the correction field as an offset from the origin time stamp in case there is a master breakdown.

- Switch delay compensation for global time master of a switch

Description:

EthTSyn supports switch delay compensation for global time master of a switch. EthTSyn masters with configured switch port in EthTSynSwitchManagementEthSwitchPortRef will act as a global master for the referred port, providing the same origin time stamp on each port(referred from the same global time domain masters) for every sync frame cycle. EthTSynPortConfig contains configuration parameters that should not port depended, but should be the same for all masters in the respective EthTSynGlobalTimeDomain(e.g. EthTSynGlobalTimeEthIfRef, EthTSynFramePrio). Therefore the configuration values used will be of the first master configured in the time domain. This EB feature requires EB specific EthSwt and EthIf modules. It is also necessary that EthTSynSwitchManagementEthSwitchPortHostRef contains a reference to the switch port with port role set to host. For more details: AUTOSAR_RfC_79958.

- Use ns fractional part to store SwitchIdx and PortIdx

  Description:

  If the configuration parameter EthTSynSwtPortIdxInCorrField is set to true, SwitchIdx and PortIdx shall be stored in the unused 2 bytes of fractional ns part of the correction field.

- EthTSyn calculates propagation delay independent of execution order of the mainfunctions in a task.
- Time Validation

  Description:

  EthTSyn supports Time Validation as per AUTOSAR R19-11.

- EthTSynPortNumber

  Description:

  Config parameter that is enabled if EthTSynSendSyncFrameOnlyOnHostPort is FALSE and Time Aware Bridge with GTM as Management CPU is configured or Time Aware Bridge without GTM as Management CPU is configured and acts like a boundary clock. EthTSynPortNumber is unique over all configured EthTSynTimeDomain and is added in the transmitted PTPHeader of transmitted frames(used for frame validation on receiver side).

- Time Validation on Time Aware Bridges

  Description:

  In case of a Time Aware Bridges, if time validation is enabled for the configured ports, EthTSyn will pass to StbM the same SourcePortId for all of them.

- EthSwt provides EngressTimeStamp and IngressTimeStamp

  Description:

If switch is used and time validation is enabled for the configured ports, the EgressTimeStamp and IngressTimeStamp will be retrieved from EthSwt and provided to StbM.

3.3.2.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- Pdelay is not calculated on a Time Master.

  Description:

  The propagation delay is calculated only on the Time Slave.

  Rationale:

  Global Time Synchronization in AUTOSAR does yet not define dynamic reconfiguration or backup strategies that will reassign the role as Time Master, therefore propagation delay measurements make currently no sense for a Time Master

  Requirements:

  PRS_TS_00164, PRS_TS_00149, PRS_TS_00141, SWS_EthTSyn_00201, SWS_EthTSyn_00014, PRS_TS_00050

- Pdelay is calculated on each Time Slave.

  Description:

  Feature not supported.

  Rationale:

  The measurement of the propagation delay only on that Time Domain with the lowest Time Domain ID was introduced in AUTOSAR PRS Release 1.5.1. and is not yet supported.

  Requirements:

  PRS_TS_0049

- `Eth_BufIdxType` is used instead of uint8.

  Description:

  `EthTSyn` uses `Eth_BufIdxType` instead of `uint8` for its APIs `EthTSyn_TxConfirmation()` and the API calls `EthIf_EnableEgressTimeStamp()`, `EthIf_GetEgressTimeStamp()` and `EthIf_GetIngressTimeStamp()`.

  Rationale:

With AUTOSAR 4.2.2 `Eth_BufIdxType` was introduced throughout the whole stack as new type for BufIdx.

Requirements:

SWS_EthTSyn_00042

- `Runtime Error Detection` not supported.

Description:

The errors shall be reported via Det_ReportError().

Requirements:

SWS_EthTSyn_00145, SWS_EthTSyn_00146, SWS_EthTSyn_00144

- The position of each Sub-TLV is not arbitrary.

Description:

The Sub-TLVs will be added in the order that is defined in AUTOSAR SWS.

Requirements:

PRS_TS_00072

- ResidenceTime is not supported.

Description:

rx_residence_time and tx_residence_time are not supported.

Requirements:

PRS_TS_00168, PRS_TS_00169, PRS_TS_00170, PRS_TS_00171, PRS_TS_00166, PRS_TS_00167, ECUC_EthTSyn_00061_Conf, ECUC_EthTSyn_00060_Conf

- ETHTSYN_E_CTRL_IDX not reported for EthTSyn_TxConfirmation() and EthTSyn_TrcvLinkStateChg().

Description:

If EthTSyn_TxConfirmation() or EthTSyn_TrcvLinkStateChg() are called with invalid CtrlIdx, no error shall be reported to det.

Rationale:

See info: ASCETHTSYN-223.

Requirements:

SWS_EthTSyn_00174, SWS_EthTSyn_00175

- TMAC

  Description:

  These configuration parameters are not used by EthTSyn: EthTSynGlobalTimeSecureTmacLength, EthTSynTxTmacCalculated and EthTSynRxTmacValidated.

  Rationale:

  These configuration parameters, within ASR4.4.0, are marked as "draft" and no functionality is provided for them

  Requirements:

  ECUC_EthTSyn_00077_Conf, ECUC_EthTSyn_00079_Conf, ECUC_EthTSyn_00080_Conf

- Not clarified requirements

  Description:

  These configuration parameters are not used by EthTSyn: EthTSynGlobalTimeSecureTmacLength, EthTSynTxTmacCalculated, EthTSynRxTmacValidated and EthTSynGlobalTimeMinMsgGap.

  Rationale:

  These configuration parameters, within ASR4.4.0, are marked as "draft" and no functionality is provided for them

  Requirements:

  ECUC_EthTSyn_00077_Conf, ECUC_EthTSyn_00079_Conf, ECUC_EthTSyn_00080_Conf, ECUC_-EthTSyn_00078_Conf

- T0 used as preciseOriginTimestamp.

  Description:

  Using T0 as preciseOriginTimestamp if EthTSynHardwareTimestampSupport is not supported.

  Requirements:

  SWS_EthTSyn_00189

- EthTSynGlobalTimeDebounceTime shall be the same for all switch ports.

  Description:

EthTSynGlobalTimeDebounceTime was moved from EthTSynPortConfig to EthTSynGlobalTimeDomain therefore all ports of a time domain shall use the same configured EthTSynGlobalTimeDebounceTime.

Rationale:

See info: AR-3116.

Requirements:

SWS_EthTSyn_00187

- No SequenceCounter check after a synchronization timeout.

  Description:

  Timeout is wrongfully used in this requirement. StbM_GetTimeBaseStatus() informs EthTSyn if a sync timeout occured or not.

  Requirements:

  PRS_TS_00199_SkipCheckSCounterAfterTimeout

- No EthTSynSwitchMgmtRxMessageBufferCount for EthTSyn used in a bridge configuration.

  Description:

  EthTSynSwitchMgmtRxMessageBufferCount not supported.

  Requirements:

  ECUC_EthTSyn_00059_Conf

- Pdelay value initiated at EthTSyn_Init().

  Description:

  When EthTSyn_Init() is called, the Pdelay shall be initiated with the static configured value EthTSynGlobalTimePropagationDelay.

  Requirements:

  SWS_EthTSyn_00010

3.3.2.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- The correctionField calculation is not supported

Description:

`EthTSyn` does not perform calculation of `correctionField`. The function `StbM_GetCurrentTimeRaw()` is not used for `correctionField` calculation.

Rationale:

The `correctionField` calculation is only necessary for network time aware bridges.

- Peer delay calculation is not supported for switches

Description:

Peer delay calculation is not supported for switches. `EthTSyn` does not compensate the switch delay. Therefore, static Peer delay shall be used.

- The config values of the first master entry of each time domain are used.

Description:

Multiple masters configured on the same Time Domain are used to get more references to switch ports. For remaining configuration parameters (such as `EthTSynGlobalTimeTxPeriod`), the MCG shall only use the values taken from the first entry of master port config list and ignore the other entries, on the same Time Domain.

- The supplicant must be an EthTSyn slave.

Description:

`DevAuth` module is supporting only the supplicant side.

- Corrections Std 802.1AS/Cor 1-2013 and IEEE Std 802.1AS-2011/Cor 2-2015 are not supported.

Description:

`EthTSyn` module is supporting the IEEE Standard 802.1AS - 30 of March 2011 version as specified by the Autosar SWS chapter 3.2([13]).

- The use case “Time Aware Bridge with GTM not as Management CPU” is not fully implemented, the synchronization of the Bridge/Slave is missing.

Description:

The Bridge/Slave is receiving the Sync/Fup pairs and this is forwarded to the master ports. The Fup frame contains an updated correctionField for each master switch port.

- Diagnostic event reporting for Ethernet TimeSync is not supported for switches (Bridge/Master or Bridge/Slave).

Description:

`EthTSyn` does not perform diagnostic event reporting if more than 12 Sync-/Follow_Up message in a row are dropped (missing) (Ausfall der Uhrensynchronisation). on a Bridge/Slave. `EthTSyn` does not perform diagnostic event reporting if more than 6 responses of a Pdelay_Req to a Pdelay_Resp-/Pdelay_Resp_Follow_Up are missing (Ausfall der Uhrensynchronisation) on a Bridge/Master or Bridge/Slave. `EthTSyn` does not perform diagnostic event reporting if Sync messages are received on an Ethernet switch port configured as master (Ungültige Anzahl Uhrensynchronisationsmaster) on a Bridge/Master..

- The timeout supervision of the Follow_Up frame (of the subsequent Sync message) is not supported for the use case “Time Aware Bridge with GTM not as Management CPU”.

  Description:

  Because the synchronization of the Bridge/Slave is not implemented, is not relevant to supervise the missing Follow_Up frame. This timeout can be configured on the time slave where the bridge forwards the messages.

- The EthTSynPdelayLatencyThreshold shall be limited to a range of 0..1 000 000 000 nanoseconds(1 second).

  Description:

  A calculated Pdelay over 1 second is suspicious and there are checks in implementation that prevent that. Therefore a threshold over 1 second would be obsolete.

- PostBuild selectable is not used on Time Aware Bridge with GTM not as Management CPU.

  Description:

  If Time Aware Bridge with GTM not as Management CPU is configured, `POSTBUILDVARIANTSUPPORT` can not be used.

- Two EthTSynTimeDomains can not be configured as follows: one as Time Aware Bridge with GTM not as Management CPU and one as Time Aware Bridge with GTM as Management CPU.

  Rationale:

  Currently out of scope.

- If simple bridge is used and `EthTSynSimpleBridgeTSynSendSync` is disabled, no validation shall be done on the sync frames, therefore EthTSyn can not drop invalid frames.

  Rationale:

  EthSwt performs the Sync transmission.

- No DEM reporting in case a master port receives a sync frame on a Time Aware Bridge with GTM not as Management CPU.

  Rationale:

DEM configuration is done on the `EthTSynTimeDomain`, not on each port.

- DevAuth is not implemented for switch config.
- Multiple `EthTSynTimeDomain` can not reference the same EthIf controller.
- Debouncing not used for `EthTSynTimeDomain` that have `EthTSynSimpleBridgeTSynSendSync` disabled.

  Rationale:

  Since EthSwt handles the sync frame transmission, no debounce can be added between Sync/SyncFUp frame.
- No dynamic CRC on a Time Aware Bridge with GTM not as Management CPU.

  Rationale:

  SyncFUp frame validation shall only be done based on the received TimeSecured Sub-TLV. Other Sub-TLV types can only be received, but not validated.
- EthTSyn_GetProtocolParam() works only for EthTSynTimeDomain that has EthTSynPortRole configured as slave.

  Rationale:

  Since EthTSyn_GetProtocolParam() returns the ProtocolParam received in a SyncFUp frame and only the slave TimeDomain receives that type of frames, using a master TimeBase shall be handled as using a not configured TimeBaseId.
- EthTSyn_SetProtocolParam() works only for EthTSynTimeDomain that has EthTSynPortRole configured as master.

  Rationale:

  EthTSyn_SetProtocolParam() is used to set the information TLV parameters of a SyncFup message prior transmission. Since only the master TimeDomain is responsible for transmission of a SyncFup frame, using a slave TimeBase shall be handled as using a not configured TimeBaseId.

3.3.2.6. Open-source software

EthTSyn does not use open-source software.

3.3.3. FrTSyn module release notes

- AUTOSAR R4.6 Rev 0

- AUTOSAR SWS document version: 4.6.0
- Module version: 3.0.0.B466224
- Supplier: Elektrobit Automotive GmbH

3.3.3.1. Change log

This chapter lists the changes between different versions.

Module version 3.0.0
2021-10-08

- Upgrade the FrTSyn from ASR 4.4.0 to AR 20-11.

Module version 2.0.11
2021-09-17

- Disable the retry transmit in case FrIf_Transmit() returns E_NOT_OK

Module version 2.0.10
2021-08-20

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

Module version 2.0.9
2021-06-25

- Identify the Physical Channel on a system scope through time validation notifications from FrTSyn.

Module version 2.0.8
2021-05-28

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

Module version 2.0.7
2021-03-05

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

Module version 2.0.6
2021-01-22

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

Module version 2.0.5
2020-12-18

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

Module version 2.0.4
2020-10-23

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

Module version 2.0.3
2020-09-25

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

Module version 2.0.2
2020-08-28

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

Module version 2.0.1
2020-07-31

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

Module version 2.0.0
2020-06-19

- Upgrade FrTSyn module to AUTOSAR 4.4.0.

Module version 1.4.11
2020-04-24

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

Module version 1.4.10
2020-02-21

- ASCFRTSYN-193 Fixed known issue: TxPeriod not maintained in case of transmission failure.
- Fixed known issue: ASCFRTSYN-184 Race condition during transmission can lead to data loss
- Added support for Time Validation.

Module version 1.4.9
2019-12-06

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

Module version 1.4.8
2019-11-08

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

Module version 1.4.7
2019-10-11

- ASCFRTSYN-142 Fixed known issue: FrTSyn accepts messages with stuck sequence counter.

Module version 1.4.6
2019-06-14

- Enhance FrTSyn exlusive areas to support multicore environment.

Module version 1.4.5
2019-04-18

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

Module version 1.4.4
2019-03-22

- ASCFRTSYN-155 Fixed known issue: Synchronization message delayed on start-up

Module version 1.4.3
2019-02-15

- ASCFRTSYN-153 Fixed known issue: FrTSyn fails if FrTSynRxCrcValidated equals CRC_IGNORED and SYNC message type equals 0x20
- ASCFRTSYN-154 Fixed known issue: StbM_GetCurrentVirtualLocalTime() is not called within an exclusive area for Offset Time bases

Module version 1.4.2
2019-01-25

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

Module version 1.4.1
2018-10-26

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

Module version 1.4.0
2018-09-20

- ASCFRTSYN-81 Fixed known issue: User data byte handling fails for FRTSYN_CRC_IGNORED
- ASCFRTSYN-134 Fixed known issue: FrTSyn does not send a SYNC or OFS message immediately after cyclicMsgResumeCounter expires
- Added support for enhanced precision (AUTOSAR SWS 4.4.0)

Module version 1.3.3
2018-08-24

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

Module version 1.3.2
2018-06-22

- ASCFRTSYN-118 Fixed known issue: StbM_BusSetGlobalTime() should not be called if the call to FrIf_-GetGlobalTime() fails

Module version 1.3.1
2018-05-25

- Moved FrTSynGlobalTimeSequenceCounterJumpWidth under FrTSynGlobalTimeSlave.

Module version 1.3.0
2018-04-20

- Implement Global Time Measurement Support.

Module version 1.2.0
2018-02-16

- ASCFRTSYN-102 Fixed known issue: Incorrect calculation of global time in FrTSyn time slaves.
- ASCFRTSYN-43 Added support for Offset Time Bases.

Module version 1.1.4
2018-01-19

- ASCFRTSYN-77 Fixed known issue: FrTSyn accepts messages with wrong message type.
- ASCFRTSYN-81 Fixed known issue: User data byte handling fails for FRTSYN_CRC_IGNORED.

Module version 1.1.3
2017-12-15

- Added support for disabled periodic transmission.
- Added support for Time domain specific data ID lists for CRC.
- ASCFRTSYN-87 Fixed known issue: FrTSyn transmits SYNC messages without global time base being initially set.
- Changed signature of FrTSyn_RxIndication(), aligned with FrIf.

Module version 1.1.2
2017-10-20

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

Module version 1.1.1
2017-09-22

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

Module version 1.1.0
2017-07-28

- ASCFRTSYN-55 Fixed known issue: FrTSyn_TriggerTransmit function return type is wrong.
- Support for Immediate Time Synchronization

Module version 1.0.6
2017-03-31

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

Module version 1.0.5
2017-03-10

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

Module version 1.0.4
2017-03-03

- Move integration requirements to separate reqm file.

Module version 1.0.3
2017-01-05

- Documented MISRA violation.

Module version 1.0.2
2016-12-02

- Create implementation for FrTSyn ASR 4.2.2 for the Master Domain Feature including CRC support and User Data support, without Offset Time Bases support.

Module version 1.0.1
2016-11-04

- Design updates.

Module version 1.0.0
2016-09-23

- Initial AUTOSAR 4.2 version
- Create implementation for FrTSyn ASR 4.2.2 for the Slave Domain Feature including CRC support and User Data support, without Offset Time Bases support.

3.3.3.2. New features

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

3.3.3.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- Time Validation

  Description:

  FrTSyn supports Time Validation as per AUTOSAR R19-11 and identifying the Physical Channel on a system scope through time validation notifications from FrTSyn as per AUTOSAR R20-11.

3.3.3.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- Only pre-compile configuration is supported

  Description:

The FrTSyn module only supports configuration variant VARIANT-PRE-COMPILE. VARIANT-POST-BUILD is not supported.

Variant handling is not supported.

Requirements:

SWS_FrTSyn_00077

- Not clarified requirements

Description:

These configuration parameters are not used by FrTSyn: ECUC_FrTSyn_00034 ECUC_FrTSyn_00035 ECUC_FrTSyn_00036 ECUC_FrTSyn_00037 ECUC_FrTSyn_00038 ECUC_FrTSyn_00039

Rationale:

These configration parameters, within ASR20-11, are marked as "draft" and no functionality is provided for them

Requirements:

ECUC_FrTSyn_00034 ECUC_FrTSyn_00035 ECUC_FrTSyn_00036 ECUC_FrTSyn_00037 ECUC_FrTSyn_00038 ECUC_FrTSyn_00039

3.3.3.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- Number of supported Time Domains

Description:

Up to 254 Time Domains are supported.

Rationale:

The number of supported Time Domains is reduced from Infinity to a size which fits into an uint8. Additionally one slot is reserved for the handling of multiplexed PDUs.

- Confirmation Timeout of Master Time Domain

Description:

Up to one second supported.

Rationale:

Only a timeout smaller than one second can be used because the RawTime API StbM_GetCurrentTimeDiff() from StbM is limited to intervals of one second.

- Values of configuration parameter FrTSynRxCrcValidated

  Description:

  The value of CRC_OPTIONAL for the configuration parameter FrTSynRxCrcValidated is not supported.

  Rationale:

  The implementation of FrTSyn is according to SWS version 4.2.2 with certain features taken over from 4.3.1. At this moment this hasn't been taken over.

- FrTSyn is not able to detect a wrap around

  Description:

  FrTSyn is not able to detect a wrapped around Sequence Counter, when a SYNC/OFS message is received.

  Rationale:

  When two consecutive SYNC/OFS messages with the same Sequence Counter are received, the second SYNC/OFS message will always be rejected, considering its SC to be a "stuck" one.

3.3.3.6. Open-source software

FrTSyn does not use open-source software.

3.3.4. StbM module release notes

- AUTOSAR R4.6 Rev 0
- AUTOSAR SWS document version: 4.6.0
- Module version: 4.0.0.B466224
- Supplier: Elektrobit Automotive GmbH

3.3.4.1. Change log

This chapter lists the changes between different versions.

Module version 4.0.0
2021-10-08

- Upgrade StbM module from ASR 4.4.0 to AR 20-11.

Module version 3.1.10
2021-09-17

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

Module version 3.1.9
2021-08-20

- ASCSTBM-663 Fixed known issue: StbM uses wrong name pattern for Rte interfaces.

Module version 3.1.8
2021-07-28

- ASCSTBM-653 Fixed known issue: StbM does not compile if Time Recording is used only on Offset Time Bases.

Module version 3.1.7
2021-06-25

- Make Time Recording independent from Time Validation.
- Implement Multicore Distribution.

Module version 3.1.6
2021-05-28

- ASCSTBM-643 Fixed known issue: Offset measurement data is not detected as available in StbM.

Module version 3.1.5
2021-04-30

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

Module version 3.1.4
2021-03-05

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

Module version 3.1.3
2021-02-12

- ASCSTBM-608 Fixed known issue: StbM provides the wrong time if the OsCounter is used as time source.

Module version 3.1.2
2021-01-22

- Internal module improvement. This module version update does not affect module functionality.
- ASCSTBM-607 Fixed known issue: StbM might get/set wrong time when using Os Time Stamp as time source

Module version 3.1.1
2020-12-18

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

Module version 3.1.0
2020-10-23

- Implement support for StbM_GetTimeLeap API.
- Enhanced validation of Time Stamp and User Data parameters across functions.
- ASCSTBM-584 Fixed known issue: StbM causes a reset/error hook
- Implement support for StbM_GetMasterConfig API.
- ASCSTBM-587 Fixed known issue: StbM does not generate when Notification Customers are used together with StbMUseOSGetTimeStamp

Module version 3.0.3
2020-09-25

- ASCSTBM-573 Fixed known issue: StbM does not generate if different masks are used for CompuScales with the same ShortLabel.
- ASCSTBM-565 Fixed known issue: StbM does not compile if Offset Correction is used and the time base is driven by a source other than the Eth Driver
- ASCSTBM-553 Fixed known issue: StbM provides the wrong time if Offset Correction is used

Module version 3.0.2
2020-08-28

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

Module version 3.0.1
2020-07-31

- Support for StbM_GetCurrentTimeExtended API.

Module version 3.0.0
2020-06-19

- Upgraded StbM module to Autosar 4.4 SWS.

Module version 2.6.19
2020-05-22

- ASCSTBM-526 Fixed known issue: StbM does not compile when Trigger Customers are used with an OS other than EB's

Module version 2.6.18
2020-04-24

- Support Autosar 4.2/4.3/4.4 Gpt MCALs with StbM.

- ASCSTBM-520 Fixed known issue: StbM fails to load the Global Time from NvM at initialization.

Module version 2.6.17
2020-03-25

- ASCSTBM-494 Fixed known issue: Rte wrongly generates Measurement Notification R-PORTs for MASTER StbM time bases.

Module version 2.6.16
2020-02-21

- Implement Time Validation support in StbM.
- Implement Time Notification support in StbM.

Module version 2.6.15
2019-11-08

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

Module version 2.6.14
2019-10-11

- ASCSTBM-445 Fixed known issue: Time Recording data is wrong at initialization.
- ASCSTBM-447 Fixed known issue: TimeRecording records a wrong value for the rate deviation.
- ASCSTBM-448 Fixed known issue: StbM time base data written in the shared memory location can be wrong.

Module version 2.6.13
2019-09-06

- ASCSTBM-441 Fixed known issue: Inconsistent Virtual Local Time between EthTSyn and StbM.

Module version 2.6.12
2019-08-09

- ASCSTBM-437 Fixed known issue: StbM provides wrong time if interrupted by a Time Setter function.

Module version 2.6.11
2019-07-12

- Implement VSwt support in StbM.

Module version 2.6.10
2019-06-14

- Enhance StbM exclusive areas to support multicore environment
- Implement the use of OS_GetTimeStamp()/Mk_ReadTime as EB specific extension.

Module version 2.6.9
2019-05-17

- ASCSTBM-413 Fixed known issue: Rate Correction Measurement is not initialized when only one measurement is configured for the first Time Base.
- ASCSTBM-405 Fixed known issue: StbM does not use StbMClockPrescaler/StbMClockFrequency configuration parameters.

Module version 2.6.8
2019-04-18

- ASCSTBM-404 Fixed known issue: PORT-API-OPTION has wrong DEST value for MeasurementNotification PORT-REF.

Module version 2.6.7
2019-03-22

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

Module version 2.6.6
2019-02-15

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

Module version 2.6.5
2019-01-25

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

Module version 2.6.4
2018-12-13

- ASCSTBM-370 Fixed known issue: Runnable entities with the same SHORT-NAME are generated twice when slave and master are configured.

Module version 2.6.3
2018-11-23

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

Module version 2.6.2
2018-10-26

- Use of GPT counter as source for the virtual local time base.
- ASCSTBM-350 Fixed known issue: StbM_GetRateDeviation() returns E_OK even if the slave rate correction was not calculated.
- ASCSTBM-268 Fixed known issue: Wrong order of StbM_SyncRecordTableBlockType elements leads to wrong element data.

Module version 2.6.1
2018-09-28

- ASCSTBM-296 Fixed known issue: Progression of time fails for timeouts greater than 16 seconds.
- ASCSTBM-312 Fixed known issue: Rte generates incompatible prototypes for StbM functions.
- ASCSTBM-340 Fixed known issue: Wrong order of StbM_SyncRecordTableBlockType elements leads to wrong element data.

Module version 2.6.0
2018-09-20

- Enhance precision of Global Time.

Module version 2.5.0
2018-08-24

- Add pure local time base support.

Module version 2.4.0
2018-07-27

- Provide means to store/load the time in NvM.

Module version 2.3.2
2018-06-22

- Introduce S/R StbM_StatusNotification interface.
- StbM_SetGlobalTime behaviour for offset time bases
- ASCSTBM-309 Fixed known issue: StbM_MasterRateCorrectionData[] has an unprotected write access in StbM_SetGlobalTimeOffset()
- ASCSTBM-310 Fixed known issue: StbM_OffsetSign[] has an unprotected write access in StbM_SetGlobalTimeOffset()
- ASCSTBM-311 Fixed known issue: StbM_RecordTableBlocksInfo[] has an unprotected read/write access in StbM_UpdateVirtualLocalTimeOs()

Module version 2.3.1
2018-05-25

- Introduce StbM_UpdateGlobalTime() API
- ASCSTBM-300 Fixed known issue: If StbM uses Rte, TSyn modules do not have access to function prototypes

Module version 2.3.0
2018-04-20

- ASCSTBM-290 Fixed known issue: Rate deviation returned by StbM_GetRateDeviation is wrong in some cases
- Update signature of StbM_BusSetGlobalTime to use measureDataPtr instead of syncToGateway
- ASCSTBM-285 Fixed known issue: Os counter wrap-around is not considered in case of preemption
- ASCSTBM-292 Fixed known issue: StbMEthHwTimestampBehaviourSelect parameter cannot be enabled because of compilation error

Module version 2.2.1
2018-03-16

- ASCSTBM-232 Fixed known issue: StbM module generation fails if modules that support hardware time stamp are not configured first in the StbMBswModule list
- ASCSTBM-271 Fixed known issue: Symbolic names are not generated for StbMSynchronizedTimeBaseIdentifier

Module version 2.2.0
2018-02-16

- Provide actual precision of individual time bases

Module version 2.1.1
2018-01-19

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

Module version 2.1.0
2017-12-15

- Notification interface for global time synchronization events
- Provide rate correction in addition to pure offset correction
- ASCSTBM-245 Fixed known issue: TIMELEAP detection fails

Module version 2.0.9
2017-09-22

- ASCSTBM-213 Fixed known issue: StbM compilation fails if triggered customers are used without local time
- Added immediate Time Synchronization
- Implement updates of AUTOSAR 4.2.2
- Updated signature of API StbM_GetOffset() to AUTOSAR 4.3
- Updated signature of API StbM_SetOffset() to AUTOSAR 4.3
- Updated for MISRA-C:2012 compliance

Module version 2.0.8
2017-03-31

- ASCSTBM-168 Fixed known issue: StbM does not update local time if exclusive area locks interrupts
- ASCSTBM-177 Fixed known issue: StbM uses incorrect conversion macro for Os counter

Module version 2.0.7
2016-11-04

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

Module version 2.0.6
2016-09-09

- ASCSTBM-138 Fixed known issue: StbM generates service ports with unknown CLIENT-SERVER-OPERATION

Module version 2.0.5
2016-08-05

- The first invocation of StbM_BusSetGlobalTime() does not perform time leap detection (status bit TIMELEAP)
- Added user data support

Module version 2.0.4
2016-05-25

- ASCSTBM-104 Fixed known issue: Os timestamp functions used without considering the availability of these functions
- ASCSTBM-107 Fixed known issue: Unresolved possible error references in StbM_swc_interfaces.arxml lead to warnings during import
- Internal improvements of the StbM code generator

Module version 2.0.3
2016-02-05

- Added support for BSW modules without Module Properties file

Module version 2.0.2
2015-11-06

- Added check to validate the existence of a configured BSW module

Module version 2.0.1
2015-06-19

- Add support for ComTransformer
- ASCSTBM-82 Fixed known issue: StbM uses wrong references to ImplementationDataTypes
- ASCSTBM-83 Fixed known issue: StbM always using the type EthTSyn_SyncStateType
- ASCSTBM-84 Fixed known issue: Update of time might fail for time bases using a local time and a triggered customer

Module version 2.0.0
2015-02-20

- Initial AUTOSAR 4.2 version

Module version 1.0.0
2013-02-12

- Initial Release

3.3.4.2. New features

- Implement Multicore Distribution.

3.3.4.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- Avoidance of time leaps

  Description:

  `StbM` allows to avoid leaps in the global time when using the APIs `StbM_SetGlobalTime()` and `StbM_BusSetGlobalTime()`.

  Calling `StbM_SetGlobalTime()` or `StbM_SetGlobalTime()` for an offset time base (instead of a synchronized time base) lets the time of the underlying synchronized time base unchanged. The difference of the absolute time passed via API and the actual time of the synchronized time base is stored as offset value.

  Calling `StbM_SetCurrentTime()` for an offset time base returns the sum of both values (and therefore the absolute time) again.

  Note that the new absolute time must not be smaller than the actual time of the synchronized time base.

  Rationale:

  This enhancement allows to avoid time leaps by splitting an absolute time in `SYNC` and `OFFSET` messages.

- Support of StbM_StatusNotification Sender-Receiver Interface

  Description:

  `StbM` module can now notify the CDD and also the Application when an event was detected.

  StbM will only notify the CDD through `StbM_StatusNotificationCallback{TBName}`, if `StbMNotificationInterface` equals `CALLBACK`. StbM will only notify the Application through `Rte_Write_GlobalTime_StatusEvent_{TBName}_eventNotification`, if `StbMNotificationInterface` equals `SR_INTERFACE`. StbM will notify the CDD and also the Application if `StbMNotificationInterface` equals `CALLBACK_AND_SR_INTERFACE`.

  Rationale:

  Through this enhancement, other applications can read the status, through Rte_Read_GlobalTime_StatusEvent_{TBName}_eventNotification Sender-Receiver Interface.

- Implement VSwt support in StbM

Description:

StbM module can provide data of selected time bases to a shared memory location.

MemMap section : VAR_SHARED_TIME_DATA

Rationale:

Through this enhancement, other users can read the time base data directly from the shared memory location.

- Implementation updated to handle invalid Time Stamp or User Data

  Description:

  StbM module throws a DET error in case one of the followin APIs: - StbM_SetGlobalTime() - StbM_UpdateGlobalTime() - StbM_SetOffset() - StbM_BusSetGlobalTime() are called with a parameter timeStamp that contains invalid elements (e.g. nanoseconds part > 999999999 ns) or with an invalid value of parameter userData, i.e. userDataLength > 3.

  Rationale:

  Implementation is according to: https://jira.autosar.org/browse/AR-3144.

3.3.4.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- Monitoring of StbM_Mainfunction() cyclic execution not supported

  Affected AUTOSAR releases:

  - R4.4 rev 0

  Description:

  StbM does not monitor the cyclic execution of API service StbM_Mainfunction().

  Rationale:

  Monitoring of the StbM_Mainfunction() cyclic execution is strictly dependent on the integration strategy and is not provided by StbM internally.

  Requirements:

  SWS_StbM_00031

- StbMLocalTimeHardware allows empty reference when StbMUseOSGetTimeStamp is enabled

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

The time source for the virtual local time, will be derived from Os, if one of the following steps are made: - Configure StbMLocalTimeHardware parameter to point to an OsCounter or - Set StbMUseOSGetTimeStamp parameter to TRUE. Changes made according to the following RFC: https://jira.autosar.org/browse/AR-3543.

Requirements:

StbM.ASR20-11.ECUC_StbM_00053_Conf

- StbMRateCorrectionMeasurementDuration shall be limited to 14 seconds

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

StbMRateCorrectionMeasurementDuration shall be limited to 14 seconds to not cause an overflow during the calculation of the Rate Correction.

Requirements:

SWS_StbM_00353 SWS_StbM_00356

- StbMClockFrequency/StbMClockPrescaler factor to convert the time of its local hardware reference clock to the actual time of the Virtual Local Time, is used only when a GPT channel is used as a time source by a Time Base.

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

StbM uses `StbMClockFrequency/StbMClockPrescaler` factor only when a GPT channel is used as a time source by a Time Base.

Rationale:

StbMClockFrequency/StbMClockPrescaler factor is not used when EthTSyn is configured as a time source, because when the time is taken from EthTSyn, it has the type StbM_TimeStampType (timeBaseStatus, nanoseconds, seconds and secondsHi) and there is no need for conversion.

StbMClockFrequency/StbMClockPrescaler factor is not used when OS is configured as a time source, because when the time is taken from OS, it is represented in ticks and a conversion macro OS_TICKS2 Unit_Counter(ticks) as specified by req. OS393 in the Os AUTOSAR SWS, is used.

StbMClockFrequency/StbMClockPrescaler factor is used when a GPT Channel is configured as a time source, because when the time is taken from Gpt, it has the type Gpt_ValueType (ticks) and a conversion from ticks into nanoseconds is needed.

Requirements:

SWS_StbM_00352

- StbMTimerStartThreshold is limited to 4290s

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

StbMTimerStartThreshold can have a maximum configured value of 4290s, in order for Gpt to be able to monitor the time after Gpt_StartTimer() is called. Changes made when ASCSTBM-424 feature was introduced.

Requirements:

StbM.ASR20-11.ECUC_StbM_00063

- StbM module does not support POST-BUILD loadable, or POST-BUILD selectable

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

POST-BUILD loadable and selectable are not supported by StbM module.

Requirements:

SWS_StbM_CONSTR_00001, SWS_StbM_CONSTR_00002, ECUC_StbM_00066_Conf, SWS_StbM_-00240, ECUC_StbM_00036

- Triggered Customers: The Synchronization of OsScheduleTable is based on the fact that StbM converts the nanoseconds to OsTicks

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

When OsScheduleTables are synchronized to a counter value, the conversion of the StbM local time (nanoseconds) to ticks, is made based on Os Macros.

Conversion is only done for the "nanoseconds" part of the time base, since the schedule table duration will be in the range of tens of milliseconds.

Requirements:

SWS_StbM_00303

- TimeBaseProviderNotification_Eth Client-Server Interface and its 4 Operations are generated when StbMTimeValidation is different from NULL

Affected AUTOSAR releases:

- R20-11

Description:

TimeBaseProviderNotification_Eth Client-Server Interface, together with its 4 Operations, are generated for each StbM time base, which has StbMTimeValidation configured and it's referred by an EthTSyn Time Domain. Rationale: EthTSynPortRole and all EthTSynPdelayConfig parameters have Post-Build Selectable (Post-Build Variant Value equals TRUE), that means that a TD can have different roles depending on the selected configuration (can be master on one configuration and can be slave on the other configuration) and different values for EthTSynPdelayConfig parameters on the selected configurations (can have EthTSynGlobalTimePdelayRespEnable set to FALSE on one configuration and EthTSynGlobalTimePdelayRespEnable set to TRUE on the other one). EthTSynEnableTimeValidation parameter and StbMTimeValidation container don't have Post-Build Selectable supported. EthTSyn has Post-Build Selectable while StbM does not.

Requirements:

SWS_StbM_00461 SWS_StbM_00466 SWS_StbM_00468 SWS_StbM_00522 SWS_StbM_00523

- StbM_CanSetSlaveTimingData() API will not be provided by StbM, only based on CanTSynTimeValidationSupport parameter.

Affected AUTOSAR releases:

- R19-11

Description:

StbM_CanSetSlaveTimingData() API will be provided by StbM, if there is at least 1 StbM time base which has StbMTimeValidation enabled and is refered by a SLAVE CANTSYN time domain. (CanTSyn takes care that CanTSynEnableTimeValidation to be set to TRUE on the time domain which refers an StbM time

base which has TimeValidation enabled) Rationale: In the current CanTSyn implementation, CanTSynGlobalTimeSlave container is not Post-Build Selectable yet.

Requirements:

SWS_StbM_00487

- StbM_FrSetSlaveTimingData() API will not be provided by StbM, only based on FrTSynTimeValidationSupport parameter.

Affected AUTOSAR releases:

- R19-11

Description:

StbM_FrSetSlaveTimingData() API will be provided by StbM, if there is at least 1 StbM time base which has StbMTimeValidation enabled and is refered by a SLAVE FRTSYN time domain. (FrTSyn takes care that FrTSynEnableTimeValidation to be set to TRUE on the time domain which refers an StbM time base which has TimeValidation enabled) Rationale: In the current FrTSyn implementation, FrTSynGlobalTimeSlave container is not Post-Build Selectable yet.

Requirements:

SWS_StbM_00487

- StbM_CanSetMasterTimingData() API will not be provided by StbM, only based on CanTSynTimeValidationSupport parameter.

Affected AUTOSAR releases:

- R19-11

Description:

StbM_CanSetMasterTimingData() API will be provided by StbM, if there is at least 1 StbM time base which has StbMTimeValidation enabled and is refered by a MASTER CANTSYN time domain. (CanTSyn takes care that CanTSynEnableTimeValidation to be set to TRUE on the time domain which refers an StbM time base which has TimeValidation enabled) Rationale: In the current CanTSyn implementation, CanTSynGlobalTimeMaster container is not Post-Build Selectable yet.

Requirements:

SWS_StbM_00493

- StbM_FrSetMasterTimingData() API will not be provided by StbM, only based on FrTSynTimeValidationSupport parameter.

Affected AUTOSAR releases:

- R19-11

Description:

StbM_FrSetMasterTimingData() API will be provided by StbM, if there is at least 1 StbM time base which has StbMTimeValidation enabled and is refered by a MASTER FRTSYN time domain. (FrTSyn takes care that FrTSynEnableTimeValidation to be set to TRUE on the time domain which refers an StbM time base which has TimeValidation enabled) Rationale: In the current FrTSyn implementation, FrTSynGlobalTimeMaster container is not Post-Build Selectable yet.

Requirements:

SWS_StbM_00493

- StbMOffsetCorrectionJumpThreshold and StbMOffsetCorrectionAdaptionInterval are limited

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

StbMOffsetCorrectionJumpThreshold is limited to the value of 15s. StbMOffsetCorrectionAdaptionInterval is limited to the value of 5s. This is necessary because StbM uses Q format for the calculations, and values close to 16s and bigger ca not be represented. So the time intervals, and calculated rates need to be controlled and limited.

Requirements:

ECUC_StbM_00056_Conf, ECUC_StbM_00057_Conf

- Functions StbM_GetSyncTimeRecordHead() and StbM_GetOffsetTimeRecordHead(), shall be provided by StbM through Rte, even if StbMAllowSystemWideGlobalTimeMaster is set to TRUE.

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

Functions StbM_GetSyncTimeRecordHead() and StbM_GetOffsetTimeRecordHead() shall be provided for SLAVE time bases when: - StbMAllowSystemWideGlobalTimeMaster is set to FALSE and StbMIsSystemWideGlobalTimeMaster is set to FALSE Functions StbM_GetSyncTimeRecordHead() and StbM_GetOffsetTimeRecordHead() shall be provided for GATEWAY time bases when: - StbMAllowSystemWideGlobalTimeMaster is set to FALSE and StbMIsSystemWideGlobalTimeMaster is set to FALSE - StbMAllowSystemWideGlobalTimeMaster is set to TRUE and StbMIsSystemWideGlobalTimeMaster is set to FALSE

Rationale:

This is necesary, in order for StbM_GetSyncTimeRecordHead() and StbM_GetOffsetTimeRecordHead() to be present also for GATEWAY time bases, which have StbMAllowSystemWideGlobalTimeMaster set to TRUE.

Requirements:

SWS_StbM_00225,SWS_StbM_00041

- StbM_SetRateCorrection OPERATION of the GlobalTime_Master C-S Interface, shall be provided by StbM through Rte, only for MASTER time bases which have StbMAllowMasterRateCorrection set to TRUE.

Affected AUTOSAR releases:

- R4.4 rev 0

Description:

Operation StbM_SetRateCorrection shall be provided for MASTER time bases when: - StbMAllowMaster-RateCorrection is set to TRUE

Rationale:

It make sense to set the rate correction only for Master time bases. Slave and Gateways can calculate it on their own.

Requirements:

SWS_StbM_00240

- StbMEcucPartitionRef parameter will be located under StbMEcucPartitionRefList list, instead of StbMGeneral.

Affected AUTOSAR releases:

- R20-11

Description:

StbMEcucPartitionRef shall be located under StbMEcucPartitionRefList list.

Rationale:

StbM APIs can be called from multiple partitions. A clear mapping of partitions/cores is needed for StbM to be able to support cross-core calls.

Requirements:

ECUC_StbM_00069

- StbMClockFrequency and StbMClockPrescaler is mandatory, if GPT is configured as StbMLocalTime-Hardware

Affected AUTOSAR releases:

- R20-11

Description:

StbMClockFrequency and StbMClockPrescaler shall be mandatory if GPT is configured as StbMLocal-TimeHardware.

Rationale:

StbMClockFrequency/StbMClockPrescaler factor will be used when converting Gpt ticks to nanoseconds.

Requirements:

ECUC_StbM_00051, ECUC_StbM_00052

- StbM_TimeSyncType will be of type uint8, instead of enumeration and it will be provided by Rte when available or by StbM.

Affected AUTOSAR releases:

- R20-11

Description:

StbM_TimeSyncType is of type uint8 and it will be provided by RTE when available or by StbM.

Rationale:

As per BSWM_C_020 rule from Software Construction Guidelines, enums SHALL NOT be used. • Complete control over actual size of variable of "enum" type • Blends well with all use-cases where the actual size of the data type is required

Requirements:

SWS_StbM_91009

- StbM_ProtocolParamType type will be provided by Rte when available or by StbM.

Affected AUTOSAR releases:

- R20-11

Description:

StbM_ProtocolParamType type will be provided by RTE when available or by StbM.

Rationale:

StbM cannot force the user to activate RTE, whenever an EthTSyn time domain is referrencing an StbM time base.

Requirements:

SWS_StbM_91010

3.3.4.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- Limitation on number of Os counters

  Description:

  The `StbM` can only handle one Os counter. This limitation applies to configuration parameter `StbMLocalTimeHardware`.

  Rationale:

  The alignment of multiple time bases from a single Os counter does not affect functionality or precision of the time bases.

- StbMRateCorrectionMeasurementDuration is limited

  Description:

  The `StbMRateCorrectionMeasurementDuration` parameter, and the resynchronization time of the StbM should be configured in a way that the following values do not exceed 15 seconds:

  - the difference (TGstop minus TGstart)
  - the difference (TVstop minus TVstart)

  These are part of the rate correction calculation formula: rrc = (TGStop – TGStart) / (TVStop- TVStart). If they do exceed the above value, an overflow could emerge when calculating the Rate Correction.

  This can be achieved by having a synchronization interval smaller than 15s. If the limit is exceeded StbM will not calculate the rate correction and the previous one will be used.

  Rationale:

The application does not use, float, doubles, or uint64(multiplication and division) values. Calculation is only controlled by types smaller or equal with uint32. This limitation allows for higher precision.

- StbMOffsetCorrectionAdaptionInterval is limited

  Description:

  The `StbMOffsetCorrectionAdaptionInterval` parameter, shall be configured in a way that roc does not exceed a value of 2.068: roc = (TG – TLSync) / (TCorrInt).

  In this case StbMOffsetCorrectionJumpThreshold / StbMOffsetCorrectionAdaptionInterval should be smaller than 2.068, since StbMOffsetCorrectionJumpThreshold is the upper limit of the offset.

  `StbMOffsetCorrectionAdaptionInterval` shall not exceed 5 sec.

  `StbMOffsetCorrectionJumpThreshold` shall not exceed 15 sec.

  These are part of the rate correction calculation formula: rrc = (TGStop – TGStart) / (TVStop- TVStart). If they do exceed the above value, an overflow could emerge when calculating the Rate Correction.

  StbM makes sure that the limit is no violated, and if a violation is detected at runtime, StbM will calculate the time without using the rate.

  Rationale:

  The application does not use values of type float, double or uint64(multiplication and division). Calculation is only controlled by types smaller or equal with uint32. This limitation allows for higher precision.

- Calculating Master Local Time is limited

  Description:

  The following shall be taken into consideration when configuring the StbM module.

  When calculating the local time with rrc for Master side the following formula is used:

  TL = TLSync + (TV - TVSync) * rrc

  It shall be considered that the sum (or the difference) between the two virtual local time values shall not exceed 15 seconds.

  Rationale:

  The application does not use values of type float, double or uint64. Calculation is only controlled by types smaller or equal with uint32. This limitation allows for higher precision.

- Subtracting Virtual Local Times is limited

  Description:

The following shall be taken into consideration:

Subtracting two virtual local times shall be limited to a difference of 2.305.843.004s and 918.726.656ns

This limitation is located in the StbM_SubtractVirtualLocalTimes() function

Rationale:

Summing more would cause an overflow of seconds, this should not happen in a vehicle life time

- Limitation on the number of TSyn MASTER Time Domains referencing an StbM Time Base

Description:

The `StbM` can only be referenced by 1 TSyn MASTER Time Domain. This limitation applies to the following configuration parameters: - in case the Time Domain belongs to CANTSYN: CanTSynSynchronizedTime-BaseRef - in case the Time Domain belongs to FRTSYN: FrTSynSynchronizedTimeBaseRef - in case the Time Domain belongs to ETHTSYN: EthTSynSynchronizedTimeBaseRef

Rationale:

TIME VALIDATION feature (ASCSTBM-365) does not handle StbM Time Bases which are referenced by more then 1 TSyn MASTER Time Domain.

- Limitation on the number of TSyn SLAVE Time Domains referencing an StbM Time Base

Description:

The `StbM` can only be referenced by 1 TSyn SLAVE Time Domain. This limitation applies to the following configuration parameters: - in case the Time Domain belongs to CANTSYN: CanTSynSynchronizedTime-BaseRef - in case the Time Domain belongs to FRTSYN: FrTSynSynchronizedTimeBaseRef - in case the Time Domain belongs to ETHTSYN: EthTSynSynchronizedTimeBaseRef

Rationale:

TIME VALIDATION feature (ASCSTBM-365) does not handle StbM Time Bases which are referenced by more then 1 TSyn SLAVE Time Domain

- Limitation regarding the supported Gpt Autosar version

Description:

The `StbM` only supports 4.0 GPT version.

Rationale:

As per current implementation, StbM does not support GPT 4.2 version.

- Limitation on the number of StbM_StartTimer() calls

Description:

`StbM_StartTimer()` function shall be called at least 1 time and maximum `StbMTimerStartMaxNumberOfCalls` times.

Rationale:

StbM_StartTimer() function shall be called at least 1 time, because there shall be at least one call, in order for Notification of Customers feature to be used. StbM_StartTimer() function shall be called maximum StbMTimerStartMaxNumberOfCalls times, because each time a call to StbM_StartTimer() occurres, the StbM shall calculate, save and sort the 'customerTimerExpireTime' in a list and dynamic memory allocation shall be avoided when doing that.

- Limitation on the calculation of the deviation for Notification Customers

Description:

The calculated deviation reported to the Notification Customer through the notification callback, does not include the decoupling time between the interruption from the Gpt and the time the above notification is called.

Rationale:

The Autosar requirements do not cover this part. The decoupling time was not included in the deviation because the use cases are not clear and could be different for each user.

- Limitation on multiple Gpt drivers support

Description:

StbM does not support referencing channels from multiple Gpt drivers.

Rationale:

The Autosar requirements do not cover this part. Typically there is only one Gpt Driver per project.

- Limitation on Offset Time bases

Description:

StbM does not support Time Validation on Offset Time bases.

Rationale:

The Autosar requirements do not cover this part. No Rte C-S Interfaces are generated.

- OsScheduleTable synchronization is supported only on the core the StbM_MainFunction is mapped

Description:

A schedule table will be synchronized to a counter value, only on the core the StbM_MainFunction is mapped.

Rationale:

If a Triggered Customer is configured, the Synchronized Time-Base Manager shall monitor the cyclic execution of the StbM_MainFunction(). The synchronization of the Schedule Table is performed from StbM_MainFunction(). This is to guarantee cyclic synchronization of OS schedule tables.

- Only GlobalTime_Master and GlobalTime_Slave C-S Interfaces support Multicore in an efficient way

  Description:

  The following service interfaces support multicore in an efficient way: - GlobalTime_Master (with all its operations) - GlobalTime_Slave (with all its operations) All other services benefit from the already available cross-core call mechanisms from the Rte.

- Limitation for 16bit Gpt counters

  Description:

  16bit timers will/could expire before StbM can make an update of the Virtual Local Time in the StbM_MainFunction.

  Rationale:

  The update in StbM is done every 3 seconds.

3.3.4.6. Open-source software

StbM does not use open-source software.

4. ACG8 Time Sync user guide

4.1. Overview

The ACG8 Time Sync user guide provides information about the concepts of the time synchronization in the AUTOSAR context.

- Section 4.2, “Background information” describes the concept of time synchronization in the AUTOSAR context.
- Section 4.3, “Configuring the ACG8 Time Sync” provides basic information about using the time synchronization modules in your project.

4.2. Background information

This chapter provides general information about the time synchronization concepts in the AUTOSAR context. If you are not familiar with the general concepts of AUTOSAR, read the general information provided in the EB tresos AutoCore Generic documentation first.

4.2.1. Clock synchronization between master and slave

The main functionality of time synchronization is to provide a common time base for different ECUs in a network. One particular ECU is configured to be the time master which provides its time base to other ECUs in the same network. The other time slave ECUs synchronize to the time master and update their local time to match the master's time base.

The process of time synchronization is depicted in Figure 4.1, “Time Synchronization via `Sync` and `Followup` frames” and consists of these steps:

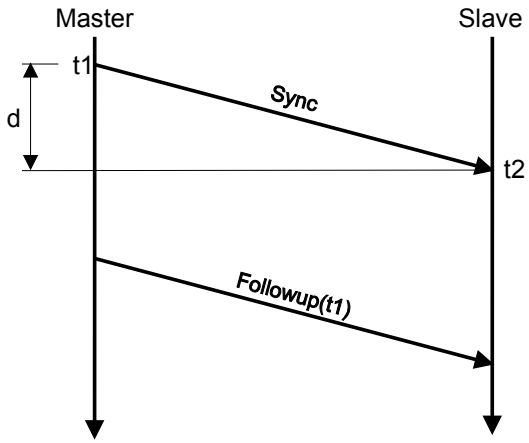

Figure 4.1. Time Synchronization via `Sync` and `Followup` frames

- The master periodically sends `Sync` frames to the slave.
- The slave measures its local time (`t2`) at the time the frame is received.
- The master sends a `Followup` frame which contains the time `t1` at which the `Sync` frame was sent.
- The slave computes the offset between its local time base and the master's time base.
  This offset is `t2-t1-d` where `d` is the so-called propagation delay (also called link delay or path delay) that is introduced on the network (e.g. by routers and switches used). If this propagation delay is not known it is assumed to be zero, leading to an inaccuracy of the computed offset. The propagation delay can be determined by the feature 'propagation delay measurement' (see Section 4.2.3, “Propagation delay measurement”).
- The slave updates its local time based on the computed offset.

4.2.2. Clock rate correction

Using clock synchronization between master and slave ECUs will regularly synchronize the slave time to the master time. However, if both clocks do not run at the same frequency there will continuously be an increasing time offset between master and slave. This inaccuracy can be reduced if the slave determines the offsets of two subsequent synchronizations and computes a correction for its clock rate. The clock rate is then updated in a way that the expected offset for the next synchronization period is zero. The `EthTSyn` module reports the measured time offsets to the `Eth` module which then updates its clock rate if this is supported by the Ethernet controller.

4.2.3. Propagation delay measurement

The time synchronization between master and slave can be improved if the propagation delay is known. The propagation delay is the time between transmission of the frame by the master and the reception of the frame by the slave. There is a specific value for each slave that can be measured by the following mechanism (see Figure 4.2, “Propagation delay measurement”):

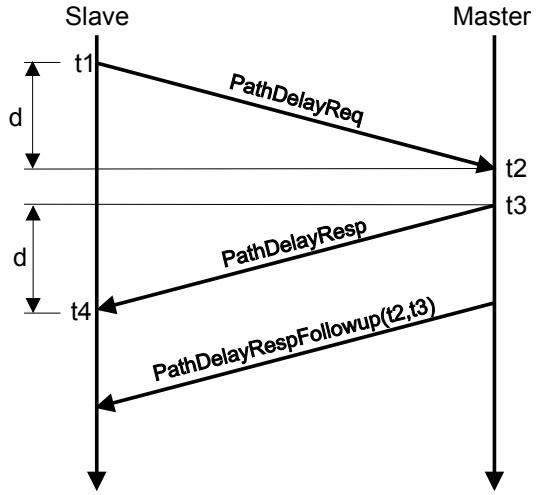

Figure 4.2. Propagation delay measurement

- The slave sends a `PathDelayReq` frame to the master. The slave stores time `t1` at which the frame was transmitted, the master stores time `t2` at which the frame was received.
- The master replies with a `PathDelayResp` frame. It saves time `t3` at which the frame was transmitted, the slave stores time `t4` at which the frame was received.
- The master sends a `PathDelayRespFollowup` frame which includes `t2` and `t3` in the frame payload. Now the slave can compute the propagation delay which is `((t4-t1)-(t3-t2))/2`.
- The slave uses the propagation delay during time synchronization with the master to improve the accuracy of the offset calculation.

NOTE

ACG8 Time Sync time masters also transmit `PathDelayReq` frames

In ACG8 Time Sync, the time masters also transmit `PathDelayReq` frames. This avoids compatibility issues with some AVB switches.

4.2.4. Hardware time-stamping

If the Ethernet controller has an integrated clock which provides a hardware time-stamp for incoming Ethernet frames the accuracy of time synchronization can be increased. EB tresos AutoCore supports `Eth` drivers with and without hardware time-stamping support.

4.2.5. Time synchronization gateway functionality

A single gateway ECU can act as time slave in one network and as time master in another network. This way, a global time can be propagated from one network to another network. EB tresos AutoCore supports time synchronization gateway functionality.

4.2.6. Time synchronization priority announcement

Time master ECUs are expected to send `Announce` frames to report their priority and accuracy with respect to time synchronization. This information could be used to determine the time master with the most accurate clock and elect this time master as global time master (the grand master). Transmitting `Announce` frames may also avoid compatibility issues with some AVB switches which expect to receive these frames. EB tresos AutoCore supports configurable transmission of `Announce` frames.

4.2.7. Multi-core support

4.2.7.1. Multi-core use without partitioning

When multiple cores are used without partitioning, the `StbM` needs to process APIs reporting current time-stamps without any delay directly in the caller context. This means all these APIs are executed in different contexts and `StbM` must protect the access to global data with multi-core capable means.

WARNING

All shared data must be protected from mutual access

To protect all shared data from mutual access, the configuration of spinlocks is mandatory. Consider the information in Section 5.5.3, “Integration notes” about the following exclusive areas:

- SCHM_STBM_EXCLUSIVE_AREA_0
- SCHM_STBM_EXCLUSIVE_AREA_1
- SCHM_STBM_EXCLUSIVE_AREA_2

If you do not configure the spinlocks, you risk the corruption of time-related data.

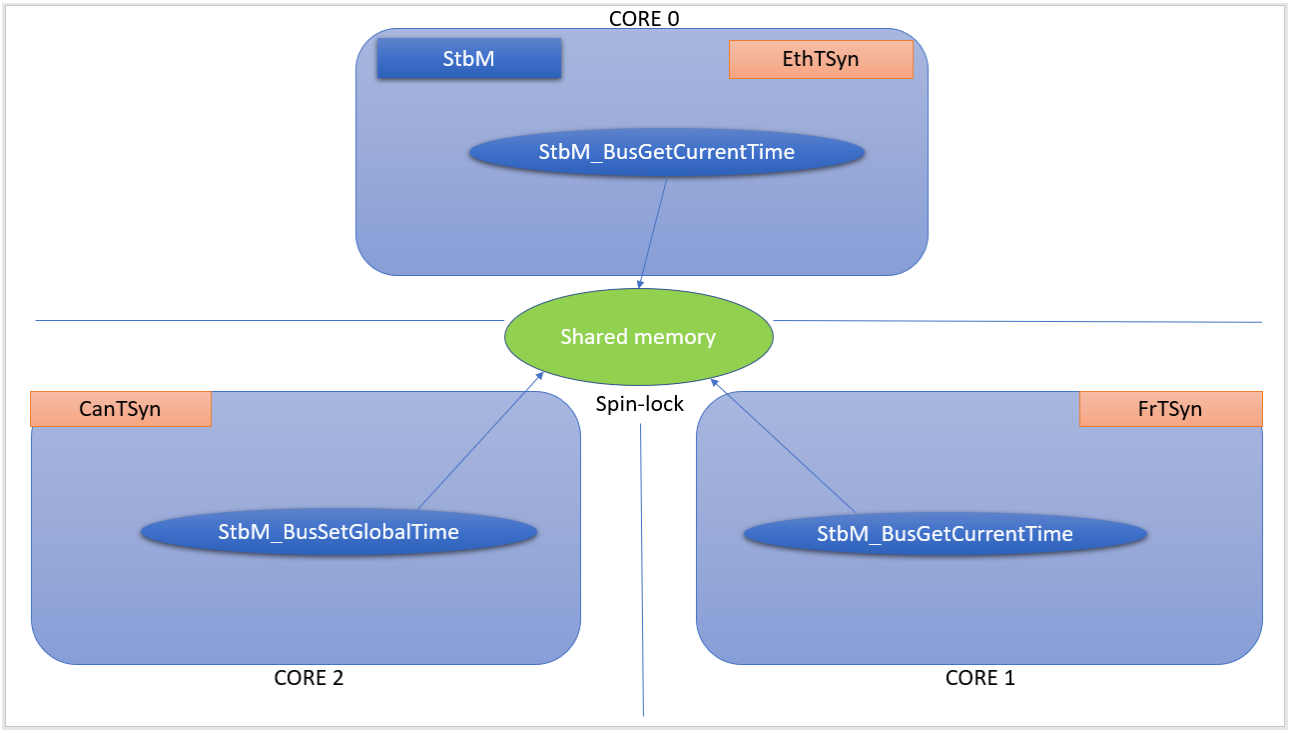

Figure 4.3. Multi-core distribution without Rte involved

`Rte` calls to `StbM` can still be used by enabling the `StbMRteUsage` parameter. In this case, the calls go through the configured `Rte` multi-core mechanisms, e.g. the Inter OS Application Communicator (IOC). This introduces delays in getting/setting the time. These delays can be drastically reduced by enabling the `StbMMulticoreSupport` parameter (see Section 4.2.7.2, “Multi-core use with partitioning”), or by using direct calls from the integration code.

NOTE

No partitioning required if `Rte` calls from the same core as `StbM`

`Rte` calls from the same core as `StbM` do not introduce delays, so no partitioning is required.

WARNING

No Time Sync module on more than one core in the same ECU

A module from the Time Sync stack can only be present on one core in the same ECU. That means you cannot have `CanTSyn` on core 0 and core 1.

4.2.7.2. Multi-core use with partitioning

The `StbM` needs to ensure the precision of Synchronized Time Bases (i.e. the Global Time). Therefore, `StbM` must ensure that APIs reporting current timestamps are processed immediately. The APIs must support a master/satellite approach. `StbM` does a synchronous processing directly in the caller context. This means all these APIs are executed in different contexts. So `StbM` must protect the access to global data with multi-core capable means.

See the `Integration notes` in the module references for details on the following exclusive areas:

- SCHM_STBM_EXCLUSIVE_AREA_0
- SCHM_STBM_EXCLUSIVE_AREA_1
- SCHM_STBM_EXCLUSIVE_AREA_2

Also consider the following integration requirements listed in the `Integration notes`:

- Section 5.5.3.4.23, “lim.StbM.EB_INTREQ_StbM_0023”
- Section 5.5.3.4.24, “lim.StbM.EB_INTREQ_StbM_0024”
- Section 5.5.3.4.25, “lim.StbM.EB_INTREQ_StbM_0025”

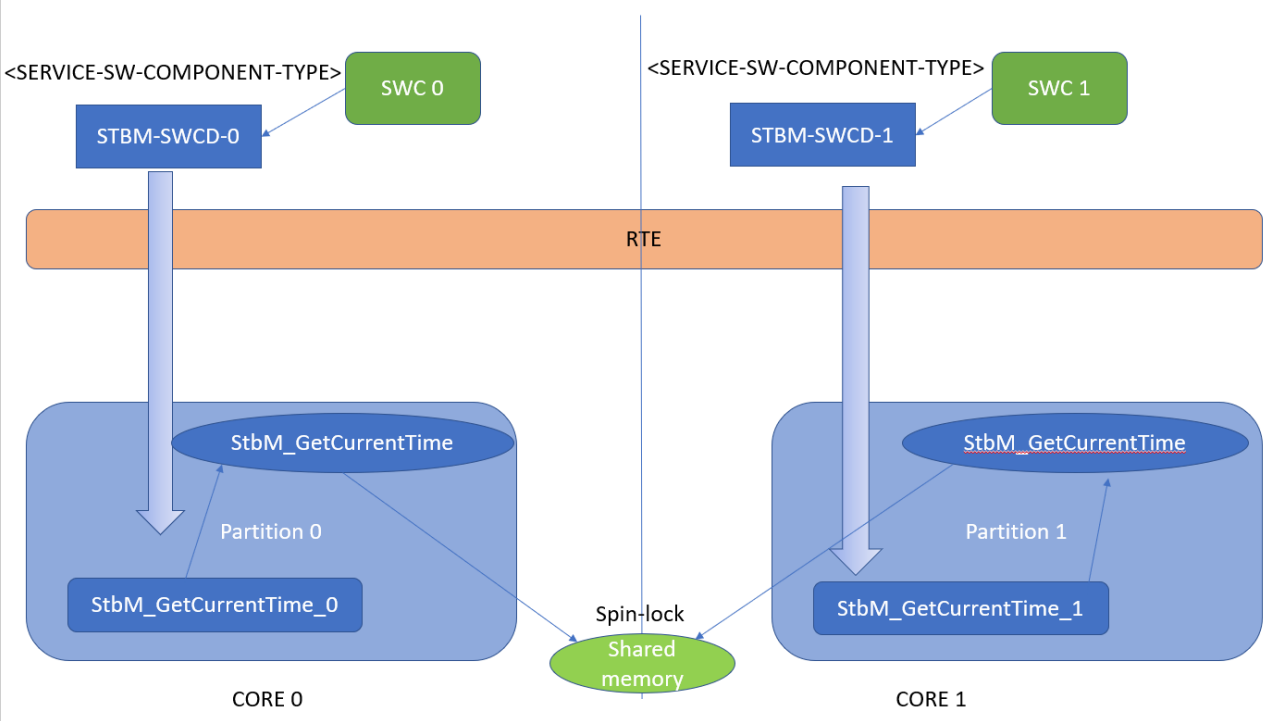

Figure 4.4. Multi-core distribution with RTE involved

For each partition, `StbM` generates a SWCD. Every time a call to a C/S interface, e.g. `Rte_Call_[SWC_-shortname][PORT_name][OPERATION_name]`, is made in the context of a certain partition, the corresponding `StbM` API is called, by calling a specific wrapper in the same partition context.

4.2.7.2.1. Multi-core supporting service interfaces

The following service interfaces support multi-core distribution:

- GlobalTime_Master
- GlobalTime_Slave

For more details, see the following table.

Direction	Port type	Port name	API (AUTOSAR 4.4)	StbM multi-core capable from SWC	Data access	HW clock access	Comment
Provided	Client-Server	GlobalTime_Master	GetMasterConfig	YES			Does not use spinlock. Access only ROM data.
			SetGlobalTime	YES	X	X	Needed to start master time base or set time from another time source
			SetOffset	YES	X	X	Set time from another time source
			SetRateCorrection	YES	X	X	Combined with measurements to calculate extra rate correction
			SetUserData	YES	X		To share MetaData, e.g. can be used to manage reset of master
			TriggerTimeTransmission	YES	X		To apply update of MetaData
			UpdateGlobalTime	YES	X	X	Needed to start master time base or set time from another time source
Provided	Client-Server	GlobalTime_Slave	GetCurrentTime	YES	X	X	Get current time (update virtual local time)
			GetCurrentTimeExtended	YES	X	X	Get current time (update virtual local time)
			GetOffsetTimeRecordHead	YES	X		Does not use spinlock. Access only ROM data.
			GetRateDeviation	YES	X		Obtain rate correction
			GetSyncTimeRecordHead	YES	X		Does not use spinlock. Access only ROM data.
			GetTimeBaseStatus	YES	X		Needed to monitor time base (sync time base can be got using GetCurrentTime but not the offset time base status)
			GetTimeLeap	YES	X		
Provided	Sender-Receiver	GlobalTime_StatusEvent	StatusNotification	NO			Callback in main function. Event in time base. Depends on tresos conf. (e.g. on reception of bus timestamp). Does not use spinlock.
Provided	Client-Server	StartTimer	StartTimer	NO	X	X	Timer feature
Required	Client-Server	GlobalTime_TimeEvent	TimeNotification	NO	CBK	CBK	Callback in main function. Notification of a timeout (timer feature).
Required	Client-Server	GlobalTime_Measurement	SetOffsetTimeRecordTable	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
			SetSyncTimeRecordTable	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
Required	Client-Server	TimeBaseProviderNotification_Eth	SetMasterTimingData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
			SetPdelayInitiatorData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
			SetPdelayResponderData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
			SetSlaveTimingData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
Required	Client-Server	TimeBaseProviderNotification_Fr	SetMasterTimingData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
			SetSlaveTimingData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
Required	Client-Server	TimeBaseProviderNotification_Can	SetMasterTimingData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.
			SetSlaveTimingData	NO	CBK	CBK	Callback in main function. Transmit measurement data to APP for storage.

Figure 4.5. Multi-core capable service interfaces

4.2.7.2.2. Multi-core supporting APIs

The following table shows which APIs support multi-core distribution:

API (AUTOSAR 4.4)	StbM multi-core capable	Data access	HW clock access	Comment
StbM_GetVersionInfo()	yes			Does not use spinlock. Access only ROM data.
StbM_Init()	no	X	X	
StbM_GetCurrentVirtualLocalTime()	yes	X	X	Get/update virtual local time. Used by <Bus>TSyn.

API (AUTOSAR 4.4)	StbM multi-core capable	Data access	HW clock access	Comment
StbM_GetOffset()	yes	X	X	Get offset time base time. Used by <Bus>TSyn.
StbM_BusGetCurrentTime()	yes	X	X	Get current time. Used by <Bus>TSyn.
StbM_BusSetGlobalTime()	yes	X	X	Set current time. Used by <Bus>TSyn.
StbM_GetTimeBaseUpdateCounter()	yes	X		Used for immediate transmission. Used by <Bus>TSyn.
StbM_MainFunction()	no	X	X	
StbM_CanSetSlaveTimingData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_CanSetSlaveTimingData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_FrSetSlaveTimingData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_EthSetSlaveTimingData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_CanSetMasterTimingData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_FrSetMasterTimingData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_EthSetPdelayInitiatorData()	yes	X		Used for time validation. Used by <Bus>TSyn.
StbM_EthSetPdelayResponderData()	yes	X		Used for time validation. Used by <Bus>TSyn.

For information on how to configure the multi-core support, see Section 4.3.1, “Configuring multi-core distribution in StbM”.

4.2.8. Access to protocol parameters

The Time Sync stack supports setting/getting information related to bus-specific protocol parameters. The following services can be used:

- `StbM_SetBusProtocolParam()`
  This service can be called only on a time master or a time gateway. If it is called on a time gateway, it sets the parameters of the master part of the gateway.
- `StbM_GetBusProtocolParam()`
  This service can be called only on a time slave or a time gateway. If it is called on a time gateway, it obtains the parameters from the slave part of the gateway.

The following parameters can be changed/retrieved:

- `cumulativeScaledRateOffset`: the cumulative rate offset of the time master according to IEEE 802.-1AS
- `gmTimeBaseIndicator`: the time base indicator of the current global time master according to IEEE 802.1AS
- `lastGmPhaseChange`: the phase change of the current global time master according to IEEE 802.1AS
- `scaledLastGmFreqChange`: the scaled last frequency change of the global time master according to IEEE 802.1AS

NOTE **Limited to Ethernet**

Currently this feature is supported by AUTOSAR only for `EthTSyn`.

4.2.9. Reliable TxConfirmation support

The Time Sync stack supports the use of reliable TxConfirmation in `CanTSyn` and `EthTSyn`. The `CanTSyn_TxConfirmation()` and `EthTSyn_TxConfirmation()` APIs have an extra parameter called `result`. Based on this parameter, the module can judge the correctness of the confirmation and react accordingly.

The TxConfirmation is ensured by the lower layers, so the TSyn modules do not need a time-out to monitor the confirmation anymore.

WARNING **Dependency on lower layers**

For the TxConfirmation feature to work, the lower layers also need to support this feature (i.e. `CanIf`, `Can`, `EthIf`, `Eth`).

Elektrobit `CanIf` supports this feature, while `EthIf` does not yet support it.

4.3. Configuring the ACG8 Time Sync

This section gives brief instructions for integrating the time synchronization modules in your project.

To use time synchronization, you must configure two modules. The first is the `StbM` module. The second is a network-specific module, for example the `EthTSyn` module.

You find information about configuring the modules in the parameter descriptions provided in the module references section Chapter 5, “ACG8 Time Sync module references“. You also see this information in EB tresos Studio in the **Properties** tab. Click on the parameter name and select the **Description** page of the **Properties** tab to read the parameter description.

You find additional information about integrating the modules in the `Integration notes` section of the module references. The integration notes provide information about exclusive areas used by the module, production errors and memory mapping.

4.3.1. Configuring multi-core distribution in StbM

For background information on multi-core support in ACG8 Time Sync, see Section 4.2.7, “Multi-core support”.

Configuring the StbM

Step 1
Enable the `StbMMulticoreSupport` configuration parameter.

Step 2
Enable the `StbMRteUsage` configuration parameter.

Step 3
In `StbMEcucPartitionRefList`, configure references to `EcuC` partitions. Each partition has a reference to a /EcucPartitionCollection/EcucPartition in `EcuC`, which is then mapped to an OsApplication.

4.3.2. Configuring reliable TxConfirmation in CanTSyn

For background information on reliable TxConfirmation support in ACG8 Time Sync, see Section 4.2.9, “Reliable TxConfirmation support”.

Configuring the CanTSyn

Step 1
Enable the `CanTSynReliableTxConfirmation` configuration parameter.

Configuring the CanIf

Step 1
Enable the `CanIfPublicTxConfResultSupport` configuration parameter.

4.3.3. Configuring reliable TxConfirmation in EthTSyn

For background information on reliable TxConfirmation support in ACG8 Time Sync, see Section 4.2.9, “Reliable TxConfirmation support”.

Configuring the EthTSyn

Step 1
Enable the `EthTSynReliableTxConfirmationSupport` configuration parameter.

5. ACG8 Time Sync module references

5.1. Overview

This chapter provides module references for the ACG8 Time Sync product modules. These include a detailed description of all configuration parameters. Furthermore this chapter lists the application programming interface with all data types, constants and functions.

The content of the sections is sorted alphabetically according the EB tresos AutoCore Generic module names.

For further information on the functional behavior of these modules, refer to the chapter ACG8 Time Sync user's guide.

5.1.1. Notation in EB module references

EB notation may differ from the AUTOSAR standard notation in the software specification documents (SWS). This section describes the notation of *default value* and *range* fields in the EB module references.

5.1.1.1. Default value of configuration parameters

If there is no default value specified for a parameter, the default value field is omitted to prevent ambiguity with parameters that have `--` as default values.

Example: The parameter `BswMCompuConstText` of the `BswM` module of EB tresos AutoCore Generic 8 Mode Management has no default value field, therefore it is omitted.

5.1.1.2. Range information of configuration parameters

The range of a configuration parameter contains an upper and a lower boundary. However, in special cases the range of allowed values can be computed by means of an XPath function that is evaluated at configuration time. An XPath function can either be a standard `xpath:<function>()` or a custom `cxpath:<function>()` function. The range of a configuration parameter may be computed based on other configuration parameters that are referenced from the XPath function. For more information on custom XPath functions, see section *Custom XPath Functions API* of the EB tresos Studio developer's guide.

Example: The parameter `BswMCompuConstText` of the `BswM` module of EB tresos AutoCore Generic 8 Mode Management has the custom XPath function `cxpath:getCompuMethodsVT()` in the range field which provides the allowed values.

5.2. CanTSyn

5.2.1. Configuration parameters

Containers included		
Container name	**Multiplicity**	**Description**
CanTSynGeneral	1..1	This container contains the general configuration parameters of the CanTSyn module.
CanTSynGlobalTimeDomain	1..n	This represents the existence of a global time domain on CAN. The CanTSyn module can administrate several global time domains at the same time that in itself form a hierarchy of domains and sub-domains.
CommonPublishedInformation	1..1	**Label:** Common Published Information Common container, aggregated by all modules. It contains published information about vendor and versions.
PublishedInformation	1..1	**Label:** EB Published Information Additional published parameters not covered by CommonPublishedInformation 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	VariantPreCompile
Range	VariantPreCompile

Configuration class	**VariantPreCompile:**	VariantPreCompile

5.2.1.1. CanTSynGeneral

Parameters included	
Parameter name	**Multiplicity**
CanTSynDevErrorDetect	1..1
CanTSynMainFunctionPeriod	1..1
CanTSynVersionInfoApi	1..1
CanTSynReliableTxConfirmation	1..1
CanTSynTimeValidationSupport	1..1

Parameter Name	**CanTSynDevErrorDetect**	
Description	Switch for enabling the development error detection.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynMainFunctionPeriod**	
Description	Schedule period of the main function CanTSyn_MainFunction(). Unit: [s].	
Multiplicity	1..1	
Type	FLOAT	
Range	<=1000 >0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynVersionInfoApi**
Description	Activates the CanTSyn_GetVersionInfo() API.
Multiplicity	1..1
Type	BOOLEAN

Default value	FALSE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynReliableTxConfirmation**	
Description	If this parameter is set, CanTSyn will support CanTSyn_TxConfirmation() according to AUTOSAR R20-11. Additional parameter result was added, which will provide the status of the transmission: E_OK: The transmission was successful E_NOT_OK: The transmission failed. CanIf must support reliable TxConfirmation(CanIfPublicTxConfResultSupport parameter must be enabled and true).	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	Elektrobit Automotive GmbH	

Parameter Name	**CanTSynTimeValidationSupport**	
Description	The CanTSyn shall support Time Validation, if CanTSynTimeValidationSupport set to TRUE.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	M4_ECUC_ORIGIN_ASRR19-11	

5.2.1.2. CanTSynGlobalTimeDomain

Containers included		
Container name	**Multiplicity**	**Description**

Containers included		
CanTSynGlobalTimeFup-DataIDList	0..1	The DataIDList for FUP messages ensures the identification of data elements due to CRC calculation process.
CanTSynGlobalTimeOfns-DataIDList	0..1	The DataIDList for OFNS messages ensures the identification of data elements due to CRC calculation process.
CanTSynGlobalTimeOfs-DataIDList	0..1	The DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.
CanTSynGlobalTimeSync-DataIDList	0..1	The DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.
CanTSynGlobalTimeMaster	0..1	Configuration of the global time master. Each global time domain is required to have exactly one global time master. This master may or may not exist on the configured ECU.
CanTSynGlobalTimeSlave	0..1	Configuration of a global time slave. Each global time domain is required to have at least one time slave. The configured ECU may or may not represent a time slave.

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeDomainId	1..1
CanTSynGlobalTimeNetworkSegmentId	1..1
CanTSynGlobalTimeSecureTmacLength	1..1
CanTSynEnableTimeValidation	1..1
CanTSynSynchronizedTimeBaseRef	1..1
CanTSynUseExtendedMsgFormat	1..1

Parameter Name	CanTSynGlobalTimeDomainId	
Description	The global time domain ID.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=31 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	CanTSynGlobalTimeNetworkSegmentId

Description	This represents the numerical identifier of the network on system level scope where this Global Time has been communicated on.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=255	
	>=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
	VariantPreCompile:	VariantPreCompile
Origin	M4_ECUC_ORIGIN_ASRR20-11	

Parameter Name	**CanTSynGlobalTimeSecureTmacLength**	
Description	Represents the number of bytes for the used Truncated Message Authentication Code (TMAC). If 0, no message authentication will be used. TMAC is not supported.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=16	
	>=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynEnableTimeValidation**	
Description	CanTSyn shall do time recording for TimeValidation for this Time Domain, if CanTSynEnableTimeValidation set to TRUE and CanTSynEnableTimeValidation of this time domain is set to TRUE.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**PreCompile:**	VariantPreCompile
Origin	M4_ECUC_ORIGIN_ASRR19-11	

Parameter Name	**CanTSynSynchronizedTimeBaseRef**
Description	Mandatory reference to the required synchronized time-base.

Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynUseExtendedMsgFormat**
Description	Switches support for 16 Byte Timesync messages on or off (for CAN FD only) ► true: use 16 byte Timesync message formats (for CAN FD only). ► false: use 8 byte Timesync message formats.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

5.2.1.3. CanTSynGlobalTimeFupDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
CanTSynGlobalTimeFup-DataIDListElement	16..16	Element of the DataIDList for FUP messages ensures the identification of data elements due to CRC calculation process.

5.2.1.4. CanTSynGlobalTimeFupDataIDListElement

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeFupDataIDListIndex	1..1
CanTSynGlobalTimeFupDataIDListValue	1..1

Parameter Name	**CanTSynGlobalTimeFupDataIDListIndex**
Description	Index of the DataIDList for FUP messages ensures the identification of data elements due to CRC calculation process.
Multiplicity	1..1

Type	INTEGER	
Default value	0	
Range	<=15 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynGlobalTimeFupDataIDListValue**	
Description	Value of the DataIDList for FUP messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=255 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.2.1.5. CanTSynGlobalTimeOfnsDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
CanTSynGlobalTimeOfnsDataIDListElement	16..16	The DataIDList for OFNS messages ensures the identification of data elements due to CRC calculation process.

5.2.1.6. CanTSynGlobalTimeOfnsDataIDListElement

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeOfnsDataIDListIndex	1..1
CanTSynGlobalTimeOfnsDataIDListValue	1..1

Parameter Name	**CanTSynGlobalTimeOfnsDataIDListIndex**

Description	The DataIDList for OFNS messages ensures the identification of data elements due to CRC calculation process.
Multiplicity	1..1
Type	INTEGER
Default value	0
Range	<=15 >=0
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

Parameter Name	**CanTSynGlobalTimeOfnsDataIDListValue**
Description	The DataIDList for OFNS messages ensures the identification of data elements due to CRC calculation process.
Multiplicity	1..1
Type	INTEGER
Default value	0
Range	<=255 >=0
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

5.2.1.7. CanTSynGlobalTimeOfsDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
CanTSynGlobalTimeOfs-DataIDListElement	16..16	Element of the DataIDList for FUP messages ensures the identification of data elements due to CRC calculation process.

5.2.1.8. CanTSynGlobalTimeOfsDataIDListElement

Parameters included	
Parameter name	**Multiplicity**

Parameters included	
CanTSynGlobalTimeOfsDataIDListIndex	1..1
CanTSynGlobalTimeOfsDataIDListValue	1..1

Parameter Name	CanTSynGlobalTimeOfsDataIDListIndex	
Description	Index of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=15 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	CanTSynGlobalTimeOfsDataIDListValue	
Description	Value of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=255 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.2.1.9. CanTSynGlobalTimeSyncDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
CanTSynGlobalTimeSyncDataIDListElement	16..16	Element of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.

5.2.1.10. CanTSynGlobalTimeSyncDataIDListElement

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeSyncDataIDListIndex	1..1
CanTSynGlobalTimeSyncDataIDListValue	1..1

Parameter Name	CanTSynGlobalTimeSyncDataIDListIndex	
Description	Index for the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=15 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	CanTSynGlobalTimeSyncDataIDListValue	
Description	Value of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=255 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.2.1.11. CanTSynGlobalTimeMaster

Containers included		
Container name	**Multiplicity**	**Description**

Containers included		
CanTSynGlobalTimeMasterPdu	1..1	This container encloses the configuration of the PDU that is supposed to contain the global time information.

Parameters included	
Parameter name	**Multiplicity**
CanTSynCyclicMsgResumeTime	1..1
CanTSynGlobalTimeDebounceTime	1..1
CanTSynGlobalTimeTxCrcSecured	1..1
CanTSynGlobalTimeTxPeriod	1..1
CanTSynMasterConfirmationTimeout	1..1
CanTSynImmediateTimeSync	1..1
CanTSynTxTmacCalculated	1..1

Parameter Name	CanTSynCyclicMsgResumeTime	
Description	Defines the time where the 1st regular cycle time based message transmission takes place, after an immediate transmission before. Unit: seconds	
Multiplicity	1..1	
Type	FLOAT	
Range	<=65535 >=0.0	
Configuration class	**PreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	CanTSynGlobalTimeDebounceTime	
Description	This represents the configuration of a TX debounce time for SYNC, FUP, OFS and OFNS messages compared to a message before with the same PDU. Unit: seconds	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	CanTSynGlobalTimeTxCrcSecured
Description	This represents the configuration of whether or not CRC is supported.

Multiplicity	1..1	
Type	ENUMERATION	
Range	CRC_NOT_SUPPORTED CRC_SUPPORTED	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynGlobalTimeTxPeriod**	
Description	This represents configuration of the TX period. Unit: seconds **Note:** If the value of this configuration parameter is 0 the cyclic transmission of SYNC and OFS messages is disabled. The immediate transmission if it is enabled with CanTSynImmediateTimeSync is not affected by the value of this parameter.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynMasterConfirmationTimeout**	
Description	This represents the confirmation timeout after transmission of a SYNC message resp. OFS message. Unit: seconds. This parameter will be enabled only if CanTSynReliableTxConfirmation is FALSE, else the value of 3s will be used (for more details please check CanTSyn Autosar CPR20-11 requirement: SWS_CanTSyn_00033).	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**PreCompile:**	VariantPreCompile
Origin	Elektrobit Automotive GmbH	

Parameter Name	**CanTSynImmediateTimeSync**
Description	Enables/Disables the cyclic polling of StbM_GetTimeBaseUpdateCounter() within CanTSyn_MainFunction().

Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynTxTmacCalculated**
Description	This parameter controls whether or not TMAC calculation shall be supported. TMAC is not supported.
Multiplicity	1..1
Type	ENUMERATION
Default value	TMAC_NOT_CALCULATED
Range	TMAC_CALCULATED TMAC_NOT_CALCULATED
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

5.2.1.12. CanTSynGlobalTimeMasterPdu

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeMasterConfirmationHandleId	1..1
CanTSynGlobalTimePduRef	1..1

Parameter Name	**CanTSynGlobalTimeMasterConfirmationHandleId**
Description	This represents the handle ID of the PDU that contains the global time information.
Multiplicity	1..1
Type	INTEGER
Default value	0
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

Parameter Name	CanTSynGlobalTimePduRef	
Description	This represents the reference to the Pdu taken to transmit the global time information. The global time master of a global time domain acts as the sender of the Pdu while all the time slaves are supposed to receive the Pdu.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.2.1.13. CanTSynGlobalTimeSlave

Containers included		
Container name	**Multiplicity**	**Description**
CanTSynGlobalTimeSlaveP-du	1..1	This container encloses the configuration of the PDU that is supposed to contain the global time information.

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeFollowUpTimeout	1..1
CanTSynGlobalTimeMinMsgGap	1..1
CanTSynRxCrcValidated	1..1
CanTSynRxTmacValidated	1..1
CanTSynGlobalTimeSequenceCounterJumpWidth	1..1

Parameter Name	CanTSynGlobalTimeFollowUpTimeout	
Description	Rx timeout for the follow-up message. This is only relevant for selected bus systems. A value of zero shall disable CanTSynGlobalTimeFollowUpTimeout. Unit:seconds Unit: [s] **Note:** This parameter is only relevant for configured CanTSyn slaves of this time domain.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	AUTOSAR_ECUC

Parameter Name	**CanTSynGlobalTimeMinMsgGap**	
Description	This parameter represents the configuration of a minimum message gap time for received Timesync messages compared to a message before with the same PDU . If PDUs are received more often in between than this parameter allows, they shall be ignored. Unit: seconds CanTSynGlobalTimeMinMsgGap is not supported.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Range	<=65535 >=0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynRxCrcValidated**	
Description	Definition of whether or not validation of the CRC is supported.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	CRC_IGNORED CRC_NOT_VALIDATED CRC_OPTIONAL CRC_VALIDATED	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynRxTmacValidated**
Description	This parameter controls whether or not TMAC validation shall be supported. TMAC is not supported.
Multiplicity	1..1
Type	ENUMERATION
Default value	TMAC_NOT_VALIDATED

Range	TMAC_NOT_VALIDATED	
	TMAC_VALIDATED	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**CanTSynGlobalTimeSequenceCounterJumpWidth**	
Description	The SequenceCounterJumpWidth specifies the maximum allowed gap of the Sequence Counter between two SYNC resp. two OFS messages. Unit: [s] **Note:** This parameter is only relevant for configured CanTSyn slaves of this time domain.	
Multiplicity	1..1	
Type	INTEGER	
Default value	1	
Range	<=15	
	>=1	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.2.1.14. CanTSynGlobalTimeSlavePdu

Parameters included	
Parameter name	**Multiplicity**
CanTSynGlobalTimeSlaveHandleId	1..1
CanTSynGlobalTimePduRef	1..1

Parameter Name	**CanTSynGlobalTimeSlaveHandleId**	
Description	This represents the handle ID of the PDU that contains the global time information.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	CanTSynGlobalTimePduRef	
Description	This represents the reference to the Pdu taken to transmit the global time information. The global time master of a global time domain acts as the sender of the Pdu while all the time slaves are supposed to receive the Pdu.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.2.1.15. CommonPublishedInformation

Parameters included	
Parameter name	**Multiplicity**
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	4	
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	6
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	3
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	0
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	161
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

5.2.1.16. PublishedInformation

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

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

5.2.2. Application programming interface (API)

5.2.2.1. Macro constants

5.2.2.1.1. CANTSYN_E_INIT_FAILED

Purpose	

Value	0x04U
Description	DET error code: CanTSyn initialization failed

5.2.2.1.2. CANTSYN_E_INVALID_PDUID

Purpose	
Value	0x01U
Description	DET error code: Module called with wrong PDU or SDU ID

5.2.2.1.3. CANTSYN_E_INV_CTRL_IDX

Purpose	
Value	0x06U
Description	DET error code: Invalid Controller index

5.2.2.1.4. CANTSYN_E_NULL_POINTER

Purpose	
Value	0x03U
Description	DET error code: Invalid pointer (NULL_PTR)

5.2.2.1.5. CANTSYN_E_PARAM

Purpose	
Value	0x05U
Description	DET error code: API called with invalid parameter

5.2.2.1.6. CANTSYN_E_UNINIT

Purpose	
Value	0x02U

Description	DET error code: Module not initialized

5.2.2.1.7. CANTSYN_INSTANCE_ID

Purpose	Module instance ID.
Value	0U
Description	Defines the instance number of this module. Since multiple instances are not supported this ID is always zero.

5.2.2.1.8. CANTSYN_SID_GETVERSIONINFO

Purpose	Defines API id of function CanTSyn_GetVersionInfo().
Value	0x02U

5.2.2.1.9. CANTSYN_SID_INIT

Purpose	Defines API id of function CanTSyn_Init().
Value	0x01U

5.2.2.1.10. CANTSYN_SID_MAINFUNCTION

Purpose	Defines API id of function CanTSyn_MainFunction().
Value	0x06U

5.2.2.1.11. CANTSYN_SID_RXINDICATION

Purpose	Defines API id of function CanTSyn_RxIndication().
Value	0x42U

5.2.2.1.12. CANTSYN_SID_SETTRANSMISSIONMODE

Purpose	Defines API id of function CanTSyn_SetCurrentTime().

Value	0x03U

5.2.2.1.13. CANTSYN_SID_TXCONFIRMATION

Purpose	Defines API id of function CanTSyn_TxConfirmation().
Value	0x40U

5.2.2.2. Functions

5.2.2.2.1. CanTSyn_GetVersionInfo

Purpose	API to get the module version information.	
Synopsis	`void CanTSyn_GetVersionInfo ( Std_VersionInfoType * versioninfo );`	
Service ID	0x02	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (out)	`versioninfo`	- Pointer to return the module version information.
Description	This service returns the version information of this module.	

5.2.2.2.2. CanTSyn_Init

Purpose	Initializes the CanTSyn module.	
Synopsis	`void CanTSyn_Init ( const CanTSyn_ConfigType * configPtr );`	
Service ID	0x01	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`configPtr`	- Address of the configuration data structure.
Description	This service initializes the CanTSyn module. It shall be the first function of the module to be called.	

5.2.2.2.3. CanTSyn_MainFunction

Purpose	CanTSyn module main function.
Synopsis	`void` **`CanTSyn_MainFunction`** `( void );`
Service ID	0x06
Sync/Async	Synchronous
Reentrancy	Non Reentrant

5.2.2.2.4. CanTSyn_RxIndication

Purpose	Rx-Indication function.	
Synopsis	`void` **`CanTSyn_RxIndication`** `( PduIdType RxPduId , PduInfoType * PduInfoPtr );`	
Service ID	0x42	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`RxPduId`	- ID of the received I-PDU.
	`PduInfoPtr`	- Pointer to a buffer containing the I-PDU.
Description	This service is called by CanIf in case a reception is indicated.	

5.2.2.2.5. CanTSyn_SetTransmissionMode

Purpose	This service is used to turn on or off the TX capabilities of CanTSyn.	
Synopsis	`void` **`CanTSyn_SetTransmissionMode`** `( uint8 CtrlIdx , CanTSyn_TransmissionModeType Mode );`	
Service ID	0x03	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`CtrlIdx`	- Index of the CAN channel.
	`Mode`	- Mode to indicate if frame shall be transmitted.

5.2.2.2.6. CanTSyn_TxConfirmation

Purpose	Tx-Confirmation callback function.	
Synopsis	`void CanTSyn_TxConfirmation ( PduIdType TxPduId , Std_ReturnType result );`	
Service ID	0x07	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`TxPduId`	- ID of the I-PDU that has been transmitted.
	`result`	- E_OK: The PDU was transmitted. E_-NOT_OK: Transmission of the PDU failed.
Description	This service is called by CanIf in case a transmission is confirmed.	

5.2.3. Integration notes

5.2.3.1. Exclusive areas

This section describes the exclusive areas used by the `CanTSyn` module.

5.2.3.1.1. SCHM_CANTSYN_EXCLUSIVE_AREA_0

Protected data structures	The exclusive area protects the shared variable access of all global variables.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. The locking mechanism can be disabled if it is ensured that: ▶ `CanTSyn_MainFunction()` does not preempt `CanTSyn_RxIndication()` and vice versa.

	▶ No CanTSyn API function preempts `CanTSyn_Init()`.

5.2.3.1.2. SCHM_CANTSYN_EXCLUSIVE_AREA_1

Protected data structures	In order to provide enhanced precision of the Virtual Local Time the module CanTSyn requires this exclusive area to increase the precision of the Virtual Local Time.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. The locking mechanism can be disabled if the Time Base that is referenced in StbM uses an OsCounter as Hardware reference.

5.2.3.2. Production errors

Production errors are not reported by the `CanTSyn` module.

5.2.3.3. Memory mapping

General information about memory mapping is provided in the EB tresos AutoCore Generic documentation. Refer to the section `Memory mapping and compiler abstraction` in the `Integration notes` section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section
CONFIG_DATA_UNSPECIFIED
CONST_8
VAR_INIT_8
VAR_CLEARED_UNSPECIFIED
VAR_CLEARED_8
CODE

5.2.3.4. Integration requirements

WARNING **Integration requirements list is not exhaustive**

The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

Integration requirements are not listed for the CanTSyn module.

5.3. EthTSyn

5.3.1. 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.
EthTSynDefensiveProgramming	1..1	**Label:** Defensive Programming Options Parameters for defensive programming
EthTSynGeneral	1..1	This container contains the general configuration parameters of the EthTSyn module.
EthTSynGlobalTimeDomain	1..n	This represents the existence of a global time domain on Ethernet. The EthTSyn module can administrate several global time domains at the same time that in itself form a hierarchy of domains and sub-domains.
EthTSynReportError	1..1	**Label:** Production error handling Production error handling
PublishedInformation	1..1	**Label:** EB Published Information Additional published parameters not covered by CommonPublishedInformation container.

Parameters included	
Parameter name	**Multiplicity**

Parameters included	
IMPLEMENTATION_CONFIG_VARIANT	1..1

Parameter Name	IMPLEMENTATION_CONFIG_VARIANT	
Label	Config Variant	
Description	The module supports different post-build variants (previously known as post-build selectable configuration sets), but not post-build loadable configuration.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	VariantPostBuild	
Range	VariantPostBuild	
Configuration class	**VariantPostBuild:**	VariantPostBuild

5.3.1.1. CommonPublishedInformation

Parameters included	
Parameter name	**Multiplicity**
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	4

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	6
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	3
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	0
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	164
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

5.3.1.2. EthTSynDefensiveProgramming

Parameters included	
Parameter name	**Multiplicity**
EthTSynDefProgEnabled	1..1
EthTSynPrecondAssertEnabled	1..1
EthTSynPostcondAssertEnabled	1..1
EthTSynStaticAssertEnabled	1..1
EthTSynUnreachAssertEnabled	1..1
EthTSynInvariantAssertEnabled	1..1

Parameter Name	**EthTSynDefProgEnabled**
Label	Enable Defensive Programming
Description	Enables or disables the defensive programming feature for the module EthTSyn. Note: This feature is dependent on the use of the development error detection module. To use the defensive programming feature, proceed as follows: 1. Enable development error detection 2. Enable defensive programming 3. Enable assertions as required
Multiplicity	1..1
Type	BOOLEAN
Default value	false

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

Parameter Name	**EthTSynPrecondAssertEnabled**
Label	Enable Precondition Assertions
Description	Enables handling of precondition assertion checks reported from the module EthTSyn. Dependency on parameter(s): ► Enable Development Error Detection (`EthTSynDevErrorDetect`): must be enabled ► Enable Defensive Programming (`EthTSynDefProgEnabled`): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	**EthTSynPostcondAssertEnabled**
Label	Enable Postcondition Assertions
Description	Enables handling of postcondition assertion checks reported from the module EthTSyn. Dependency on parameter(s): ► Enable Development Error Detection (`EthTSynDevErrorDetect`): must be enabled ► Enable Defensive Programming (`EthTSynDefProgEnabled`): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	**EthTSynStaticAssertEnabled**

Label	Enable Static Assertions	
Description	Enables handling of static assertion checks reported from the module EthTSyn. Dependency on parameter(s): ► Enable Development Error Detection (`EthTSynDevErrorDetect`): must be enabled ► Enable Defensive Programming (`EthTSynDefProgEnabled`): must be enabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynUnreachAssertEnabled**	
Label	Enable Unreachable Code Assertions	
Description	Enables handling of unreachable code assertion checks reported from the module EthTSyn. Dependency on parameter(s): ► Enable Development Error Detection (`EthTSynDevErrorDetect`): must be enabled ► Enable Defensive Programming (`EthTSynDefProgEnabled`): must be enabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynInvariantAssertEnabled**
Label	Enable Invariant Assertions
Description	Enables handling of invariant assertion checks reported from functions of the module EthTSyn. Dependency on parameter(s):

	▶ Enable Development Error Detection (`EthTSynDevErrorDetect`): must be enabled ▶ Enable Defensive Programming (`EthTSynDefProgEnabled`): must be enabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.3. EthTSynGeneral

Parameters included	
Parameter name	**Multiplicity**
EthTSynDestPhyAddr	1..1
EthTSynDevErrorDetect	1..1
EthTSynGlobalTimeRxToUplinkSwitchResidenceTime	0..1
EthTSynGlobalTimeUplinkToTxSwitchResidenceTime	0..1
EthTSynHardwareTimestampSupport	1..1
EthTSynMainFunctionPeriod	1..1
EthTSynMasterSlaveConflictDetection	1..1
EthTSynMessageCompliance	1..1
EthTSynSwitchMgmtRxMessageBufferCount	0..1
EthTSynVersionInfoApi	1..1
EthTSynTimeValidationSupport	1..1
EthTSynAnnounceFrameSupport	1..1
EthTSynSendSyncFrameOnlyOnHostPort	1..1
EthTSynReliableTxConfirmationSupport	1..1
EthTSynEnableDeviceAuthenticationSupport	1..1
EthTSynEthIfFrameType	1..1
EthTSynVariantHandlingIsUsed	1..1
EthTSyNumberOfConfiguredSlaves	1..1
EthTSynMaxNumberOfSwtPorts	1..1

Parameter Name	EthTSynDestPhyAddr	
Description	Destination Physical Address (MAC-Address). With this parameter it is possible to overwrite the default IEEE 802.1AS MAC address 01:80:C2:00:00:0E by an individual address. Use the format FF:FF:FF:FF:FF:FF . Caution: By using another MAC address the IEEE 802.1AS is left which may cause a decrease of accuracy.	
Multiplicity	1..1	
Type	STRING	
Default value	01:80:C2:00:00:0E	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynDevErrorDetect	
Description	Switches the development error detection and notification on or off.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynGlobalTimeRxToUplinkSwitchResidenceTime	
Description	This parameter is specifying the default value used for the residence time of the Ethernet Switch [Ingress to Uplink]. **Note: This configuration parameter is not used.**	
Multiplicity	0..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynGlobalTimeUplinkToTxSwitchResidenceTime
Description	This parameter is specifying the default value used for the residence time of the Ethernet Switch [Uplink to Egress].

	Note: This configuration parameter is not used.	
Multiplicity	0..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynHardwareTimestampSupport**	
Description	Activate/Deactivate the hardware time stamping functionality of the Ethernet hardware. **True:** Time stamp is retrieved from the Ethernet hardware. **False:** Time stamp is retrieved from the StbM.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynMainFunctionPeriod**	
Description	Schedule period of the main function EthTSyn_MainFunction.	
Multiplicity	1..1	
Type	FLOAT	
Default value	1.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynMasterSlaveConflictDetection**	
Description	Enables master / slave conflict detection and notification.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynMessageCompliance	
Description	* true: IEEE 802.1AS compliant message format will be used.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynSwitchMgmtRxMessageBufferCount	
Description	This parameter is used to determine the amount of Rx message buffers available in the EthTSyn when EthTSyn is used in a Bridge configuration. **Note: This configuration parameter is not used**.	
Multiplicity	0..1	
Type	INTEGER	
Default value	10	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynVersionInfoApi	
Description	Activate/Deactivate the version information API (EthTSyn_GetVersionInfo). True: version information API activated False: version information API deactivated.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	TRUE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynTimeValidationSupport	
Description	Enables/Disables the usage of the recording functionality for Synchronized time-bases for Global Time precision measurement purpose.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild

Origin	Elektrobit Automotive GmbH

Parameter Name	**EthTSynAnnounceFrameSupport**	
Description	Switch for enabling the transmission of IEEE 802.1AS-2011 announce frames. If enabled, a EthTSyn master periodically sends announce frames in the same interval as defined for sync frames. **Note: This parameter has no effect for configured EthTSyn slaves.**	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynSendSyncFrameOnlyOnHostPort**	
Description	If this parameter is set, EthTSyn will send only one sync frame on host port and EthSwt will handle the transmission on the other ports. This parameter makes sense only if Switch delay compensation feature is enabled.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynReliableTxConfirmationSupport**
Description	If this parameter is set, EthTSyn will support EthTSyn_TxConfirmation() according to AUTOSAR R20-11. Additional parameter Result was added, which will provide the status of the transmission: E_OK: The transmission was successful E_NOT_OK: The transmission failed. EthIf must support reliable TxConfirmation(AR-58249).
Multiplicity	1..1
Type	BOOLEAN
Default value	false

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

Parameter Name	**EthTSynEnableDeviceAuthenticationSupport**	
Description	Enables support of Authentication Challenge TLV for Device Authentication. DevAuth module supports only the supplicant side. The supplicant must be an EthTSyn slave.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynEthIfFrameType**	
Description	The chosen frame owner determines which frames (in respect to ethertype) are received. **Note: This configuration parameter is not used.**	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynVariantHandlingIsUsed**	
Description	If multiple configuration variants are used this parameter must be set to true in order to activate EthTSyNumberOfConfiguredSlaves and EthTSynMaxNumberOfSwtPorts parameters.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSyNumberOfConfiguredSlaves**

Description	Number of EthTSyn ports with the Role of the EthTSyn-Port as Slave. This parameter is enabled if Variant handlings is used (EthTSynVariantHandlingIsUsed = TRUE). Example on how to configure this parameter: Consider that 3 variants are used: - One variant contains 2 ports as slave - One variant contains 5 ports as slave - One variant contains 4 ports as slave EthTSyNumberOfConfiguredSlaves must be set to 5 (the greatest value of all 3 variants)	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynMaxNumberOfSwtPorts**
Description	Maximum number of EthTSyn ports which contains a valid reference to an EthSwt port. This parameter is enabled if Variant handling is used (EthTSynVariantHandlingIsUsed = TRUE). Relevant only if Time measurement with Switches is used. Example on how to configure this parameter: Consider that 3 variants are used: - One variant contains 4 ports (one host port and 3 non-host ports) - One variant contains 7 ports (one host port and 6 non-host ports) - One variant contains 5 ports (one host port and 4 non-host ports) EthTSynMaxNumberOfSwtPorts must be set to 7 (the greatest value of all 3 variants)
Multiplicity	1..1
Type	INTEGER

Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.4. EthTSynGlobalTimeDomain

Containers included		
Container name	**Multiplicity**	**Description**
EthTSynPortRole	1..1	Specifying the Role of the EthTSyn-Port (Master or Slave).
EthTSynGlobalTimeFollowUpDataIDList	0..1	The DataIDList for Follow_Up message ensures the identification of data elements due to CRC calculation process.
EthTSynPortConfig	1..n	Configuration of the EthTSyn-Ports within the TimeDomain.

Parameters included	
Parameter name	**Multiplicity**
EthTSynGlobalTimeDomainId	1..1
EthTSynGlobalTimeSecureTmacLength	1..1
EthTSynSynchronizedTimeBaseRef	1..1
EthTSynFramePrio	0..1
EthTSynGlobalTimeDebounceTime	1..1
EthTSynGlobalTimeEthIfRef	1..1
EthTSynSwitchManagementEthSwitchPortHostRef	0..1

Parameter Name	**EthTSynGlobalTimeDomainId**	
Description	The global time domain ID. Synchronized time bases: 0 and 15 - EthTSynSynchronizedTimeBaseRef must refer a SyncTimeBase Offset time bases: 16 and 31 - EthTSynSynchronizedTimeBaseRef must refer an OffsetTimeBase If configuration parameter EthTSynMessageCompliance is TRUE only one TimeDomain is allowed and the value of EthTSynGlobalTimeDomainId is 0 by default.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild

Origin	AUTOSAR_ECUC

Parameter Name	**EthTSynGlobalTimeSecureTmacLength**	
Description	Represents the number of bytes for the used Truncated Message Authentication Code (TMAC). If 0, no message authentication will be used.Tags: atp.Status=draft	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=16 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynSynchronizedTimeBaseRef**	
Description	Mandatory reference to the required synchronized time-base.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynFramePrio**	
Description	This optional parameter, if present, indicates the priority of outgoing EthTSyn messages, if sent via VLAN (used for the 3-bit PCP field of the VLAN tag). If this optional parameter is not present, frames are sent without a priority and VLAN field. **Note: If this optional parameter is not present, frames are sent without a priority and VLAN field.**	
Multiplicity	0..1	
Type	INTEGER	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynGlobalTimeDebounceTime**

Description	This represents the configuration of a TX debounce time for all Timesync PDUs of the same time domain and for the same port. Unit: seconds. If value of EthTSynGlobalTimeDebounceTime is 0 it will be sent immediately, else, if greater, EthTSynGlobalTimeDebounceTime will be round up to full main function periods.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynGlobalTimeEthIfRef**	
Description	This represents the reference to the Ethernet interface taken to fetch the global time information.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynSwitchManagementEthSwitchPortHostRef**	
Multiplicity	0..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PreCompile:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.5. EthTSynPortRole

Containers included		
Container name	**Multiplicity**	**Description**
EthTSynGlobalTimeMaster	1..1	Configuration of the global time master. Each global time domain is required to have exactly one global time master. This master may or may not exist on the configured ECU.
EthTSynGlobalTimeSlave	1..1	Configuration of a time slave. Each global time domain is required to have at least one time slave. The configured ECU may or may not represent a time slave.

5.3.1.6. EthTSynGlobalTimeMaster

Containers included		
Container name	**Multiplicity**	**Description**
EthTSynCrcTimeFlagsTxSecured	0..1	This container collects definitions which parts of the Follow_Up message elements shall be used for CRC calculation.
EthTSynDemEventParameterRefs	0..1	**Label:** Dem Events 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.

Parameters included	
Parameter name	**Multiplicity**
EthTSynCyclicMsgResumeTime	1..1
EthTSynGlobalTimeTxCrcSecured	1..1
EthTSynGlobalTimeTxPeriod	1..1
EthTSynImmediateTimeSync	1..1
EthTSynTLVFollowUpOFSSubTLV	1..1
EthTSynTLVFollowUpStatusSubTLV	1..1
EthTSynTLVFollowUpTimeSubTLV	1..1
EthTSynTLVFollowUpUserDataSubTLV	1..1
EthTSynTxTmacCalculated	1..1
EthTSynSwtPortIdxInCorrField	1..1

Parameter Name	**EthTSynCyclicMsgResumeTime**	
Description	Defines the time where the 1st regular cycle time based message transmission takes place, after an immediate transmission before. EthTSynImmediateTimeSync must be set to true to get this parameter enabled. Unit: seconds.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild

Origin	AUTOSAR_ECUC

Parameter Name	**EthTSynGlobalTimeTxCrcSecured**	
Description	This represents the configuration of whether or not CRC is supported.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	CRC_NOT_SUPPORTED	
Range	CRC_NOT_SUPPORTED CRC_SUPPORTED	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynGlobalTimeTxPeriod**	
Description	This represents configuration of the TX period. Unit: seconds.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.125	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynImmediateTimeSync**	
Description	Enables/Disables the cyclic polling of StbM_GetTimeBaseUpdateCounter() within EthTSyn_MainFunction().	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynTLVFollowUpOFSSubTLV**
Description	This represents the configuration of whether an AUTOSAR Follow_Up TLV OFS Sub-TLV is used or not.
Multiplicity	1..1
Type	BOOLEAN

Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynTLVFollowUpStatusSubTLV**	
Description	This represents the configuration of whether an AUTOSAR Follow_Up TLV Status Sub-TLV is used or not.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynTLVFollowUpTimeSubTLV**	
Description	This represents the configuration of whether an AUTOSAR Follow_Up TLV Time Sub-TLV is used or not.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynTLVFollowUpUserDataSubTLV**	
Description	This represents the configuration of whether an AUTOSAR Follow_Up TLV UserData Sub-TLV is used or not.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynTxTmacCalculated**
Description	This parameter controls whether or not TMAC calculation shall be supported.Tags: atp.Status=draft

Multiplicity	1..1	
Type	ENUMERATION	
Range	TMAC_CALCULATED TMAC_NOT_CALCULATED	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynSwtPortIdxInCorrField**	
Description	If this parameter is set, then the feature which allows to save in the fractional ns part of the correction field, the SwitchIdx (in most significant byte) and PortIdx (in least significant byte), is enabled.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.7. EthTSynCrcTimeFlagsTxSecured

Parameters included	
Parameter name	**Multiplicity**
EthTSynCrcCorrectionField	1..1
EthTSynCrcDomainNumber	1..1
EthTSynCrcMessageLength	1..1
EthTSynCrcPreciseOriginTimestamp	1..1
EthTSynCrcSequenceId	1..1
EthTSynCrcSourcePortIdentity	1..1

Parameter Name	**EthTSynCrcCorrectionField**	
Description	The correctionField from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	

Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynCrcDomainNumber**	
Description	The domainNumber from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynCrcMessageLength**	
Description	The messageLength from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynCrcPreciseOriginTimestamp**	
Description	The preciseOriginTimestamp from the Follow_Up Message Field shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynCrcSequenceId**
Description	The sequenceId from the Follow_Up Message Header shall be included in CRC calculation.
Multiplicity	1..1

Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynCrcSourcePortIdentity**	
Description	The sourcePortIdentity from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

5.3.1.8. EthTSynDemEventParameterRefs

Parameters included	
Parameter name	**Multiplicity**
ETHTSYN_E_UNEXPECTED_SYNC	0..1

Parameter Name	**ETHTSYN_E_UNEXPECTED_SYNC**
Label	ETHTSYN_E_UNEXPECTED_SYNC
Description	Reference to the configured DEM event to report the unexpected reception of SYNC frames. Dependency on parameter(s): ► `EthTSynUnexpectedSyncReportError`: Select DEM to enable the reporting of `ETHTSYN_E_UNEXPECTED_SYNC` event. Further notes: ► Activation: Thrown, if a SYNC frame is received by a port configured as EthTSynPortRole to EthTSynGlobalTimeMaster. ► Healing: Will not be healed. ► Trigger debounce: None. The error is reported on first occurrence. ► Rate of diagnostic checks: Checked everytime a SYNC frame is received.

Multiplicity	0..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**PreCompile:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.9. EthTSynGlobalTimeSlave

Containers included		
Container name	**Multiplicity**	**Description**
EthTSynCrcFlagsRxValidated	0..1	This container collects definitions which parts of the Follow_Up message elements shall be included in CRC validation.
EthTSynDemEventParameterRefs	0..1	**Label:** Dem Events 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.

Parameters included	
Parameter name	**Multiplicity**
EthTSynGlobalTimeFollowUpTimeout	1..1
EthTSynRxCrcValidated	1..1
EthTSynRxTmacValidated	1..1
EthTSynGlobalTimeSequenceCounterJumpWidth	1..1
EthTSynPdelayFilter	1..1
EthTSynReceivePdelayRespFupPairsPeriod	1..1
EthTSynReceiveSyncFupPairsPeriod	1..1
EthTSynSimpleBridge	1..1
EthTSynSynchronizeSimpleBridge	1..1
EthTSynSimpleBridgeValidateSync	1..1
EthTSynSimpleBridgeTSynSendSync	1..1
EthTSynBridgeTxPeriod	1..1

Parameter Name	EthTSynGlobalTimeFollowUpTimeout	
Description	Timeout value of the Follow_Up message (of the subsequent Sync message). Unit: seconds. **Note: This parameter is only relevant for configured EthTSyn slaves of this time domain.** A value of 0 deactivates this timeout observation.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynRxCrcValidated	
Description	Definition of whether or not validation of the CRC is supported.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	CRC_IGNORED	
Range	CRC_IGNORED CRC_NOT_VALIDATED CRC_OPTIONAL CRC_VALIDATED	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynRxTmacValidated	
Description	This parameter controls whether or not TMAC validation shall be supported.Tags: atp.Status=draft	
Multiplicity	1..1	
Type	ENUMERATION	
Range	TMAC_NOT_VALIDATED TMAC_VALIDATED	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynGlobalTimeSequenceCounterJumpWidth	
Description	The SequenceCounterJumpWidth specifies the maximum allowed jump of the Sequence Counter between two consecutive Sync messages..	
Multiplicity	1..1	
Type	INTEGER	
Default value	65535	
Range	<=65535 >=1	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	EthTSynPdelayFilter	
Description	This parameter allows to filter the measured peer delays. Filtering measured values will reduce big impacts of wrong values and makes the peer delay steady. The filtered peer delay is calculated based on the formular below. The filter performs shift operations to reduce the impact of the peer delay delta between old and new value. Pdelay = PdelayOld + [(PdelayOld - PdelayNew) / 2^EthTSynPdelayFilter] **Note: Setting this parameter to zero will disable any filtering. Setting the filter too high will filter away all deltas causing peer delay to be zero.**	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	EthTSynReceivePdelayRespFupPairsPeriod
Description	This parameter contains the timeout of an unrecieved PdelayResp/Fup frame/pair. If a timeout occurs and EthTSynReportError is set to DEM, Dem_ReportErrorStatus() shall be called with the configured DemEventId taken from ETHTSYN_E_PDELAY_FAILED reference and error status DEM_EVENT_STATUS_PREFAILED. If a timeout occurs and EthTSynReportError is set to DET, Det_ReportError() shall be called with ErrorId set to configured value of EthTSynPdelay-

	FailedReportError. **Note: This parameter is only relevant if EthTSynPdelay-FailedReportError is not equal with DISABLE.**	
Multiplicity	1..1	
Type	FLOAT	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynReceiveSyncFupPairsPeriod**	
Description	This parameter contains the timeout of an unrecieved Sync/Fup pair. If a timeout occurs and EthTSynReportError is set to DEM, Dem_ReportErrorStatus() shall be called with the configured DemEventId taken from ETHTSYN_E_SYNC_FAILED reference and error status DEM_EVENT_STATUS_PREFAILED. If a timeout occurs and EthTSynReportError is set to DET, Det_ReportError() shall be called with ErrorId set to configured value of EthTSynSyncFailedReportErrorDetErrorId. Note: This parameter is closely related to the Sync period of the master (EthTSynGlobalTimeTxPeriod) and the propagation delay of the frames, must be bigger then EthTSynGlobalTimeTxPeriod + Pdelay. **Note: This parameter is only relevant if EthTSynSyncFailedReportError is not equal with DISABLE.**	
Multiplicity	1..1	
Type	FLOAT	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynSimpleBridge**
Description	Editable only if the EthTSynTimeDomain is configured with a Time Aware Bridge(PortRole = Slave, Ports configured > 1). If TRUE, the Time Aware Bridge will act as a transparent clock, and will foward the received Sync messages to the connected slaves. The Sync frame will be unchanged. In the Fup frame the correction field will be updated with the residence time. If the Fup frame contains also a timesecured TLV, also the CRC_Time_1 will be re-calculated. If FALSE, the Time Aware Bridge will act as a boundary clock. This means that the timebase will, firstly, be synchronized with the time received from the GTM.

	After synchronization is done, the EthTSyn will send periodically, new sync and fup pairs to the slaves.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	**EthTSynSynchronizeSimpleBridge**
Description	Editable only if the EthTSynSimpleBridge is enabled and TRUE. If TRUE, the Time Aware Bridge will act as a transparent clock, but will be capable to synchronize the referenced time base. If FALSE, the Time Aware Bridge acting as a transparent clock will not synchronize the referenced time base.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	**EthTSynSimpleBridgeValidateSync**
Description	Editable only if the EthTSynSimpleBridge is enabled and TRUE. This parameter activates/deactivates the checks Autosar does on the SYNC. This also means that if the checks fail, the SYNC will be dropped already at Bridge level. This gives flexibility in configuring, as the IEEE1588 does not require SYNC validations on Transparent clock, and the Simple Bridge is based on a Transparent Clock from IEEE1588.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	**EthTSynSimpleBridgeTSynSendSync**

Description	Editable only if the EthTSynSimpleBridge is enabled and TRUE. If this parameter is set to TRUE, EthTSyn will handle the fowarding of the sync frames to the connected ports. If this parameter is set to FALSE, EthSwt will handle the fowarding of the sync frame to the connected ports.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynBridgeTxPeriod**	
Description	This represents configuration of the TX period. Unit: seconds. This period is used by the boundary clock, to transmit the synchronization frames to the connected slaves based on the formula provided by the standard IEEE 802.1AS-2011: (rcvdPSSync && (currentTime ? lastSyncSentTime >= 0.-5*syncInterval(mainfunction cycles)) OR (currentTime ? lastSyncSentTime >= syncInterval) Enabled only if: Time Aware Bridge without GTM as Management CPU is configured and acts like a boundary clock. (PorRole = Slave, switch references are valid and EthTSynSimpleBridge = FALSE).	
Multiplicity	1..1	
Type	FLOAT	
Default value	0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.10. EthTSynCrcFlagsRxValidated

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

Parameters included	
EthTSynCrcDomainNumber	1..1
EthTSynCrcMessageLength	1..1
EthTSynCrcPreciseOriginTimestamp	1..1
EthTSynCrcSequenceId	1..1
EthTSynCrcSourcePortIdentity	1..1

Parameter Name	EthTSynCrcCorrectionField	
Description	The correctionField from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynCrcDomainNumber	
Description	The domainNumber from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynCrcMessageLength	
Description	The messageLength from the Follow_Up Message Header shall be included in CRC calculation.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	EthTSynCrcPreciseOriginTimestamp

Description	The preciseOriginTimestamp from the Follow_Up Message Field shall be included in CRC calculation.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EthTSynCrcSequenceId**
Description	The sequenceId from the Follow_Up Message Header shall be included in CRC calculation.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EthTSynCrcSourcePortIdentity**
Description	The sourcePortIdentity from the Follow_Up Message Header shall be included in CRC calculation.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

5.3.1.11. EthTSynDemEventParameterRefs

Parameters included	
Parameter name	**Multiplicity**
ETHTSYN_E_PDELAY_FAILED	0..1
ETHTSYN_E_SYNC_FAILED	0..1

Parameter Name	**ETHTSYN_E_PDELAY_FAILED**

Label	ETHTSYN_E_PDELAY_FAILED
Description	Reference to the configured DEM event to report a time-out supervision error for Pdelay_Resp-/Pdelay_Resp_Follow_Up frames for this port. Dependency on parameter(s): ► `EthTSynPdelayFailedReportError`: Select DEM to enable the reporting of `ETHTSYN_E_PDELAY_FAILED` event. Further notes: ► Activation: DEM_EVENT_STATUS_PREFAILED status is reported to the Dem module, every time a Pdelay_Resp-/Pdelay_Resp_Follow_Up error occurs. ► Debounce is done by the Dem module by configuring the DemDebounceCounterFailedThreshold of the referenced event. ► Default trigger debounce: 6 subsequent failures of Pdelay_Resp-/Pdelay_Resp_Follow_Up message pairs. ► Healing: Will be healed, if a correct Pdelay_Resp-/Pdelay_Resp_Follow_Up pair is received. ► Rate of diagnostic checks: Checked everytime a Pdelay_Resp-/Pdelay_Resp_Follow_Up pair is received.
Multiplicity	0..1
Type	SYMBOLIC-NAME-REFERENCE
Configuration class	**PreCompile:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	**ETHTSYN_E_SYNC_FAILED**
Label	ETHTSYN_E_SYNC_FAILED
Description	Reference to the configured DEM event to report a time-out supervision error for SYNC/FUP frames for this port. Dependency on parameter(s): ► `EthTSynSyncFailedReportError`: Select DEM to enable the reporting of `ETHTSYN_E_SYNC_FAILED` event. Further notes: ► Activation: DEM_EVENT_STATUS_PREFAILED status is reported to the Dem module, every time a SYNC/FUP error occurs.

	▶ Debounce is done by the Dem module by configuring the DemDebounceCounterFailedThreshold of the referenced event. ▶ Default trigger debounce: 12 subsequent failures of SYNC/FUP message pairs. ▶ Healing: Will be healed, if a correct SYNC/FUP pair is received. ▶ Rate of diagnostic checks: Checked everytime a SYNC/FUP pair is received.
Multiplicity	0..1
Type	SYMBOLIC-NAME-REFERENCE
Configuration class	**PreCompile:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

5.3.1.12. EthTSynGlobalTimeFollowUpDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
EthTSynGlobalTimeFollowUpDataIDListElement	16..16	Element of the DataIDList for Follow_Up message ensures the identification of data elements due to CRC calculation process.

5.3.1.13. EthTSynGlobalTimeFollowUpDataIDListElement

Parameters included	
Parameter name	**Multiplicity**
EthTSynGlobalTimeFollowUpDataIDListIndex	1..1
EthTSynGlobalTimeFollowUpDataIDListValue	1..1

Parameter Name	**EthTSynGlobalTimeFollowUpDataIDListIndex**
Description	Index of the DataIDList for Follow_Up message ensures the identification of data elements due to CRC calculation process.
Multiplicity	1..1
Type	INTEGER
Range	<=15 >=0

Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynGlobalTimeFollowUpDataIDListValue**	
Description	Value of the DataIDList for Follow_Up message ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=255 >=0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

5.3.1.14. EthTSynPortConfig

Containers included		
Container name	**Multiplicity**	**Description**
EthTSynPdelayConfig	1..1	Configuration of cyclic propagation delay measurement.

Parameters included	
Parameter name	**Multiplicity**
EthTSynGlobalTimeMinMsgGap	1..1
EthTSynSwitchManagementEthSwitchPortRef	0..1
EthTSynEnableTimeValidation	1..1
EthTSynPortNumber	1..1

Parameter Name	**EthTSynGlobalTimeMinMsgGap**
Description	This parameter represents the configuration of a minimum message gap time for received Timesync messages compared to a message before with the same PDU. If PDUs are received more often in between than this parameter allows, they shall be ignored.Tags: atp.Status=draft
Multiplicity	1..1
Type	FLOAT
Default value	0.0

Range	<Infinity	
	>=0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynSwitchManagementEthSwitchPortRef**	
Description	In an AVB-Bridge config, this reference is used to assign the EthTSyn-Port to an Ethernet Switch-Port.	
Multiplicity	0..1	
Type	REFERENCE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynEnableTimeValidation**	
Description	Enables/disables time recording for time validation for a specific Time Domain. To be able to use this feature EthTSynTimeValidationSupport must be enabled. True: time validation is enabled. False: time validation is disabled.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynPortNumber**
Description	The portNumber values for the PTP Ports on a time-aware system shall be distinct in the range 1, 2,... 0xFFFE. The portNumber value 0 is assigned to the interface between the ClockMaster and ClockSource entities. The value 0xFFFF is reserved. Enabled only if: The configuration parameter EthTSynSendSyncFrameOnlyOnHostPort is FALSE.

	AND Time Aware Bridge with GTM as Management CPU is configured (PorRole = Master and switch references are valid). OR Time Aware Bridge without GTM as Management CPU is configured and acts like a boundary clock. (PorRole = Slave, switch references are valid and EthTSynSimpleBridge = FALSE).
Multiplicity	1..1
Type	INTEGER
Default value	1
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	Elektrobit Automotive GmbH

5.3.1.15. EthTSynPdelayConfig

Parameters included	
Parameter name	**Multiplicity**
EthTSynGlobalTimePdelayRespEnable	1..1
EthTSynGlobalTimePropagationDelay	1..1
EthTSynGlobalTimeTxPdelayReqPeriod	1..1
EthTSynPdelayLatencyThreshold	0..1
EthTSynPdelayRespAndRespFollowUpTimeout	1..1

Parameter Name	**EthTSynGlobalTimePdelayRespEnable**
Description	This parameter allows disabling Pdelay_Resp / Pdelay_Resp_Follow_Up transmission, if no Pdelay_Req messages are expected.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	**VariantPostBuild:** VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	**EthTSynGlobalTimePropagationDelay**

Description	If cyclic propagation delay measurement is enabled, this parameter represents the default value of the propagation delay until the first actually measured propagation delay is available. If cyclic propagation delay measurement is disable, this parameter represents a static propagation delay.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynGlobalTimeTxPdelayReqPeriod**	
Description	This represents configuration of the TX period for Pdelay_Req messages. Unit: seconds.	
Multiplicity	1..1	
Type	FLOAT	
Default value	1.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynPdelayLatencyThreshold**	
Description	Threshold for calculated Pdelay. If a measured Pdelay exceeds EthTSynPdelayLatencyThreshold, this value is discarded.	
Multiplicity	0..1	
Type	FLOAT	
Default value	1.0E-5	
Configuration class	**PreCompile:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	**EthTSynPdelayRespAndRespFollowUpTimeout**	
Description	Timeout value for Pdelay_Resp and Pdelay_Resp_Follow_Up after a Pdelay_Req has been transmitted resp. a Pdelay_Resp has been received.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.0	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	AUTOSAR_ECUC	

Origin	AUTOSAR_ECUC

5.3.1.16. EthTSynReportError

Parameters included	
Parameter name	**Multiplicity**
EthTSynSyncFailedReportError	1..1
EthTSynSyncFailedReportErrorDetErrorId	1..1
EthTSynPdelayFailedReportError	1..1
EthTSynPdelayFailedReportErrorDetErrorId	1..1
EthTSynUnexpectedSyncReportError	1..1
EthTSynUnexpectedSyncReportErrorDetErrorId	1..1

Parameter Name	**EthTSynSyncFailedReportError**
Label	Timeout supervision for SYNC/FUP Frames error
Description	Selects the handling of the production error: *Timeout supervision for SYNC/FUP Frames error* ► `DEM`: All errors are reported to the Diagnostics Event Manager (Dem). ► `DET`: All errors are reported to the Development Error Tracer (Det) if enabled. Error passed to DET module if more than 12 Sync-/Follow_Up messages in a row are dropped on a port configured as EthTSynGlobalTimeSlave. ► `DISABLE`: Production errors are not reported at all. **Optimization Effect:** ► **ROM reduction (code):** Setting this parameter to a value of DISABLE reduces the ROM consumption of the module code. ► **Execution time reduction (code):** Setting this parameter to a value of DISABLE reduces the execution time of the module code.
Multiplicity	1..1
Type	ENUMERATION
Default value	DISABLE
Range	DEM DET

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

Parameter Name	**EthTSynSyncFailedReportErrorDetErrorId**	
Label	Timeout supervision for SYNC/FUP Frames error Det error ID	
Description	If a production error is reported towards the Det, this parameter defines the error id of the production errors ETHTSYN_E_SYNC_FAILED for all controllers. The Det instance id is the EthIf controller ID (parameter EthIfCtrlIdx of the EthIf controller referenced by parameter EthTSynGlobalTimeEthIfRef).	
Multiplicity	1..1	
Type	INTEGER	
Default value	255	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynPdelayFailedReportError**
Label	Timeout supervision for Pdelay_Resp-/Pdelay_Resp_Follow_Up Frames error
Description	Selects the handling of the production error: *Timeout supervision for Pdelay_Resp-/Pdelay_Resp_Follow_Up error* ► `DEM`: All errors are reported to the Diagnostics Event Manager (Dem). ► `DET`: All errors are reported to the Development Error Tracer (Det) if enabled. Error passed to DET module if more than 6 Pdelay_Resp-/Pdelay_Resp_Follow_Up messages in a row are dropped on a port configured as EthTSynGlobalTimeSlave. ► `DISABLE`: Production errors are not reported at all. **Optimization Effect:** ► **ROM reduction (code):** Setting this parameter to a value of DISABLE reduces the ROM consumption of the module code. ► **Execution time reduction (code):** Setting this parameter to a value of DISABLE reduces the execution time of the module code.
Multiplicity	1..1
Type	ENUMERATION
Default value	DISABLE

Range	DEM	
	DET	
	DISABLE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynPdelayFailedReportErrorDetErrorId**	
Label	Timeout supervision for Pdelay_Resp-/Pdelay_Resp_Follow_Up Frame error Det error ID	
Description	If a production error is reported towards the Det, this parameter defines the error id of the production errors ETHTSYN_E_PDELAY_FAILED for all controllers. The Det instance id is the EthIf controller ID (parameter EthIfCtrlIdx of the EthIf controller referenced by parameter EthTSynGlobalTimeEthIfRef).	
Multiplicity	1..1	
Type	INTEGER	
Default value	255	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynUnexpectedSyncReportError**
Label	Unexpected reception of SYNC frames error
Description	Selects the handling of the production error: *Unexpected reception of SYNC frames error* ► `DEM`: All errors are reported to the Diagnostics Event Manager (Dem). ► `DET`: All errors are reported to the Development Error Tracer (Det) if enabled. ► `DISABLE`: Production errors are not reported at all. **Optimization Effect:** ► **ROM reduction (code):** Setting this parameter to a value of DISABLE reduces the ROM consumption of the module code. ► **Execution time reduction (code):** Setting this parameter to a value of DISABLE reduces the execution time of the module code.
Multiplicity	1..1

Type	ENUMERATION	
Default value	DISABLE	
Range	DEM DET DISABLE	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	**EthTSynUnexpectedSyncReportErrorDetErrorId**	
Label	Unexpected reception of SYNC frames error Det error ID	
Description	If a production error is reported towards the Det, this parameter defines the error id of the production errors ETHTSYN_E_UNEXPECTED_SYNC for all controllers. The Det instance id is the EthIf controller ID (parameter EthIfCtrlIdx of the EthIf controller referenced by parameter EthTSynGlobalTimeEthIfRef). If EthTSynMasterSlaveConflictDetection is enabled, in case an EthTSyn master receives a sync frame, two DETs shall be reported. One having the static ErrorId specified in AUTOSAR, one the configured EthTSynUnexpectedSyncReportErrorDetErrorId.	
Multiplicity	1..1	
Type	INTEGER	
Default value	255	
Configuration class	**VariantPostBuild:**	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

5.3.1.17. PublishedInformation

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

Parameter Name	**PbcfgMSupport**
Label	PbcfgM support

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

5.3.2. Application programming interface (API)

5.3.2.1. Type definitions

5.3.2.1.1. EthTSyn_PtrUint8

Purpose	The data pointer used to get rid of MISRA violation.
Type	`uint8 *`

5.3.2.2. Macro constants

5.3.2.2.1. ETHTSYN_E_CTRL_IDX

Purpose	
Value	0x22U
Description	DET error code: Invalid controller index

5.3.2.2.2. ETHTSYN_E_INIT_FAILED

Purpose	
Value	0x21U
Description	DET error code: EthTSyn initialization failed

5.3.2.2.3. ETHTSYN_E_PARAM

Purpose	
Value	0x24U
Description	DET error code: Invalid parameter

5.3.2.2.4. ETHTSYN_E_PARAM_POINTER

Purpose	
Value	0x23U
Description	DET error code: Invalid pointer (NULL_PTR)

5.3.2.2.5. ETHTSYN_E_UNINIT

Purpose	
Value	0x20U
Description	DET error code: Module not initialized

5.3.2.2.6. ETHTSYN_INSTANCE_ID

Purpose	Module instance ID.
Value	0U
Description	Defines the instance number of this module. Since multiple instances are not supported this ID is always zero.

5.3.2.2.7. ETHTSYN_INTERNAL_SVCID

Purpose	Defines API id of internal functions.
Value	0xFFU

5.3.2.2.8. ETHTSYN_SID_ETHSWTPTPRXINDICATION

Purpose	Defines API id of function EthTSyn_EthSwtPtpRxIndication().
Value	0xE1U

5.3.2.2.9. ETHTSYN_SID_ETHSWTPTPTIMESTAMPINDICATION

Purpose	Defines API id of function EthTSyn_EthSwtPtpTimeStampIndication().
Value	0xE0U

5.3.2.2.10. ETHTSYN_SID_GETPROTOCOLPARAM

Purpose	Defines API id of function EthTSyn_GetProtocolParam().
Value	0xBU

5.3.2.2.11. ETHTSYN_SID_GETVERSIONINFO

Purpose	Defines API id of function EthTSyn_GetVersionInfo().
Value	0x02U

5.3.2.2.12. ETHTSYN_SID_INIT

Purpose	Defines API id of function EthTSyn_Init().
Value	0x01U

5.3.2.2.13. ETHTSYN_SID_MAINFUNCTION

Purpose	Defines API id of function EthTSyn_MainFunction().
Value	0x09U

5.3.2.2.14. ETHTSYN_SID_RXINDICATION

Purpose	Defines API id of function EthTSyn_RxIndication().
Value	0x06U

5.3.2.2.15. ETHTSYN_SID_SETPROTOCOLPARAM

Purpose	Defines API id of function EthTSyn_SetProtocolParam().

Value	0xAU

5.3.2.2.16. ETHTSYN_SID_SETTRANSMISSIONMODE

Purpose	Defines API id of function EthTSyn_SetTransmissionMode().
Value	0x05U

5.3.2.2.17. ETHTSYN_SID_TRCVLINKSTATECHG

Purpose	Defines API id of function EthTSyn_TrcvLinkStateChg().
Value	0x08U

5.3.2.2.18. ETHTSYN_SID_TXCONFIRMATION

Purpose	Defines API id of function EthTSyn_TxConfirmation().
Value	0x07U

5.3.2.3. Functions

5.3.2.3.1. EthTSyn_EthSwtPtpRxIndication

Purpose	By this API service the EthTSyn gets an indication and the data of a received frame from EthSwt.	
Synopsis	`boolean` **`EthTSyn_EthSwtPtpRxIndication`** `( const EthSwt_MgmtInfoType * MgmtInfoPtr , Eth_TimeStampType * IngressTimeStampPtr , Eth_DataType * DataPtr , uint8 EthIfCtrlIdx , uint16 Len , uint32 UniqueId );`	
Parameters (in)	`MgmtInfoPtr`	- Pointer that holds the switch info: SwitchIdx and SwitchPortIdx
	`IngressTimeStampPtr`	- Pointer that holds the IngressTimeStamp of the received frame.
	`DataPtr`	- Pointer to payload of the received Ethernet frame

	`EthIfCtrlIdx`	- Index of the controller within the context of the EthIf.
	`Len`	- Length of received data.
	`UniqueId`	- Unique Id provided by EthSwt used to identify the right TimeStamps for certain frames.
Return Value	boolean - True - if received frame was processed succefully and a new TimeStamp is required. ► False - frame processing failed and no new TimeStamp is required.	

5.3.2.3.2. EthTSyn_EthSwtPtpTimeStampIndication

Purpose	API used to retrieve the timestamps from EthSwt.	
Synopsis	`void` **`EthTSyn_EthSwtPtpTimeStampIndication`** `( uint8 EthIfCtrlIdx , EthSwt_MgmtInfoType MgmtInfo , uint32 UniqueId , Eth_TimeStampType * PortTimeStampPtr );`	
Parameters (in)	`EthIfCtrlIdx`	- Index of the controller within the context of the EthIf.
	`MgmtInfo`	- Pointer that holds the switch info: SwitchIdx and SwitchPortIdx
	`UniqueId`	- Unique Id provided by EthSwt used to identify the right TimeStamps for certain frames.
	`PortTimeStampPtr`	- Pointer that holds the TimeStamp provided by EthSwt for a certain port.

5.3.2.3.3. EthTSyn_GetProtocolParam

Purpose	EthTSyn_GetProtocolParam function. This API is used to read FollowUp information TLV parameters from received Follow_Up message.
Synopsis	`Std_ReturnType` **`EthTSyn_GetProtocolParam`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_ProtocolParamType * protocolParam );`
Service ID	0xB
Sync/Async	Synchronous
Reentrancy	Reentrant

Parameters (in)	`timeBaseId`	- ID of the synchronized time base
Parameters (out)	`protocolParam`	- structure to store received Follow_Up information TLV parameters
Return Value	E_OK: successful E_NOT_OK: failed	

5.3.2.3.4. EthTSyn_GetVersionInfo

Purpose	API to get the module version information.	
Synopsis	`void` **`EthTSyn_GetVersionInfo`** `( Std_VersionInfoType * VersionInfo );`	
Service ID	0x02	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (out)	`VersionInfo`	- Pointer to return the module version information.
Description	This service returns the version information of this module.	

5.3.2.3.5. EthTSyn_Init

Purpose	Initializes the EthTSyn module.	
Synopsis	`void` **`EthTSyn_Init`** `( const EthTSyn_ConfigType * CfgPtr );`	
Service ID	0x01	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`CfgPtr`	- Address of the configuration data structure.
Description	This service initializes the EthTSyn module. It shall be the first function of the module to be called.	

5.3.2.3.6. EthTSyn_MainFunction

Purpose	EthTSyn module main function.
Synopsis	`void` **`EthTSyn_MainFunction`** `( void );`

Service ID	0x09
Sync/Async	Synchronous
Reentrancy	Non Reentrant
Production Errors	▶ ETHTSYN_E_SYNC_FAILED: thrown, if more than 12 pairs of Sync-/Follow_Up messages in a row are dropped. ▶ ETHTSYN_E_PDELAY_FAILED: thrown, if more than 6 pairs of Pdelay_Resp/ Pdelay_Resp_Follow_Up messages in a row are dropped.

5.3.2.3.7. EthTSyn_RxIndication

Purpose	Rx-Indication function.	
Synopsis	`void` **`EthTSyn_RxIndication`** `( uint8 CtrlIdx , Eth_FrameType FrameType , boolean IsBroadcast , uint8 * PhysAddrPtr , uint8 * DataPtr , uint16 LenByte );`	
Service ID	0x06	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Production Errors	▶ ETHTSYN_E_SYNC_FAILED: thrown, if more than 12 pairs of Sync-/Follow_Up messages in a row are dropped. ▶ ETHTSYN_E_PDELAY_FAILED: thrown, if more than 6 pairs of Pdelay_Resp/ Pdelay_Resp_Follow_Up messages in a row are dropped. ▶ ETHTSYN_E_UNEXPECTED_SYNC: thrown, if a SYNC frame is received on an Ethernet port configured as master.	
Parameters (in)	`CtrlIdx`	- Index of the controller within the context of the EthIf.
	`FrameType`	- Ethernet Frame type.
	`IsBroadcast`	- Indicates whether Ethernet destination address is a broadcast address or not.
	`PhysAddrPtr`	- Pointer to Physical source address (MAC address in network byte order) of received Ethernet frame.
	`DataPtr`	- Address of the received payload.
	`LenByte`	- Length of the payload contained in the received Rx buffer.
Description	This service is called by EthIf in case a reception is indicated.	

5.3.2.3.8. EthTSyn_SetProtocolParam

Purpose	EthTSyn_SetProtocolParam function. This API is used to set FollowUp information TLV parameters of a Follow_Up message prior transmission.	
Synopsis	`Std_ReturnType` **`EthTSyn_SetProtocolParam`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_ProtocolParamType * protocolParam );`	
Service ID	0xA	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time base
	`protocolParam`	- structure with Follow_Up information TLV parameters
Return Value	E_OK: successful E_NOT_OK: failed	

5.3.2.3.9. EthTSyn_SetTransmissionMode

Purpose	This service enables/disables the transmission of frames for the whole module.	
Synopsis	`void` **`EthTSyn_SetTransmissionMode`** `( uint8 CtrlIdx , EthTSyn_TransmissionModeType Mode );`	
Service ID	0x05	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`CtrlIdx`	- Index of the EthTSyn controller.
	`Mode`	- Mode to indicate if frame shall be transmitted.

5.3.2.3.10. EthTSyn_TrcvLinkStateChg

Purpose	Transceiver link state change indication.
Synopsis	`void` **`EthTSyn_TrcvLinkStateChg`** `( uint8 CtrlIdx , EthTrcv_LinkStateType LinkState );`
Service ID	0x08
Sync/Async	Synchronous

Reentrancy	Non Reentrant	
Parameters (in)	`CtrlIdx`	- Index of the EthIf controller associated with the EthTrcv.
	`LinkState`	- New state the transceiver link has changed to.
Description	This service indicates a change of the transceiver link change.	

5.3.2.3.11. EthTSyn_TxConfirmation

Purpose	Tx-Confirmation callback function.	
Synopsis	`void` **`EthTSyn_TxConfirmation`** `( uint8 CtrlIdx , Eth_BufIdxType BufIdx , Std_ReturnType Result );`	
Service ID	0x07	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`CtrlIdx`	- Index of the controller within the context of the EthIf.
	`BufIdx`	- Index of the transmitted buffer.
	`Result`	- E_OK: The transmission was successful, E_NOT_OK: The transmission failed.
Description	This service is called by EthIf in case a transmission is confirmed.	

5.3.3. Integration notes

5.3.3.1. Exclusive areas

This section describes the exclusive areas used by the `EthTSyn` module.

5.3.3.1.1. SCHM_ETHTSYN_EXCLUSIVE_AREA_0

Protected data structures	The exclusive area protects the shared variable access off the sourcePortIdentity (see IEEE 802.1AS-2011, chapter 10.-

	2.2.1.3). This identity is used as unique identifier to find the corresponding response and response follow up frame of a peer delay request. Furthermore the initialization of all global variables is protected with this exclusive area.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. The locking mechanism can be disabled if it is ensured that: ► `EthTSyn_MainFunction()` does not preempt `EthTSyn_RxIndication()` and vice versa. ► No EthTSyn API function preempts `EthTSyn_Init()`.

5.3.3.1.2. SCHM_ETHTSYN_EXCLUSIVE_AREA_1

Protect precision of Virtual Local Time	In order to provide enhanced precision of the Virtual Local Time the module EthTSyn requires this exclusive area to increase the precision of the Virtual Local Time.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. The locking mechanism can be disabled if the Time Base that is referenced in StbM uses an OsCounter as Hardware reference.

5.3.3.1.3. SCHM_ETHTSYN_EXCLUSIVE_AREA_2

Protect protocolParam	EthTSyn requires this exclusive area in order to protect the protocolParam.

Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. This exclusive area shall be configured to a spin-lock if EthTSyn is used in a multicore environment with cross-core calls.

5.3.3.2. Production errors

ETHTSYN_E_PDELAY_FAILED	▶ EthTSyn_MainFunction ▶ EthTSyn_RxIndication
ETHTSYN_E_SYNC_FAILED	▶ EthTSyn_MainFunction ▶ EthTSyn_RxIndication
ETHTSYN_E_UNEXPECTED_SYNC	▶ EthTSyn_RxIndication

5.3.3.3. Memory mapping

General information about memory mapping is provided in the EB tresos AutoCore Generic documentation. Refer to the section `Memory mapping and compiler abstraction` in the `Integration notes` section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section
CONFIG_DATA_UNSPECIFIED
VAR_INIT_8
VAR_INIT_16
VAR_INIT_UNSPECIFIED
VAR_CLEARED_UNSPECIFIED
CONST_8
CONST_UNSPECIFIED
CODE

5.3.3.4. Integration requirements

WARNING **Integration requirements list is not exhaustive**

The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user's guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

5.3.3.4.1. lim.EthTSyn.EB_INTREQ_EthTSyn_0001

Description	The reinitialization process shall not interrupt other module functions. If reinitialization of the module is required, the call of EthTSyn_Init() shall not interrupt other module functions.
Rationale	The reinitialization process resets all internal variables. Continuing an interrupted module function after reinitialization can lead to undefined module behavior.

5.3.3.4.2. lim.EthTSyn.EB_INTREQ_EthTSyn_0002

Description	EthTSyn_RxIndication() and EthTSyn_TxConfirmation() shall not preempt each other. The module functions EthTSyn_RxIndication() and EthTSyn_TxConfirmation() shall not preempt each other.
Rationale	A preemption of these two function can be easily avoided during integration. This limitation allows to reduce the number of interrupt locks to protect shared variables and therefore increases the module performance.

5.3.3.4.3. lim.EthTSyn.EB_INTREQ_EthTSyn_0003

Description	EthTSyn_TxConfirmation() shall be non reentrant The module functions EthTSyn_TxConfirmation() shall not interrupt itself.
Rationale	This preemption can be easily avoided during integration. This limitation allows to reduce the number of interrupt locks to protect shared variables and therefore increases the module performance.

5.3.3.4.4. lim.EthTSyn.EB_INTREQ_EthTSyn_0004

Description	EthTSyn_MainFunction must not preempt or be preempted by EthIf_MainFunctionRx/Tx(). The integrator must assure that EthIf_MainFunctionRx/Tx() can not preempt

	EthTSyn_MainFunction(). The integrator also must assure that EthTSyn_MainFunction() does not preempt EthIf_MainFunctionRx/Tx().
Rationale	This limitation reduces code size and execution time by eliminating the need for extensive use of exclusive areas.

5.3.3.4.5. lim.EthTSyn.EB_INTREQ_EthTSyn_0005

Description	Invalid value for BufIdx. EthTSyn_TxConfirmation() shall not be called parameter BufIdx set to 0xFFFFFFFFU.
Rationale	0xFFFFFFFFU is used as invalid value internally. If Eth_BufIdxType is below uint32, this limitation is always fulfilled.

5.3.3.4.6. lim.EthTSyn.EB_INTREQ_EthTSyn_0006

Description	VLAN tags in synchronization frames require custom switch behavior to compensate the switch processing delay equal to IEEE 802.1AS time aware switches. This is necessary to reach the same synchronization accuracy.
Rationale	IEEE 802.1AS does not allow the usage of VLAN tags and therefore, time aware switches following this standard can not be used. Not time aware switches block the default IEEE 802.1AS MAC address and decrease synchronization accuracy if another MAC address is used.

5.3.3.4.7. lim.EthTSyn.EB_INTREQ_EthTSyn_0007

Description	If bridge is used over switch and TimeSecured Sub-TLV is received from GTM, the slave port and all master ports where the Sync Follow-Up frame is forwarded need to have the same flags configured in EthTSynCrcFlagsRxValidated as the GTM where the Sync Follow-Up originated.
Rationale	Not using the same configuration for all ports shall result in different CRC, therefore the Sync Follow-Up frame shall be dropped.

5.3.3.4.8. lim.EthTSyn.EB_INTREQ_EthTSyn_0008

Description	If the time synchronization over Ethernet is used, there should be maximum one slave per TimeDomain.

Rationale	Possible scenarios: If a master is configured and no slave -> the master is the GTM for that TimeDomain. If a slave and multiple masters are configured for a TimeDomain -> the ECU is a gateway. The slave is used to receive the time from the GTM. The masters will send this time to all other nodes from that TimeDomain. If just slave is configured the TimeDomain shall synchronize with the received time from GTM.

5.3.3.4.9. lim.EthTSyn.EB_INTREQ_EthTSyn_0009

Description	If the time synchronization over Ethernet is used with multiple configuration variants, EthTSynVariantHandlingIsUsed parameter must be set to TRUE in order to manual update the maximum number of slaves(e.g.EthTSyNumberOfConfiguredSlaves please check the description of EthTSyNumberOfConfiguredSlaves to see how to configure). Also if the Time measurement with Switches feature is used, parameter EthTSynMaxNumberOfSwtPorts must be updated as well (please check the description of EthTSyNumberOfConfiguredSlaves to see how to configure).
Rationale	Is not possible to access values of a different variant. In Tresos documentation, chapter 5.7.7: By intention, it is not possible to programmatically switch between the different variants. It is also not possible to access values of a different variant. The reason behind this is the concept that every code always only works on one consistent variant. For example a code generator shall always only generate code for one, the currently selected variant.

5.3.3.4.10. lim.EthTSyn.EB_INTREQ_EthTSyn_0010

Description	EthTSyn_Init() shall never be called with NULL_PTR.
Rationale	EthTSyn uses PostBuild selectable and therefore a valid configuration needs to be provided via EthTSyn_Init() function.

5.4. FrTSyn

5.4.1. 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.
FrTSynGeneral	1..1	This container holds the general parameters of the Flexray-specific Synchronized Time-base Manager.
FrTSynGlobalTimeDomain	1..254	This represents the existence of a global time domain on Flexray. The FrTSyn module can administrate several global time domains at the same time that in itself form a hierarchy of domains and sub-domains.
PublishedInformation	1..1	**Label:** EB Published Information Additional published parameters not covered by CommonPublishedInformation 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	VariantPreCompile	
Range	VariantPreCompile	
Configuration class	**VariantPreCompile:**	VariantPreCompile

5.4.1.1. CommonPublishedInformation

Parameters included	
Parameter name	**Multiplicity**
ArMajorVersion	1..1
ArMinorVersion	1..1
ArPatchVersion	1..1

Parameters included	
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	4
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	6
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	3
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	0
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	163
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

5.4.1.2. FrTSynGeneral

Parameters included	
Parameter name	**Multiplicity**
FrTSynDevErrorDetect	1..1
FrTSynMainFunctionPeriod	1..1
FrTSynTimeValidationSupport	1..1

Parameters included	
FrTSynVersionInfoApi	1..1

Parameter Name	FrTSynDevErrorDetect	
Description	Switches the Default Error Tracer (Det) detection and notification ON or OFF.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	FrTSynMainFunctionPeriod	
Description	Schedule period of the main function FrTSyn_MainFunction. Unit: [s].	
Multiplicity	1..1	
Type	FLOAT	
Range	<=65535 >0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	FrTSynTimeValidationSupport	
Description	Switches support for Time Validation on or off.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	FrTSynVersionInfoApi
Description	Activate/Deactivate the version information API (FrTSyn_GetVersionInfo). True: version information API activated False: version information API deactivated.
Multiplicity	1..1
Type	BOOLEAN
Default value	false

Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.3. FrTSynGlobalTimeDomain

Containers included		
Container name	**Multiplicity**	**Description**
FrTSynGlobalTimeMaster	0..1	Configuration of the global time master. Each global time domain is required to have exactly one global time master. This master may or may not exist on the configured ECU. Configuration of the global time master. Each global time domain is required to have exactly one global time master. This master may or may not exist on the configured ECU.
FrTSynGlobalTimeSlave	0..1	This represents the time slave for the enclosing global time domain. This represents the time slave for the enclosing global time domain.
FrTSynGlobalTimeOfsDataIDList	0..1	The DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process. The DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.
FrTSynGlobalTimeSyncDataIDList	0..1	The DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process. The DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.

Parameters included	
Parameter name	**Multiplicity**
FrTSynEnableTimeValidation	1..1
FrTSynGlobalTimeDomainId	1..1
FrTSynGlobalTimeNetworkSegmentId	1..1
FrTSynSynchronizedTimeBaseRef	1..1

Parameter Name	**FrTSynEnableTimeValidation**
Description	Enables/disables time recording for Time Validation for a specific Time Domain.
Multiplicity	1..1
Type	BOOLEAN

Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynGlobalTimeDomainId**	
Description	The global time domain ID.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=31 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynGlobalTimeNetworkSegmentId**	
Description	The numerical identifier of the network on system level scope where this Global Time has been communicated on.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=255 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynSynchronizedTimeBaseRef**	
Description	Mandatory reference to the required synchronized time-base.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.4. FrTSynGlobalTimeMaster

Containers included		
Container name	**Multiplicity**	**Description**

Containers included		
FrTSynGlobalTimeMasterPdu	1..1	This container carries all properties required to configure the PDU sent by the global time master for the given global time domain.

Parameters included	
Parameter name	**Multiplicity**
FrTSynGlobalTimeTxPeriod	1..1
FrTSynGlobalTimeTxCrcSecured	1..1
FrTSynCyclicMsgResumeTime	1..1
FrTSynGlobalTimeDebounceTime	1..1
FrTSynImmediateTimeSync	1..1

Parameter Name	FrTSynGlobalTimeTxPeriod	
Description	This represents the TX period. Unit: seconds.	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	FrTSynGlobalTimeTxCrcSecured	
Description	This represents the configuration of whether or not CRC is supported.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	CRC_NOT_SUPPORTED CRC_SUPPORTED	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	FrTSynCyclicMsgResumeTime	
Description	Defines the time where the 1st regular cycle time based message transmission takes place, after an immediate transmission before. Unit: seconds	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	AUTOSAR_ECUC

Parameter Name	**FrTSynGlobalTimeDebounceTime**	
Description	This represents the configuration of a TX debounce time for SYNC and OFS messages compared to a message before with the same PDU. Unit: seconds	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynImmediateTimeSync**	
Description	Enables/Disables the cyclic polling of StbM_GetTimeBaseUpdateCounter() within FrTSyn_MainFunction().	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.5. FrTSynGlobalTimeMasterPdu

Parameters included	
Parameter name	**Multiplicity**
FrTSynGlobalTimeMasterHandleId	1..1
FrTSynGlobalTimePduRef	1..1

Parameter Name	**FrTSynGlobalTimeMasterHandleId**	
Description	This represents the handle ID of the PDU that contains the global time information.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	AUTOSAR_ECUC

Parameter Name	**FrTSynGlobalTimePduRef**	
Description	This represents the reference to the Pdu taken to transmit the global time information. The global time master of a global time domain acts as the sender of the Pdu while all the time slaves are supposed to receive the Pdu.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.6. FrTSynGlobalTimeSlave

Containers included		
Container name	**Multiplicity**	**Description**
FrTSynGlobalTimeSlavePdu	1..1	This container carries all properties required to configure the PDU received by the time slave for the given global time domain.

Parameters included	
Parameter name	**Multiplicity**
FrTSynRxCrcValidated	1..1
FrTSynGlobalTimeSequenceCounterJumpWidth	1..1

Parameter Name	**FrTSynRxCrcValidated**	
Description	This parameter controls whether or not CRC validation shall be supported.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	CRC_IGNORED	
Range	CRC_IGNORED	
	CRC_NOT_VALIDATED	
	CRC_OPTIONAL	
	CRC_VALIDATED	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	AUTOSAR_ECUC

Parameter Name	**FrTSynGlobalTimeSequenceCounterJumpWidth**	
Description	The SequenceCounterJumpWidth specifies the maximum allowed gap of the Sequence Counter between two SYNC resp. two OFS messages.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=15 >=1	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.7. FrTSynGlobalTimeSlavePdu

Parameters included	
Parameter name	**Multiplicity**
FrTSynGlobalTimeSlaveHandleId	1..1
FrTSynGlobalTimePduRef	1..1

Parameter Name	**FrTSynGlobalTimeSlaveHandleId**	
Description	This represents the handle ID of the PDU that contains the global time information.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynGlobalTimePduRef**
Description	This represents the reference to the Pdu taken to transmit the global time information. The global time master of a global time domain acts as the sender of the Pdu while all the time slaves are supposed to receive the Pdu.
Multiplicity	1..1

Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.8. FrTSynGlobalTimeOfsDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
FrTSynGlobalTimeOfs-DataIDListElement	16..16	Element of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process. Element of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.

5.4.1.9. FrTSynGlobalTimeOfsDataIDListElement

Parameters included	
Parameter name	**Multiplicity**
FrTSynGlobalTimeOfsDataIDListIndex	1..1
FrTSynGlobalTimeOfsDataIDListValue	1..1

Parameter Name	**FrTSynGlobalTimeOfsDataIDListIndex**	
Description	Index of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process. Index of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=15 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynGlobalTimeOfsDataIDListValue**

Description	Value of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process. Value of the DataIDList for OFS messages ensures the identification of data elements due to CRC calculation process.
Multiplicity	1..1
Type	INTEGER
Range	<=255 >=0
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

5.4.1.10. FrTSynGlobalTimeSyncDataIDList

Containers included		
Container name	**Multiplicity**	**Description**
FrTSynGlobalTimeSyncDataIDListElement	16..16	Element of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process. Element of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.

5.4.1.11. FrTSynGlobalTimeSyncDataIDListElement

Parameters included	
Parameter name	**Multiplicity**
FrTSynGlobalTimeSyncDataIDListIndex	1..1
FrTSynGlobalTimeSyncDataIDListValue	1..1

Parameter Name	**FrTSynGlobalTimeSyncDataIDListIndex**
Description	Index of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process. Index of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.
Multiplicity	1..1

Type	INTEGER	
Range	<=15 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**FrTSynGlobalTimeSyncDataIDListValue**	
Description	Value of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process. Value of the DataIDList for SYNC messages ensures the identification of data elements due to CRC calculation process.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=255 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.4.1.12. PublishedInformation

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

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

5.4.2. Application programming interface (API)

5.4.2.1. Macro constants

5.4.2.1.1. FRTSYN_E_INIT_FAILED

Purpose	
Value	0x22U
Description	DET error code: FrTSyn initialization failed

5.4.2.1.2. FRTSYN_E_INVALID_PDUID

Purpose	
Value	0x01U
Description	DET error code: Module called with wrong PDU or SDU ID

5.4.2.1.3. FRTSYN_E_INV_CTRL_IDX

Purpose	
Value	0x24U
Description	DET error code: Invalid Controller index

5.4.2.1.4. FRTSYN_E_NULL_POINTER

Purpose	
Value	0x21U
Description	DET error code: Invalid pointer (NULL_PTR)

5.4.2.1.5. FRTSYN_E_PARAM

Purpose	
Value	0x23U

Description	DET error code: API called with invalid parameter

5.4.2.1.6. FRTSYN_E_UNINIT

Purpose	
Value	0x20U
Description	DET error code: API service used in un-initialized state

5.4.2.1.7. FRTSYN_GET_DELTA_DIFF

Purpose	
Value	(((delta) > (deltaMax)) ? ((delta) - (deltaMax)) : ((deltaMax) - (delta)))

5.4.2.1.8. FRTSYN_GET_NEW_SEQ_COUNTER

Purpose	
Value	(((seqCounter) < FRTSYN_MAX_SEQ_COUNTER)?((seqCounter)+1U):(0U))

5.4.2.1.9. FRTSYN_INSTANCE_ID

Purpose	Module instance ID.
Value	0U
Description	Defines the instance number of this module. Since multiple instances are not supported this ID is always zero.

5.4.2.1.10. FRTSYN_IS_CRC_SUPPORTED

Purpose	
Value	((FrTSyn_MasterConfig[(index)].txCrcSecured == FRTSYN_CRC_SUPPORTED)?TRUE:FALSE)

5.4.2.1.11. FRTSYN_IS_MESSAGE_TYPE_VALID

Purpose	

Value	((((uint8)(value) == (uint8)FrTSyn_SlaveConfig[(index)].rxCrcValidated) \ && (((uint8)(value) != (uint8)(FRTSYN_CRC_IGNORED)) && ((uint8)(value) != (uint8)(FRTSYN_CRC_OPTIONAL)))) \ || ((((uint8)(FRTSYN_CRC_IGNORED) == (uint8)FrTSyn_SlaveConfig[(index)].rxCrcValidated)|| ((uint8)(FRTSYN_-CRC_OPTIONAL) == (uint8)FrTSyn_SlaveConfig[(index)].rxCrcValidated)) \ && (((FrTSyn_SlaveConfig[(index)].timeDomainId < (uint8)0x10) && (((uint8)(val-ue) == (uint8)FRTSYN_SYNC_CRC_NOT_VALIDATED) || ((uint8)(value) == (uint8)FRTSYN_SYNC_CRC_VALIDATED)))\ ||((FrTSyn_SlaveConfig[(index)].time-DomainId > (uint8)0x0F) && (((uint8)(value) == (uint8)FRTSYN_OFS_CRC_NOT_-VALIDATED) || ((uint8)(value) == (uint8)FRTSYN_OFS_CRC_VALIDATED))))))

5.4.2.1.12. FRTSYN_MAX_SEQ_COUNTER

Purpose	
Value	15U

5.4.2.1.13. FRTSYN_SID_GETVERSIONINFO

Purpose	Defines API id of function FrTSyn_GetVersionInfo().
Value	0x02U

5.4.2.1.14. FRTSYN_SID_INIT

Purpose	Defines API id of function FrTSyn_Init().
Value	0x01U

5.4.2.1.15. FRTSYN_SID_MAINFUNCTION

Purpose	Defines API id of function FrTSyn_MainFunction().
Value	0x04U

5.4.2.1.16. FRTSYN_SID_RXINDICATION

Purpose	Defines API id of function FrTSyn_RxIndication().
Value	0x42U

5.4.2.1.17. FRTSYN_SID_SETTRANSMISSIONMODE

Purpose	Defines API id of function FrTSyn_SetTransmissionMode().
Value	0x03U

5.4.2.1.18. FRTSYN_SID_TRIGGERTRANSMIT

Purpose	Defines API id of function FrTSyn_TriggerTransmit().
Value	0x41U

5.4.2.2. Functions

5.4.2.2.1. FrTSyn_GetVersionInfo

Purpose	API to get the module version information.	
Synopsis	`void FrTSyn_GetVersionInfo ( Std_VersionInfoType * versioninfo );`	
Service ID	0x02	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (out)	`versioninfo`	- Pointer to return the module version information.
Description	This service returns the version information of this module.	

5.4.2.2.2. FrTSyn_Init

Purpose	Initializes the FrTSyn module.
Synopsis	`void FrTSyn_Init ( const FrTSyn_ConfigType * configPtr );`
Service ID	0x01
Sync/Async	Synchronous
Reentrancy	Non Reentrant

Parameters (in)	`CfgPtr`	- Address of the configuration data structure.
Description	This service initializes the FrTSyn module. It shall be the first function of the module to be called.	

5.4.2.2.3. FrTSyn_MainFunction

Purpose	FrTSyn module main function.
Synopsis	`void` **`FrTSyn_MainFunction`** `( void );`
Service ID	0x04
Sync/Async	Synchronous
Reentrancy	Non Reentrant

5.4.2.2.4. FrTSyn_RxIndication

Purpose	Rx-Indication function.	
Synopsis	`void` **`FrTSyn_RxIndication`** `( PduIdType RxPduId , PduInfoType * PduInfoPtr );`	
Service ID	0x42	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non Reentrant for the same PduId.	
Parameters (in)	`RxPduId`	- ID of the received I-PDU.
	`PduInfoPtr`	- Pointer to a buffer containing the I-PDU.
Description	This service is called by FrIf in case a reception is indicated.	

5.4.2.2.5. FrTSyn_SetTransmissionMode

Purpose	Initializes the FrTSyn module.
Synopsis	`void` **`FrTSyn_SetTransmissionMode`** `( uint8 CtrlIdx , FrTSyn_TransmissionModeType Mode );`
Service ID	0x03
Sync/Async	Synchronous
Reentrancy	Non Reentrant

Parameters (in)	`CtrlIdx`	- Index of the FlexRay channel.
	`Mode`	- FRTSYN_TX_OFF ► FRTSYN_TX_ON
Description	This API is used to turn on and off the TX capabilities of the FrTSyn.	

5.4.2.2.6. FrTSyn_TriggerTransmit

Purpose	Trigger-Transmit function.	
Synopsis	`Std_ReturnType` **`FrTSyn_TriggerTransmit`** `( PduIdType TxPduId , PduInfoType * PduInfoPtr );`	
Service ID	0x41	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non Reentrant for the same PduId.	
Parameters (in)	`TxPduId`	- ID of the received I-PDU.
	`PduInfoPtr`	- Pointer to a buffer containing the I-PDU.
Return Value	Result of operation	
	`E_OK:`	SDU has been copied and SduLength indicates the number of copied bytes.
	`E_NOT_OK:`	No SDU has been copied. PduInfoPtr must not be used since it may contain a NULL pointer or point to invalid data.
Description	This service is called by FrIf in case a triggered transmit.	

5.4.2.2.7. FrTSyn_TxConfirmation

Purpose	TxConfirmation dummy function.	
Synopsis	`void` **`FrTSyn_TxConfirmation`** `( PduIdType TxPduId );`	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non Reentrant for the same PduId.	
Parameters (in)	`TxPduId`	- ID of the received I-PDU.
Description	This service is called by FrIf in case of a Tx Confirmation. But since this is a dummy, this function does not provide any functionality.	

5.4.3. Integration notes

5.4.3.1. Exclusive areas

This section describes the exclusive areas used by the `FrTSyn` module.

5.4.3.1.1. SCHM_FRTSYN_EXCLUSIVE_AREA_0

Protected data structures	The exclusive area protects the shared variable access of all global variables.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. The locking mechanism can be disabled if it is ensured that: ▶ `FrTSyn_MainFunction()` does not preempt `FrTSyn_RxIndication()` and vice versa. ▶ No FrTSyn API function preempts `FrTSyn_Init()`.

5.4.3.1.2. SCHM_FRTSYN_EXCLUSIVE_AREA_1

Protected data structures	In order to provide enhanced precision of the Virtual Local Time the module FrTSyn requires this exclusive area to increase the precision of the Virtual Local Time.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the `EB tresos AutoCore Generic documentation`. Refer to the section `Mapping exclusive areas in the basic software modules` in the `Integration notes` section for details. The locking mechanism can be disabled if the Time Base that is referenced in StbM uses an OsCounter as Hardware reference.

5.4.3.2. Production errors

Production errors information is not available for this module.

5.4.3.3. Memory mapping

General information about memory mapping is provided in the EB tresos AutoCore Generic documentation. Refer to the section `Memory mapping and compiler abstraction` in the `Integration notes` section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section
CONFIG_DATA_UNSPECIFIED
CONFIG_DATA_8
CONST_UNSPECIFIED
VAR_INIT_8
VAR_CLEARED_UNSPECIFIED
VAR_CLEARED_8
CODE

5.4.3.4. Integration requirements

WARNING 

Integration requirements list is not exhaustive

The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user's guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

5.4.3.4.1. lim.FrTSyn.EB_INTREQ_FrTSyn_0001

Description	Limitation on order of invocation of FrTSyn_Init function. The invocations of FrTSyn_-Init, should be called after FrIf_Init is called.
Rationale	The calls to FrIf_GetMacrotickDuration() and FrIf_GetMacroticksPerCycle() will raise a DET, because these functions are called during initialization of the FrTSyn module.

5.4.3.4.2. lim.FrTSyn.EB_INTREQ_FrTSyn_0002

Description	It shall be assured that the TxPeriod is configured in a way that a second transmission does not interrupt the TriggerTransmit for the first one (on the same TimeDomain).
Rationale	FrTSyn can not handle this scenario since it has no control over TriggerTransmit.

5.5. StbM

5.5.1. 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.
PublishedInformation	1..1	**Label:** EB Published Information Additional published parameters not covered by CommonPublishedInformation container.
StbMDefensiveProgramming	1..1	**Label:** Defensive Programming Options Parameters for defensive programming
StbMGeneral	1..1	Container holding the general parameters for the Synchronized Time-base Manager.
StbMSynchronizedTimeBase	1..n	Container holding the list of Synchronized time-base providers. The Synchronized time-base provider collects the information about specific time-base/s within the system.
StbMTriggeredCustomer	0..n	The triggered customer is directly triggered by the Synchronized Timebase Manager by getting synchronized with the current (global) definition of time and passage of time.
StbMBswModules	0..n	Defines the schemaNode path of an adjacent BSW module. Each container describes a specific BSW module adjacent to and controlled by the StbM. There are three preconditions to fully integrate a BSW module to the StbM:

Containers included		
		▶ The BSW module shall provide a property file defining specific parameters. ▶ This property file shall be registered at startup of tresos Studio. ▶ The BWM module shall be added to this list (by its schemaNode path). The StbM considers the property files of all BSW module defined here. Modules without property file must not be defined in this list. A BSW module without a property file is not fully integrated and suffers the following limitations: ▶ StbM can not control a master time domain of this BSW module. ▶ StbM can not use a possible hardware timestamp capability of this BSW module. ▶ The StbM configuration must contain a reserved time base for this BSW module. ▶ Parameter StbMDevErrorDetect must be disabled. A BSW module with a property file shall define the following parameters for successful integration: ▶ Name of module config. ▶ List of header files for inclusion. ▶ List of time base references of master time domains. ▶ List of time base references of slave time domains. ▶ Support of hardware timestamps. ▶ Name of time getter API. ▶ Name of time setter API.

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

Parameter Name	IMPLEMENTATION_CONFIG_VARIANT

Multiplicity	1..1	
Type	ENUMERATION	
Default value	VariantPreCompile	
Range	VariantPreCompile	
Configuration class	**VariantPreCompile:**	VariantPreCompile

5.5.1.1. CommonPublishedInformation

Parameters included	
Parameter name	**Multiplicity**
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	4	
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	6
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	4
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	0
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	160
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

5.5.1.2. PublishedInformation

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

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

5.5.1.3. StbMDefensiveProgramming

Parameters included	
Parameter name	**Multiplicity**
StbMDefProgEnabled	1..1
StbMPrecondAssertEnabled	1..1
StbMPostcondAssertEnabled	1..1
StbMStaticAssertEnabled	1..1
StbMUnreachAssertEnabled	1..1
StbMInvariantAssertEnabled	1..1

Parameter Name	**StbMDefProgEnabled**

Label	Enable Defensive Programming
Description	Enables or disables the defensive programming feature for the module StbM. Note: This feature is dependent on the use of the development error detection module. To use the defensive programming feature, proceed as follows: 1. Enable development error detection 2. Enable defensive programming 3. Enable assertions as required
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

Parameter Name	**StbMPrecondAssertEnabled**
Label	Enable Precondition Assertions
Description	Enables handling of precondition assertion checks reported from the module StbM. Dependency on parameter(s): ► Enable Development Error Detection (`StbMDevErrorDetect`): must be enabled ► Enable Defensive Programming (`StbMDefProgEnabled`): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

Parameter Name	**StbMPostcondAssertEnabled**
Label	Enable Postcondition Assertions
Description	Enables handling of postcondition assertion checks reported from the module StbM. Dependency on parameter(s):

	► Enable Development Error Detection (`StbMDevErrorDetect`): must be enabled ► Enable Defensive Programming (`StbMDefProgEnabled`): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

Parameter Name	**StbMStaticAssertEnabled**
Label	Enable Static Assertions
Description	Enables handling of static assertion checks reported from the module StbM. Dependency on parameter(s): ► Enable Development Error Detection (`StbMDevErrorDetect`): must be enabled ► Enable Defensive Programming (`StbMDefProgEnabled`): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

Parameter Name	**StbMUnreachAssertEnabled**
Label	Enable Unreachable Code Assertions
Description	Enables handling of unreachable code assertion checks reported from the module StbM. Dependency on parameter(s): ► Enable Development Error Detection (`StbMDevErrorDetect`): must be enabled ► Enable Defensive Programming (`StbMDefProgEnabled`): must be enabled
Multiplicity	1..1

Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	Elektrobit Automotive GmbH	

Parameter Name	**StbMInvariantAssertEnabled**
Label	Enable Invariant Assertions
Description	Enables handling of invariant assertion checks reported from functions of the module StbM. Dependency on parameter(s): ► Enable Development Error Detection (`StbMDevErrorDetect`): must be enabled ► Enable Defensive Programming (`StbMDefProgEnabled`): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

5.5.1.4. StbMGeneral

Containers included		
Container name	**Multiplicity**	**Description**
StbMEcucPartitionRefList	0..50	

Parameters included	
Parameter name	**Multiplicity**
StbMDevErrorDetect	1..1
StbMGetCurrentTimeExtendedAvailable	1..1
StbMMainFunctionPeriod	1..1
StbMTimeRecordingSupport	1..1
StbMTimerStartThreshold	1..1

Parameters included	
StbMVersionInfoApi	1..1
StbMGptTimerRef	1..1
StbMRteUsage	1..1
StbMTimerStartMaxNumberOfCalls	1..1
StbMMulticoreSupport	1..1

Parameter Name	StbMDevErrorDetect	
Description	Enables and disables the development error detection and notification mechanism. ► True: Enabled ► False: Disabled **Optimization Effect:** ► **ROM increase (code):** Enabling this parameter increases the ROM consumption of the module code. ► **Execution time increase (code):** Enabling this parameter increases the execution time of the module code.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMGetCurrentTimeExtendedAvailable	
Description	This allows to define whether an additional variant of the API GetCurrentTime with a 64 bit argument is provided.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMMainFunctionPeriod

Description	Schedule period of the main function StbM_MainFunction. Unit: [s].	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.01	
Range	<=1000 >0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMTimeRecordingSupport**	
Description	Enables/Disables the usage of the recording functionality for Synchronized and Offset timebases for Global Time precision measurement purpose.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMTimerStartThreshold**	
Description	EN: This interval defines, when a GPT Timer shall be started for Time Notification Customers for which the corresponding Customer Timer is running [unit: seconds]. This parameter is enabled only if there are StbMNotificationCostumers configured.	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMVersionInfoApi**
Description	Enables and disables the API StbM_GetVersionInfo(). ► True: Enabled ► False: Disabled

	Optimization Effect: ► **ROM increase (code):** Enabling this parameter increases the ROM consumption of the module code.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMGptTimerRef**
Description	EN: This represents an optional sub-container in case any Time Notification Customer is configured. This parameter is enabled only if there are StbMNotificationCostumers configured. If StbMGptTimerRef is configured, please make sure that GptNotification is enabled and configured! The referenced GptChannel shall not be used for other purposes, this is the only place it shall be configured!
Multiplicity	1..1
Type	REFERENCE
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

Parameter Name	**StbMRteUsage**
Description	Enables RTE Usage. If enabled, the StbM will generate an SWCD and supply the specified software component interfaces.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

Parameter Name	**StbMTimerStartMaxNumberOfCalls**

Description	EN: This defines the maximum number of StbM_StartTimer() function calls that StbM can handle at once during runtime. If this is exceded, StbM_StartTimer() will return E_NOT_OK.
Multiplicity	1..1
Type	INTEGER
Range	<4294967295 >0
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

Parameter Name	**StbMMulticoreSupport**
Description	Enables/Disables multicore support with respect to Rte cross-core calls. If this parameter is disabled, StbM is still multicore capable as long as the exclusive areas are correctly configured(i.e to spin-lock) ► True: Enabled ► False: Disabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

5.5.1.5. StbMEcucPartitionRefList

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

Parameter Name	**StbMEcucPartitionRef**
Description	Reference to EcucPartition, where StbM module is assigned to.
Multiplicity	1..1
Type	REFERENCE

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

5.5.1.6. StbMSynchronizedTimeBase

Containers included		
Container name	**Multiplicity**	**Description**
StbMLocalTimeClock	0..1	References the hardware reference clock of this Synchronized Time Base.
StbMNotificationCustomer	0..n	EN: This container holds the configuration of a notification customer, which is notified is informed about the occurrence of a Time-base related event.
StbMTimeCorrection	0..1	Collects the information relevant for the rate- and offset correction of a Time Base.
StbMTimeRecording	0..1	Collects the information relevant for configuration of the precision measurement of a Time Base.
StbMTimeValidation	0..1	Container with Time Validation configuration for Time Base.

Parameters included	
Parameter name	**Multiplicity**
StbMAllowSystemWideGlobalTimeMaster	0..1
StbMIsSystemWideGlobalTimeMaster	1..1
StbMNotificationInterface	1..1
StbMStoreTimebaseNonVolatile	0..1
StbMSyncLossTimeout	0..1
StbMSynchronizedTimeBaseIdentifier	1..1
StbMOffsetTimeBase	0..1
StbMClearTimeleapCount	0..1
StbMTimeLeapFutureThreshold	0..1
StbMTimeLeapPastThreshold	0..1
StbMStatusNotificationCallback	1..1
StbMStatusNotificationMask	1..1
StbMGlobalNvMBlockDescriptor	1..1
StbMUseOSGetTimeStamp	1..1

Parameters included	
StbMProvideDataInSharedMemory	1..1

Parameter Name	StbMAllowSystemWideGlobalTimeMaster	
Description	*For postbuild variant of the StbM this parameter has to be set to true for a Global Time Master that may act as a system-wide source of time. Otherwise no corresponding service ports/interfaces is provided. The Global Time Master functionality behind the service ports/interfaces has to be enabled/disabled separately via parameter StbMIsSystemWideGlobalTimeMaster.*	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMIsSystemWideGlobalTimeMaster	
Description	This parameter shall be set to true for a global time master that acts as a system-wide source of time information with respect to global time. It is possible that several global time masters exist that have set this parameter set to true because the global time masters exist once per global time domain and one ECU may start several global time domains on different busses it is connected to.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMNotificationInterface
Description	The parameter defines what type of interface shall be used to notify a customer of a status event.
Multiplicity	1..1
Type	ENUMERATION
Default value	NO_NOTIFICATION
Range	NO_NOTIFICATION CALLBACK

	SR_INTERFACE	
	CALLBACK_AND_SR_INTERFACE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMStoreTimebaseNonVolatile**	
Description	This allows for specifying that the timebase shall be stored in the NvRam.	
Multiplicity	0..1	
Type	ENUMERATION	
Range	NO_STORAGE	
	STORAGE_AT_SHUTDOWN	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMSyncLossTimeout**	
Description	This attribute describes the timeout for the situation that the time synchronization gets lost in the scope of the time domain. Unit: seconds	
Multiplicity	0..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMSynchronizedTimeBaseIdentifier**	
Description	ID of a synchronized time-base via a unique identifier. The value for this parameter can be calculated automatic or manually entered. Allowed ranges are: 0..15 for Synchronized Time Bases 16..31 for Offset Time Bases 32..127 for Pure Local Time Bases 123..65535 Reserved	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMOffsetTimeBase**
Description	Reference to another time base.

	In this case this reference is used, this time base is treated as an offset time base.	
Multiplicity	0..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMClearTimeleapCount**	
Description	This attribute describes the required number of updates to the Time Base where the time difference to the previous value has to remain below StbMTimeLeapPastThreshold/StbMTimeLeapFutureThreshold until the TIMELEAP_PAST/TIMELEAP_FUTURE bit within timeBaseStatus of the Time Base is cleared.	
Multiplicity	0..1	
Type	INTEGER	
Default value	1	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMTimeLeapFutureThreshold**	
Description	This represents the maximum allowed positive difference between a newly received Global Time Base value and the current Local Time Base value [unit: seconds].	
Multiplicity	0..1	
Type	FLOAT	
Default value	0.0	
Range	<Infinity >=0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMTimeLeapPastThreshold**
Description	This represents the maximum allowed negative difference between the current Local Time Base value and a newly received Global Time Base value [unit: seconds].
Multiplicity	0..1

Type	FLOAT	
Default value	0.0	
Range	<Infinity >=0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMStatusNotificationCallback**
Description	Name of the customer specific status notification callback function, which shall be called, if a non-masked status event occurs. Naming example: `StbM_StatusNotificationCallback` If the name of the Time Base is `StbMSync1`, the generated function name will be: `StbM_StatusNotificationCallbackStbMSync1`
Multiplicity	1..1
Type	FUNCTION-NAME
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

Parameter Name	**StbMStatusNotificationMask**
Description	The parameter defines the initial value for NotificationMask mask, which defines the events for which the event notification callback function shall be called.
Multiplicity	1..1
Type	INTEGER
Default value	0
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

Parameter Name	**StbMGlobalNvMBlockDescriptor**
Label	Global NVRAM Block Descriptor
Description	Reference to NVRAM block containing the non volatile data. Index of the NvMBlockDescriptor will be used by the StbM module for the identification of the NvM block

Multiplicity	1..1	
Type	SYMBOLIC-NAME-REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	Elektrobit Automotive GmbH	

Parameter Name	**StbMUseOSGetTimeStamp**	
Description	The time source of this Time Base will be given by calling OS_GetTimeStamp. Please make sure StbMLocalTimeClock is enabled and StbMLocalTimeHardware is left empty. ► True: Enabled ► False: Disabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	Elektrobit Automotive GmbH	

Parameter Name	**StbMProvideDataInSharedMemory**	
Description	The StbM provides the offset of the local time of this time base with respect to the Eth HwCounter. The value is provided to a shared location that should be mapped by the integrator in order for the value to be used by other 'user' modules. MemMap section: VAR_STBM_SHARED_TIME_DATA IMPORTANT : this must be activated only for Synchronized Time Bases that have a reference to a EthTSynGlobalTimeDomain (an Eth Driver), or for Offset Time Bases that have a reference to a Synchronized Time Base as described above. ► True: Enabled ► False: Disabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	Elektrobit Automotive GmbH

5.5.1.7. StbMLocalTimeClock

Parameters included	
Parameter name	**Multiplicity**
StbMClockFrequency	1..1
StbMClockPrescaler	1..1
StbMLocalTimeHardware	1..1

Parameter Name	**StbMClockFrequency**
Description	Represents the frequency [Hz] of the HW reference clock used by the StbM. In case StbMLocalTimeHardware has a reference to Os or StbMUseOSGetTimeStamp is enabled for this time base, StbMClockFrequency will not be used when converting Os ticks to nanoseconds, since Os provides an own macro to do this. StbMClockFrequency shall be configured to a value equal to Os frequency, in order to be mapped to the Header Element HWfrequency of the record table belonging to the Synchronized Time Base (if Time Recording is activated on the Time Base). Otherwise, it does not matter what value it has. In case StbMLocalTimeHardware has a reference to EthTSyn, StbMClockFrequency will not be used when converting, because no conversion from ticks to nanoseconds will take place at StbM level. It does not matter what value it has, StbM will always use a value of 1GHz. In case StbMLocalTimeHardware has a reference to Gpt, StbMClockFrequency/StbMClockPrescaler factor will be used when converting Gpt ticks to nanoseconds, and it is mandatory to configure both parameters. Typically the resulting frequency from the StbMClockFrequency/StbMClockPrescaler division should be the same as the one from the Gpt channel. For the following frequencies: 10 kHz, 16 kHz, 1,001675 MHz, 2 MHz, 2.08 MHz, 3 MHz, 3.25 MHz, 3.5 MHz, 4 MHz, 5 MHz, 6.25 MHz, 8 MHz, 10 MHz, 12.5 MHz, 15 MHz, 16 MHz, 20 MHz, 24 Mhz, 24.55 Mhz, 25 MHz, 26 MHz, 32 MHz, 33,3 MHz, 37.5 MHz, 40 MHz, 45 MHz, 48 MHz, 50 MHz, 55 MHz, 56 MHz, 60 MHz, 62.5 MHz, 64 MHz, 70 MHz, 75 MHz, 80 MHz, 90 MHz, 96 MHz, 100 MHz, 110 MHz, 120 MHz, 132 MHz, 133 MHz, 140 MHz, 125 MHz, 150 MHz, 160 MHz, 180 MHz, 200 MHz, 240 MHz, 250 MHz, 264 MHz, 300 MHz,

	330 MHz, 400 MHz, 800 MHz. the conversion is done in a more precise way. For other frequencies, a more general conversion takes place, which is less precise.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMClockPrescaler**
Description	Represents the prescaler to calculate the resulting frequency of the HW reference clock used by the StbM. In case StbMLocalTimeHardware has a reference to Os or StbMUseOSGetTimeStamp is enabled for this time base, StbMClockPrescaler will not be used when converting Os ticks to nanoseconds, since Os provides an own macro to do this. StbMClockPrescaler shall be configured, in order to be mapped to the Header Element HWfrequency of the record table belonging to the Synchronized Time Base (if Time Recording is activated on the Time Base). Otherwise, it does not matter what value it has. In case StbMLocalTimeHardware has a reference to EthTSyn, StbMClockPrescaler will not be used when converting, because no conversion from ticks to nanoseconds will take place at StbM level. It does not matter what value it has, StbM will always use a value of 1. In case StbMLocalTimeHardware has a reference to Gpt, StbMClockFrequency/StbMClockPrescaler factor will be used when converting Gpt ticks to nanoseconds, and it is mandatory to configure both parameters. Typically StbMClockPrescaler shall be configured in such a manner that the resulting frequency from the StbMClockFrequency/StbMClockPrescaler division should be the same as the one from the Gpt channel. For the following frequencies: 10 kHz, 16 kHz, 1,001675 MHz, 2 Mhz, 2.08 Mhz, 3 MHz, 3.25 MHz, 3.5 MHz, 4 MHz, 5 MHz, 6.25 MHz, 8 MHz, 10 MHz, 12.5 MHz, 15 MHz, 16 MHz, 20 MHz, 24 Mhz, 24.55 Mhz, 25 MHz, 26 MHz, 32 MHz, 33,3 MHz, 37.5 MHz, 40 MHz, 45 MHz, 48 MHz, 50 MHz, 55 MHz, 56 MHz, 60 MHz, 62.5 MHz, 64 MHz, 70 MHz, 75 MHz, 80 MHz, 90 MHz, 96 MHz, 100 MHz, 110 MHz, 120 MHz, 132 MHz, 133 MHz, 140 MHz, 125 MHz, 150 MHz, 160 MHz, 180 MHz, 200 MHz, 240 MHz, 250 MHz, 264 MHz, 300 MHz, 330 MHz, 400 MHz, 800 MHz. the conversion is done in a more precise way. For other frequencies, a more general conversion takes place, which is less precise.

Multiplicity	1..1	
Type	INTEGER	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMLocalTimeHardware**
Description	Reference to the local time hardware. In case a reference to Os is used, the designated OS counter has to be configured properly: 1. the counter is directly driven by a HW timer 2. the counter's `OsCounterTicksPerBase` is one tick in x nanoseconds [ns] In case a reference to EthTSyn is used, the following shall happen: the underlying Eth driver shall support hardware timestamping. In case a reference to a Gpt is used, the frequency shall be configured in such a manner, that the resulting frequency to be equal to the Gpt Channel Frequency. Attention (EthTSyn): This reference must point to a time domain which is connected to this time base. In detail this means that the time domain referenced by this parameter must refer to this time base in turn. Attention (for SYNC/PURE time bases): StbMLocalTimeHardware can be left empty for SYNC/PURE time bases, if and only if, StbMUseOSGetTimeStamp is enabled.
Multiplicity	1..1
Type	CHOICE-REFERENCE
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	AUTOSAR_ECUC

5.5.1.8. StbMNotificationCustomer

Parameters included	
Parameter name	**Multiplicity**
StbMNotificationCustomerId	1..1
StbMTimeNotificationCallback	0..1

Parameter Name	StbMNotificationCustomerId	
Description	EN: Identification of a event notification customer.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMTimeNotificationCallback	
Description	EN: Name of the customer specific notification callback function, which shall be called, if the time previously set by the customer is reached. The name of the StbMTimeNotificationCallback shall be configured to: `TimeNotificationCallback` The callback will be generated like this: `[Customer]_TimeNotificationCallback[TimeBase]` If the name of the Time Base is `StbMSynchronizedTimeBase_0_Sync`, and the name of the Notification Customer is `NotificationCustomer_1` the generated function name will be: `NotificationCustomer_1_TimeNotificationCallbackStbMSynchronizedTimeBase_0_Sync`	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	**VariantPreCompile:**	VariantPreCompile
	VariantPreCompile:	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.5.1.9. StbMTimeCorrection

Parameters included	
Parameter name	**Multiplicity**
StbMAllowMasterRateCorrection	0..1
StbMMasterRateDeviationMax	0..1

Parameters included	
StbMOffsetCorrectionAdaptionInterval	0..1
StbMOffsetCorrectionJumpThreshold	0..1
StbMRateCorrectionMeasurementDuration	0..1
StbMRateCorrectionsPerMeasurementDuration	0..1

Parameter Name	StbMAllowMasterRateCorrection	
Description	EN: This attribute describes whether the rate correction value of a Time Base can be set by StbM_SetRateCorrection(): - false: the rate correction value can not be set by StbM_SetRateCorrection() - true: the rate correction value can be set by StbM_SetRateCorrection()	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	false	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMMasterRateDeviationMax	
Description	This attribute describes the maximum allowed absolute value of the rate deviation value to be set by StbM_SetRateCorrection() [unit: ppm].	
Multiplicity	0..1	
Type	INTEGER	
Default value	0	
Range	<=32000 >=0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	StbMOffsetCorrectionAdaptionInterval
Description	EN: Defines the interval during which the adaptive rate correction cancels out the rate- and time deviation [unit: seconds]. Note: If the StbMOffsetCorrectionJumpThreshold parameter is disabled, or it's value is 0, StbMOffsetCorrectionAdaptionInterval is not used.

Multiplicity	0..1	
Type	FLOAT	
Range	<=5 >0.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMOffsetCorrectionJumpThreshold**	
Description	Threshold for the correction method. Deviations below this value will be corrected by a linear reduction over a defined timespan. Values equal- and greater than this value will be corrected by immediately setting the correct time- and rate in form of a jump [unit: seconds].	
Multiplicity	0..1	
Type	FLOAT	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMRateCorrectionMeasurementDuration**	
Description	EN: Definition of the time span [s] which is used to calculate the rate deviation. The StbMRateCorrectionMeasurementDuration parameter, and the resynchronization time of the StbM should be configured in a way that: - the difference (TGstop - TGstart) and - the difference (TVstop - Tvstart), do not exceed the value of 15 seconds. These are part of the rate correction calculation formula, (rrc = (TGStop ? TGStart) / (TVStop- TVStart)). If they do exceed the above value, an overflow could emerge when calculating the Rate Correction.	
Multiplicity	0..1	
Type	FLOAT	
Default value	1.0	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	AUTOSAR_ECUC

Parameter Name	**StbMRateCorrectionsPerMeasurementDuration**	
Description	Number of simultaneous rate measurements to determine the current rate deviation.	
Multiplicity	0..1	
Type	INTEGER	
Default value	1	
Range	<=65535 >=1	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.5.1.10. StbMTimeRecording

Parameters included	
Parameter name	**Multiplicity**
StbMOffsetTimeRecordBlockCallback	0..1
StbMOffsetTimeRecordTableBlockCount	1..1
StbMSyncTimeRecordBlockCallback	0..1
StbMSyncTimeRecordTableBlockCount	1..1

Parameter Name	**StbMOffsetTimeRecordBlockCallback**	
Description	Name of the customer specific callback function, which shall be called, if a measurement data for a Offset Time Base are available.	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMOffsetTimeRecordTableBlockCount**
Description	Represents the number of Blocks used for queing time measurement events for the Offset Time Base Record Table.

Multiplicity	1..1	
Type	INTEGER	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMSyncTimeRecordBlockCallback**	
Description	Name of the customer specific callback function, which shall be called, if a measurement data for a Synchronized Time Base are available.	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMSyncTimeRecordTableBlockCount**	
Description	Represents the number of Blocks used for queuing time measurement events for the Synchronized Time Base Record Table.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.5.1.11. StbMTimeValidation

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

Parameter Name	**StbMTimeValidationRecordTableBlockCount**	
Description	Size of record table for Time Validation (number of blocks).	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

5.5.1.12. StbMTriggeredCustomer

Parameters included	
Parameter name	**Multiplicity**
StbMTriggeredCustomerPeriod	1..1
StbMOSScheduleTableRef	1..1
StbMSynchronizedTimeBaseRef	1..1
StbMScheduleTableClockFreq	0..1
StbMScheduleTicksToNsConvFactor	0..1

Parameter Name	**StbMTriggeredCustomerPeriod**	
Description	Triggering period of the triggered customer called by the StbM_MainFunction. Note: The Triggering period value is configured in [us].	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=4294967295 >=1	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMOSScheduleTableRef**	
Description	Reference to synchronized OS ScheduleTables.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile
Origin	AUTOSAR_ECUC	

Parameter Name	**StbMSynchronizedTimeBaseRef**	
Description	Reference to the mandatory required synchronized time-base.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	**VariantPreCompile:**	VariantPreCompile

Origin	AUTOSAR_ECUC

Parameter Name	**StbMScheduleTableClockFreq**	
Description	This parameter defines the Clock Frequency [Hz] that drives the ScheduleTable. It is needed to be able to determine nanoseconds to ticks conversion in the most precise way. For the following frequencies: 10 kHz, 16 kHz, 1,001675 MHz, 2 Mhz, 2.08 Mhz, 3 MHz, 3.25 MHz, 3.5 MHz, 4 MHz, 5 MHz, 6.25 MHz, 8 MHz, 10 MHz, 12.5 MHz, 15 MHz, 16 MHz, 20 MHz, 24 Mhz, 24.55 Mhz, 25 MHz, 26 MHz, 32 MHz, 33,3 MHz, 37.5 MHz, 40 MHz, 45 MHz, 48 MHz, 50 MHz, 55 MHz, 56 MHz, 60 MHz, 62.5 MHz, 64 MHz, 70 MHz, 75 MHz, 80 MHz, 90 MHz, 96 MHz, 100 MHz, 110 MHz, 120 MHz, 132 MHz, 133 MHz, 140 MHz, 125 MHz, 150 MHz, 160 MHz, 180 MHz, 200 MHz, 240 MHz, 250 MHz, 264 MHz, 300 MHz, 330 MHz, 400 MHz, 800 MHz. the conversion is done in a more precise way. For other frequencies, a more general conversion takes place, which is less precise.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=4294967295	
	>=1	
Configuration class	**VariantPreCompile:**	VariantPreCompile
	VariantPreCompile:	VariantPreCompile
Origin	Elektrobit Automotive GmbH	

Parameter Name	**StbMScheduleTicksToNsConvFactor**
Description	This parameter defines a factor that would help obtain better results when converting nanoseconds to ticks. To improve the accuracy based on usage of the existing macros OS_TICKS2NS_xxx, the calculation could also be done differently to avoid large rounding effects on usage of small numbers as parameters to OS_TICKS2NS_xxx(). Instead of nanoseconds / OS_TICKS2NS_[!"$StbMOsCounterId"!](1U), nanoseconds * factor / OS_TICKS2NS_[!"$StbMOsCounterId"!](factor) formula is used, which reduces the theoretical maximum error at least by factor. For the given example of 80 MHz, using a factor of 2 would already reduce the error to 0 as there wouldn't be any error due to rounding on the integer calculations:

	OS_TICKS2NS_xx(2) = 2 * 5 / 2 * 5 = 25 (is the exact value of 2 ticks) compared to OS_TICKS2NS_xx(1) = 1 * 5 / 2 * 5 = 10 (instead of the exact value 12.5 for 1 tick)
Multiplicity	0..1
Type	INTEGER
Default value	1
Range	<=4 >=1
Configuration class	**VariantPreCompile:** VariantPreCompile
Origin	Elektrobit Automotive GmbH

5.5.1.13. StbMBswModules

5.5.2. Application programming interface (API)

5.5.2.1. Type definitions

5.5.2.1.1. StbM_ConfigType

Purpose	StbM configuration data type.	
Type	struct	
Members	`uint8 ConfigSet`	

5.5.2.1.2. StbM_CustomerIdType

Purpose	Enables the use of StbM's own definition for "StbM_CustomerIdType" type.
Type	`uint16`

Description	Unique identifier of a notification customer

5.5.2.1.3. StbM_MasterConfigType

Purpose	Enables the use of StbM's own definition for "StbM_MasterConfigType" type.
Type	`uint8`
Description	MasterConfigType

5.5.2.1.4. StbM_MeasurementType

Purpose	Structure which contains additional measurement data.	
Type	struct	
Members	`uint32 pathDelay`	Propagation delay in nanoseconds

5.5.2.1.5. StbM_OffsetRecordTableBlockType

Purpose	Enables the use of StbM's own definition for "StbM_OffsetRecordTableBlockType" type.	
Type	struct	
Members	`uint32 GlbSeconds`	Seconds of the Offset Time Base
	`uint32 GlbNanoSeconds`	Nanoseconds of the Offset Time Base
	`StbM_TimeBaseStatusType TimeBaseStatus`	Time Base Status of the Local Time Base directly after synchronization with the Global Time Base
Description	Offset Time Base Record Table Block	

5.5.2.1.6. StbM_OffsetRecordTableHeadType

Purpose	Enables the use of StbM's own definition for "StbM_OffsetRecordTableHeadType" type.	
Type	struct	
Members	`uint8 OffsetTimeDomain`	Time Domain 16..31
Description	Offset Time Base Record Table Header	

5.5.2.1.7. StbM_ProtocolParamType

Purpose	Enables the use of StbM's own definition for "StbM_ProtocolParamType" type.	
Type	struct	
Members	`StbM_TimeSyncType protocolType`	Indicates the underlying Time Sync module.
	`sint32 cumulativeScaledRateOffset`	The cumulative rate offset of the Time Master acc. to IEEE 802.1AS
	`uint16 gmTimeBaseIndicator`	The time base indicator of the current Global Time Master acc. to IEEE 802.1AS
	`sint32 lastGmPhaseChange`	The phase change of the current Global Time Master acc. to IEEE 802.1AS
	`uint32 scaledLastGmFreqChange`	The scaled last frequency change of the Global Time Master acc. to IEEE 802.1AS
Description	Defines TimeSync protocol specific parameters relevant for the individual TimeSync modules (only EthTSyn specific parameters are known so far) This structure defines TimeSync protocol specific parameters relevant for the individual TimeSync modules (only EthTSyn specific parameters are known so far)	

5.5.2.1.8. StbM_RateDeviationType

Purpose	Enables the use of StbM's own definition for "StbM_RateDeviationType" type.
Type	`sint16`
Description	Variables of this type are used to express a rate deviation in ppm.

5.5.2.1.9. StbM_SyncRecordTableBlockType

Purpose	Enables the use of StbM's own definition for "StbM_SyncRecordTableBlockType" type.	
Type	struct	
Members	`uint32 GlbSeconds`	Seconds of the Local Time Base directly after synchronization with the Global Time Base
	`uint32 GlbNanoSeconds`	Nanoseconds of the Local Time Base directly after synchronization with the Global Time Base

	`StbM_TimeBaseStatusType TimeBaseStatus`	Time Base Status of the Local Time Base directly after synchronization with the Global Time Base
	`uint32 VirtualLocalTimeLow`	HW counter reference value directly after synchronization with the Global Time Base
	`StbM_RateDeviationType RateDeviation`	Calculated Rate Deviation directly after rate deviation measurement
	`uint32 LocSeconds`	Seconds of the Local Time Base directly before synchronization with the Global Time Base
	`uint32 LocNanoSeconds`	Nanoseconds of the Local Time Base directly before synchronization with the Global Time Base
	`uint32 PathDelay`	Current propagation delay in nanoseconds
Description	Synchronized Time Base Record Table Block	

5.5.2.1.10. StbM_SyncRecordTableHeadType

Purpose	Enables the use of StbM's own definition for "StbM_SyncRecordTableHeadType" type.	
Type	struct	
Members	`uint8 SynchronizedTimeDomain`	Time Domain 0..15
	`uint32 HWfrequency`	HW Frequency in Hz
	`uint32 HWprescaler`	Prescaler value
Description	Synchronized Time Base Record Table Header	

5.5.2.1.11. StbM_SynchronizedTimeBaseType

Purpose	Enables the use of StbM's own definition for "StbM_SynchronizedTimeBaseType" type.
Type	`uint16`
Description	Represent the kind of synchronized time-base Variables of this type are used to represent the kind of synchronized time-base.

5.5.2.1.12. StbM_TimeBaseNotificationType

Purpose	Enables the use of StbM's own definition for "StbM_TimeBaseNotificationType" type.
Type	`uint32`
Description	Represents the time base notification type Variables of this type are used to represent the number of global time related events. The type definition is used for storing the events in the status variable NotificationEvents and for setting the mask variable NotificationMask which defines a subset of events for which an interrupt request shall be raised.

5.5.2.1.13. StbM_TimeBaseStatusType

Purpose	Enables the use of StbM's own definition for "StbM_TimeBaseStatusType" type.
Type	`uint8`
Description	Represents the time base status Bit 6 and 7 are always 0 (reserved for future usage) Variables of this type are used to express if and how a Local Time Base is synchronized to the Global Time Master. The type is a bit field of individual status bits, although not every combination is possible, i.e. any of the bits TIMEOUT, TIMELEAP_FUTURE, TIMELEAP_PAST and SYNC_TO_GATEWAY can only be set if the GLOBAL_TIME_BASE bit is set.

5.5.2.1.14. StbM_TimeDiffType

Purpose	Enables the use of StbM's own definition for "StbM_TimeDiffType" type.
Type	`sint32`
Description	Offset Time Base Record Table Header

5.5.2.1.15. StbM_TimeStampExtendedType

Purpose	Enables the use of StbM's own definition for "StbM_TimeStampExtendedType" type.	
Type	struct	
Members	`StbM_TimeBaseStatusType timeBaseStatus`	status of the time base
	`uint32 nanoseconds`	Nanoseconds part of the time
	`uint64 seconds`	48 bit Seconds part of the time

Description	Type to represent long time intervals Variables of this type are used for expressing time stamps including relative time and absolute calendar time. The absolute time starts from 1970-01-01. Start of absolute time (1970-01-01) is according to [17], Annex C/C1 (refer to parameter "approximate epoch" for PTP) 0 to 999999999ns 0 to 18446744073709551615s == 21350398233460d [0xFFFF FFFF FFFF FFFF] [0x3B9A C9FF] invalid value in nanoseconds: [0x3B9A CA00] to [0x3FFF FFFF] Bit 30 and 31 reserved, default: 0

5.5.2.1.16. StbM_TimeStampType

Purpose	Enables the use of StbM's own definition for "StbM_TimeStampType" type.	
Type	struct	
Members	`StbM_TimeBaseStatusType timeBaseStatus`	status of the time base
	`uint32 nanoseconds`	Nanoseconds part of the time
	`uint32 seconds`	32 bit LSB of the 48 bits Seconds part of the time
	`uint16 secondsHi`	16 bit MSB of the 48 bits Seconds part of the time
Description	Type to represent long time intervals Variables of this type are used for expressing time stamps including relative time and absolute calendar time. The absolute time starts from 1970-01-01 acc. to "[17], Annex C/C1" as specified for PTP. 0 to 281474976710655s == 3257812230d [0xFFFF FFFF FFFF] 0 to 999999999ns [0x3B9A C9FF] invalid value in nanoseconds: [0x3B9A CA00] to [0x3FFF FFFF] Bit 30 and 31 reserved, default: 0	

5.5.2.1.17. StbM_TimeSyncType

Purpose	Enables the use of StbM's own definition for "StbM_TimeSyncType" type.
Type	`uint8`
Description	Indicates the underlying Time Sync module

5.5.2.1.18. StbM_TimeTupleType

Purpose	Represent a Time Tuple.
Type	struct

Members	`StbM_TimeStampType globalTime`	Global Time
	`StbM_VirtualLocalTimeType virtualLocalTime`	Associated Virtual Local Time
Description	Variables of this type store a Time Tuple containing the Global Time and the associated Virtual Local Time.	

5.5.2.1.19. StbM_UserDataType

Purpose	Enables the use of StbM's own definition for "StbM_UserDataType" type.	
Type	struct	
Members	`uint8 userDataLength`	User Data Length in bytes
	`uint8 userByte0`	User Byte 0
	`uint8 userByte1`	User Byte 1
	`uint8 userByte2`	User Byte 2
Description	Current user data of the time base	

5.5.2.1.20. StbM_VirtualLocalTimeType

Purpose	Enables the use of StbM's own definition for "StbM_VirtualLocalTimeType" type.	
Type	struct	
Members	`uint32 nanosecondsLo`	Least significant 32 bits of the 64 bit Virtual Local Time
	`uint32 nanosecondsHi`	Most significant 32 bits of the 64 bit Virtual Local Time
Description	Represent the kind of synchronized time-base Variables of this type store time stamps of the Virtual Local Time. The unit is nanoseconds.	

5.5.2.2. Macro constants

5.5.2.2.1. STBM_ALL_TIME_LEAP_FLAGS_MASK

Purpose	TIMELEAP_FUTURE bit (Bit 4) and TIMELEAP_PAST bit (Bit 5) in StbM_TimeBaseStatusType.

Value	0x30U
Description	0x30: Mask for both the time leap future and time leap past bits set

5.5.2.2.2. STBM_API_BUSGETCURRENTTIME

Purpose	Service ID for StbM_BusGetCurrentTime() function.
Value	0x1FU

5.5.2.2.3. STBM_API_BUSSETGLOBALTIME

Purpose	Service ID for StbM_BusSetGlobalTime() function.
Value	0x0fU

5.5.2.2.4. STBM_API_GETBUSPROTOCOLPARAM

Purpose	Service ID for StbM_GetBusProtocolParam() function.
Value	0x29U

5.5.2.2.5. STBM_API_GETCURRENTTIME

Purpose	Service ID for StbM_GetCurrentTime() function.
Value	0x07U

5.5.2.2.6. STBM_API_GETCURRENTTIMEEXTENDED

Purpose	Service ID for StbM_GetCurrentTimeExtended() function.
Value	0x08U

5.5.2.2.7. STBM_API_GETCURRENTVIRTUALLOCALTIME

Purpose	Service ID for StbM_GetCurrentVirtualLocalTime() function.

Value	0x1EU

5.5.2.2.8. STBM_API_GETMASTERCONFIG

Purpose	Service ID for StbM_GetMasterConfig() function.
Value	0x1DU

5.5.2.2.9. STBM_API_GETOFFSET

Purpose	Service ID for StbM_GetOffset() function.
Value	0x0eU

5.5.2.2.10. STBM_API_GETOFFSETTIMERECORDHEAD

Purpose	Service ID for StbM_GetOffsetTimeRecordHead() function.
Value	0x17U

5.5.2.2.11. STBM_API_GETRATEDEVIATION

Purpose	Service ID for StbM_GetRateDeviation() function.
Value	0x11U

5.5.2.2.12. STBM_API_GETSYNCTIMERECORDHEAD

Purpose	Service ID for StbM_GetSyncTimeRecordHead() function.
Value	0x16U

5.5.2.2.13. STBM_API_GETTIMEBASESTATUS

Purpose	Service ID for StbM_GetTimeBaseStatus() function.
Value	0x14U

5.5.2.2.14. STBM_API_GETTIMEBASEUPDATECOUNTER

Purpose	Service ID for StbM_GetTimeBaseUpdateCounter() function.
Value	0x1bU

5.5.2.2.15. STBM_API_GETTIMELEAP

Purpose	Service ID for StbM_GetTimeLeap() function.
Value	0x13U

5.5.2.2.16. STBM_API_GETVERSIONINFO

Purpose	Service ID for StbM_GetVersionInfo() function.
Value	0x05U

5.5.2.2.17. STBM_API_INIT

Purpose	Service ID for StbM_Init() function.
Value	0x00U

5.5.2.2.18. STBM_API_SETBUSPROTOCOLPARAM

Purpose	Service ID for StbM_SetBusProtocolParam() function.
Value	0x2AU

5.5.2.2.19. STBM_API_SETGLOBALTIME

Purpose	Service ID for StbM_SetGlobalTime() function.
Value	0x0bU

5.5.2.2.20. STBM_API_SETOFFSET

Purpose	Service ID for StbM_SetOffset() function.

Value	0x0dU

5.5.2.2.21. STBM_API_SETRATECORRECTION

Purpose	Service ID for StbM_SetRateCorrection() function.
Value	0x12U

5.5.2.2.22. STBM_API_SETUSERDATA

Purpose	Service ID for StbM_SetUserData() function.
Value	0x0cU

5.5.2.2.23. STBM_API_STARTTIMER

Purpose	Service ID for StbM_StartTimer() function.
Value	0x15U

5.5.2.2.24. STBM_API_TRIGGERTIMETRANSMISSION

Purpose	Service ID for StbM_TriggerTimeTransmission() function.
Value	0x1cU

5.5.2.2.25. STBM_API_UPDATEGLOBALTIME

Purpose	Service ID for StbM_UpdateGlobalTime() function.
Value	0x10U

5.5.2.2.26. STBM_EV_GLOBAL_TIME

Purpose	Status notification event EV_GLOBAL_TIME.
Value	0x01U

Description	Bit 0 (LSB): 0: synchronization to global time master not changed 1: GLOBAL_TIME_-BASE in StbM_TimeBaseStatusType has changed from 0 to 1

5.5.2.2.27. STBM_EV_RATECORRECTION

Purpose	Status notification event STBM_EV_RATECORRECTION.
Value	0x0400U
Description	Bit 10 1: A valid rate correction has been calculated (not beyond limits) 0: No rate correction calculated

5.5.2.2.28. STBM_EV_RESYNC

Purpose	Status notification event STBM_EV_RESYNC.
Value	0x0200U
Description	Bit 9 1: A synchronization of the local time to the valid Global Time value has occurred 0: No resynchronization event occurred

5.5.2.2.29. STBM_EV_SYNC_TO_GLOBAL_MASTER

Purpose	Status notification event EV_SYNC_TO_GLOBAL_MASTER.
Value	0x100U
Description	Bit 8 1: SYNC_TO_GATEWAY bit of Time Domain changes from 1 to 0 0: otherwise

5.5.2.2.30. STBM_EV_SYNC_TO_SUBDOMAIN

Purpose	Status notification event EV_SYNC_TO_SUBDOMAIN.
Value	0x80U
Description	Bit 7 1: SYNC_TO_GATEWAY bit in timeBaseStatus has changed from 0 to 1 0: otherwise

5.5.2.2.31. STBM_EV_TIMELEAP_FUTURE

Purpose	Status notification event EV_TIMELEAP_FUTURE.

Value	0x08U
Description	Bit 3 1: TIMELEAP_FUTURE bit in timeBaseStatus has changed from 0 to 1 0: otherwise

5.5.2.2.32. STBM_EV_TIMELEAP_FUTURE_REMOVED

Purpose	Status notification event EV_TIMELEAP_FUTURE_REMOVED.
Value	0x10U
Description	Bit 4 1: TIMELEAP_FUTURE bit in timeBaseStatus has changed from 1 to 0 0: otherwise

5.5.2.2.33. STBM_EV_TIMELEAP_PAST

Purpose	Status notification event EV_TIMELEAP_PAST.
Value	0x20U
Description	Bit 5 1: TIMELEAP_PAST bit in timeBaseStatus has changed from 0 to 1 0: otherwise

5.5.2.2.34. STBM_EV_TIMELEAP_PAST_REMOVED

Purpose	Status notification event EV_TIMELEAP_PAST_REMOVED.
Value	0x40U
Description	Bit 6 1: TIMELEAP_PAST bit in timeBaseStatus has changed from 1 to 0 0: otherwise

5.5.2.2.35. STBM_EV_TIMEOUT_OCCURRED

Purpose	Status notification event EV_TIMEOUT_OCCURRED.
Value	0x02U
Description	Bit 1: 1: TIMEOUT bit in timeBaseStatus has changed from 0 to 1 0: otherwise

5.5.2.2.36. STBM_EV_TIMEOUT_REMOVED

Purpose	Status notification event EV_TIMEOUT_REMOVED.

Value	0x04U
Description	Bit 2 1: TIMEOUT bit in timeBaseStatus has changed from 1 to 0 0: otherwise

5.5.2.2.37. STBM_E_INIT_FAILED

Purpose	StbM DET error code for initalization failure during StbM_Init.
Value	0x11U

5.5.2.2.38. STBM_E_PARAM

Purpose	StbM DET error code for API call with invalid parameter value.
Value	0x0aU

5.5.2.2.39. STBM_E_PARAM_POINTER

Purpose	StbM DET error code for API call with invalid pointer address.
Value	0x10U

5.5.2.2.40. STBM_E_PARAM_TIMESTAMP

Purpose	StbM DET error code for API call with invalid timestamp.
Value	0x25U

5.5.2.2.41. STBM_E_PARAM_USERDATA

Purpose	StbM DET error code for API call with invalid user data.
Value	0x26U

5.5.2.2.42. STBM_E_SERVICE_DISABLED

Purpose	StbM DET error code for API disabled by configuration.

Value	0x12U

5.5.2.2.43. STBM_E_UNINIT

Purpose	StbM DET error code for API call before module initialization.
Value	0x0bU

5.5.2.2.44. STBM_GLOBAL_TIME_BASE_FLAG

Purpose	GLOBAL_TIME_BASE bit (Bit 3) in StbM_TimeBaseStatusType.
Value	0x8U
Description	0x00: Local Time Base is based on Local Time Base reference clock only (never synchronized with Global Time Base) 0x01: Local Time Base was at least synchronized with Global Time Base one time

5.5.2.2.45. STBM_SYNC_TO_GATEWAY_FLAG

Purpose	SYNC_TO_GATEWAY bit (Bit 2) in StbM_TimeBaseStatusType.
Value	0x4U
Description	0x00: Local Time Base is synchronous to Global Time Master 0x01: Local Time Base updates are based on a Time Gateway below the Global Time Master

5.5.2.2.46. STBM_TIMEOUT_FLAG

Purpose	TIMEOUT bit (Bit 0) in StbM_TimeBaseStatusType.
Value	0x1U
Description	0x00: No Timeout on receiving Synchronization Messages 0x01: Timeout on receiving Synchronization Messages

5.5.2.2.47. STBM_TIMESYNC_CAN

Purpose	STBM_TIMESYNC_CAN in StbM_TimeSyncType.
Value	2U

Description	0x02: Indicates Time Synchronization on CAN

5.5.2.2.48. STBM_TIMESYNC_ETHERNET

Purpose	STBM_TIMESYNC_ETHERNET in StbM_TimeSyncType.
Value	1U
Description	0x01: Indicates Time Synchronization on Ethernet

5.5.2.2.49. STBM_TIMESYNC_FLEXRAY

Purpose	STBM_TIMESYNC_FLEXRAY in StbM_TimeSyncType.
Value	3U
Description	0x03: Indicates Time Synchronization on Flexray

5.5.2.2.50. STBM_TIME_LEAP_FUTURE_FLAG

Purpose	TIMELEAP_FUTURE bit (Bit 4) in StbM_TimeBaseStatusType.
Value	0x10U
Description	0x00: No leap within the received time 0x10: Leap within the received time that exceeds a configured threshold in time leap future

5.5.2.2.51. STBM_TIME_LEAP_PAST_FLAG

Purpose	TIMELEAP_PAST bit (Bit 5) in StbM_TimeBaseStatusType.
Value	0x20U
Description	0x00: No leap within the received time 0x20: Leap within the received time that exceeds a configured threshold in time leap past

5.5.2.3. Objects

5.5.2.3.1. StbM_TimeStamps

Purpose	Actual time tuple of all synchronized and offset time bases.

Type	StbM_TimeTupleType

5.5.2.4. Functions

5.5.2.4.1. StbM_BusGetCurrentTime

Purpose	Service to get current Time Tuple.	
Synopsis	`Std_ReturnType` **`StbM_BusGetCurrentTime`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_TimeStampType * globalTimePtr , StbM_VirtualLocalTimeType * localTimePtr , StbM_UserDataType * userData );`	
Service ID	0x0A	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Parameters (out)	`globalTimePtr`	- Value of the local instance of the Global Time, which is sampled when the function is called.
	`localTimePtr`	- Value of the Virtual Local Time, which is sampled when the function is called.
	`userData`	- User data of the time base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Returns the Time Tuple (Global Time, Virtual Local Time) taken when the function is called.	

5.5.2.4.2. StbM_BusSetGlobalTime

Purpose	Service to set global time by bus.
Synopsis	`Std_ReturnType` **`StbM_BusSetGlobalTime`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_TimeStampType * globalTimePtr , const StbM_UserDataType * userDataPtr , const StbM_Measure-`

	mentType * measureDataPtr , const StbM_VirtualLocalTimeType * localTimePtr);	
Service ID	0x0f	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	timeBaseId	- ID of the synchronized time-base.
	globalTimePtr	- New Global Time value.
	userDataPtr	- User data of the time base.
	measureDataPtr	- New measurement data.
	localTimePtr	- Value of the Virtual Local Time associated to the new Global Time.
Return Value	the success/failure of the function call	
	E_OK	In case of successful call of the function.
	E_NOT_OK	In case of unsuccessful call of the function.
Description	Allows the Time Provider Modules to set the new Global Time tuple (i.e. the Received Time Tuple) to the StbM, which has been received on a bus.	

5.5.2.4.3. StbM_GetBusProtocolParam

Purpose	Service to get bus specific parameters from received Follow_Up message.	
Synopsis	Std_ReturnType **StbM_GetBusProtocolParam** (StbM_Synchronized-TimeBaseType timeBaseId , StbM_ProtocolParamType * protocol-Param);	
Service ID	0x29	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	timeBaseId	- Id of referenced Time Base
Parameters (out)	protocolParam	- structure to store received Follow_Up information TLV parameters
Return Value	the success/failure of the function call	
	E_OK	In case of successful call of the function.
	E_NOT_OK	In case of unsuccessful call of the function.

Description	This API is used to get bus specific parameters from received Follow_Up message

5.5.2.4.4. StbM_GetCurrentTime

Purpose	Service to get current time.	
Synopsis	`Std_ReturnType` **`StbM_GetCurrentTime`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_TimeStampType * timeStamp , StbM_UserDataType * userData );`	
Service ID	0x07	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Parameters (out)	`timeStamp`	- Time stamp containing the current time.
	`userData`	- User data of the time base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Returns a time value - Local Time derived from Global Time.	

5.5.2.4.5. StbM_GetCurrentTimeExtended

Purpose	Service to get current time.	
Synopsis	`Std_ReturnType` **`StbM_GetCurrentTimeExtended`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_TimeStampExtendedType * timeStamp , StbM_UserDataType * userData );`	
Service ID	0x08	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Parameters (out)	`timeStamp`	- extended Time stamp containing the current time.
	`userData`	- User data of the time base.
Return Value	the success/failure of the function call	

	E_OK	In case of successful call of the function.
	E_NOT_OK	In case of unsuccessful call of the function.
Description	Returns an extended time value - Local Time derived from Global Time.	

5.5.2.4.6. StbM_GetCurrentVirtualLocalTime

Purpose	Service to return the Virtual Local Time.	
Synopsis	`Std_ReturnType StbM_GetCurrentVirtualLocalTime ( StbM_SynchronizedTimeBaseType timeBaseId , StbM_VirtualLocalTimeType * localTimePtr );`	
Service ID	0x09	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	timeBaseId	- ID of the synchronized time-base.
Parameters (out)	localTimePtr	- Current Virtual Local Time value.
Return Value	the success/failure of the function call	
	E_OK	In case of successful call of the function.
	E_NOT_OK	In case of unsuccessful call of the function.
Description	Returns the Virtual Local Time of the referenced Time Base.	

5.5.2.4.7. StbM_GetMasterConfig

Purpose	Service to get the value of StbMIsSystemWideGlobalTimeMaster parameter of the time base.	
Synopsis	`Std_ReturnType StbM_GetMasterConfig ( StbM_SynchronizedTimeBaseType timeBaseId , StbM_MasterConfigType * masterConfig );`	
Service ID	0x1d	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	timeBaseId	- Time Base reference
Parameters (out)	masterConfig	- Indicates if system wide master functionality is supported

Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Indicates if the functionality for a system wide master (e.g. StbM_SetGlobalTime) for a given Time Base is available or not.	

5.5.2.4.8. StbM_GetOffset

Purpose	Service to get the value of a offset time base.	
Synopsis	`Std_ReturnType` **`StbM_GetOffset`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_TimeStampType * timeStamp , StbM_UserDataType * userData );`	
Service ID	0x0e	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Parameters (out)	`timeStamp`	- Time stamp containing the current offset time.
	`userData`	- The current user data.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Allows the Timebase Provider Modules to get the currentoffset time.	

5.5.2.4.9. StbM_GetOffsetTimeRecordHead

Purpose	Service to get Offset Header Time Record information.
Synopsis	`Std_ReturnType` **`StbM_GetOffsetTimeRecordHead`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_OffsetRecordTableHeadType * offsetRecordTableHead );`
Service ID	0x17
Sync/Async	Synchronous

Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- Time Base reference.
	`offsetRecordTableHead`	- Header of the table.
Return Value	the success/failure of the function call	
	`E_OK`	Table access done.
	`E_NOT_OK`	Table contains no data or access invalid.
Description	Accesses to the recorded snapshot data Header of the table belonging to the Offset Time Base.	

5.5.2.4.10. StbM_GetRateDeviation

Purpose	Service to return the value of the current rate deviation of a Time Base.	
Synopsis	`Std_ReturnType` **`StbM_GetRateDeviation`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_RateDeviationType * rateDeviation );`	
Service ID	0x11	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- Time Base reference.
Parameters (out)	`rateDeviation`	- Value of the current rate deviation of a Time Base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Returns value of the current rate deviation of a Time Base	

5.5.2.4.11. StbM_GetSyncTimeRecordHead

Purpose	Service to get Sync Header Time Record information.
Synopsis	`Std_ReturnType` **`StbM_GetSyncTimeRecordHead`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_SyncRecordTableHeadType * syncRecordTableHead );`
Service ID	0x16

Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- Time Base reference.
	`syncRecordTableHead`	- Header of the table.
Return Value	the success/failure of the function call	
	`E_OK`	Table access done.
	`E_NOT_OK`	Table contains no data or access invalid.
Description	Accesses to the recorded snapshot data Header of the table belonging to the Synchronized Time Base.	

5.5.2.4.12. StbM_GetTimeBaseStatus

Purpose	Service to return the default status of the time base.	
Synopsis	`Std_ReturnType` **`StbM_GetTimeBaseStatus`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_TimeBaseStatusType * syncTimeBaseStatus , StbM_TimeBaseStatusType * offsetTimeBaseStatus );`	
Service ID	0x14	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Parameters (out)	`syncTimeBaseStatus`	- Status of the Synchronized Time Base.
	`offsetTimeBaseStatus`	- Status of the Offset Time Base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Returns the detailed status of the Time Base. For Offset Time Bases the status of the Offset Time Base itself and the status of the underlying Synchronized Time Base is returned.	

5.5.2.4.13. StbM_GetTimeBaseUpdateCounter

Purpose	Service to get the counter value of the time base.

Synopsis	`uint8` **`StbM_GetTimeBaseUpdateCounter`** `( StbM_SynchronizedTime-BaseType timeBaseId );`	
Service ID	0x1b	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Return Value	Counter value belonging to the Time Base, that indicates a Time Base update to the TimeSync Modules.	
Description	Allows the TimeSync Modules to detect, whether a Time Base should be transmitted immediately in the subsequent [Bus]TSyn_MainFunction() cycle.	

5.5.2.4.14. StbM_GetTimeLeap

Purpose	Service to return the value of the current time leap of a Time Base.	
Synopsis	`Std_ReturnType` **`StbM_GetTimeLeap`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_TimeDiffType * timeJump );`	
Service ID	0x13	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- Time Base reference.
Parameters (out)	`timeJump`	- Time leap value.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Returns value of the current time leap of a Time Base	

5.5.2.4.15. StbM_GetVersionInfo

Purpose	Get module version information.
Synopsis	`void` **`StbM_GetVersionInfo`** `( Std_VersionInfoType * versioninfo );`
Service ID	0x05

Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (out)	`versioninfo`	Version information of the StbM module
Description	Returns the version information of the StbM module	

5.5.2.4.16. StbM_Init

Purpose	Initialize the StbM module.	
Synopsis	`void StbM_Init ( const StbM_ConfigType * ConfigPtr );`	
Service ID	0x00	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	`ConfigPtr`	Pointer to the selected configuration set.
Description	This function initializes the StbM module	

5.5.2.4.17. StbM_MainFunction

Purpose	StbM Mainfunction.
Synopsis	`void StbM_MainFunction ( void );`
Service ID	0x04
Sync/Async	Synchronous
Reentrancy	Non-Reentrant
Description	This function will be called cyclically by a task body provided by the BSW Scheduler. It will invoke the triggered customers and synchronize the referenced OS ScheduleTables. Timing: FIXED_CYCLIC.

5.5.2.4.18. StbM_SetBusProtocolParam

Purpose	Service to set bus specific parameters of a Time Master.

Synopsis	`Std_ReturnType` **`StbM_SetBusProtocolParam`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_ProtocolParamType * protocolParam );`	
Service ID	0x2A	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	`timeBaseId`	- Id of referenced Time Base
	`protocolParam`	- structure with Follow_Up information TLV parameters
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	This API is used to set bus specific parameters of a Time Master	

5.5.2.4.19. StbM_SetGlobalTime

Purpose	Service to set global time.	
Synopsis	`Std_ReturnType` **`StbM_SetGlobalTime`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_TimeStampType * timeStamp , const StbM_UserDataType * userData );`	
Service ID	0x0b	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
	`timeStamp`	- Time stamp containing the current time.
	`userData`	- User data of the time base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Allows the Customers to set the new global time that has to be valid for the system, which will be sent to the busses and modify HW registers behind the providers, if supported. This function will be used if a Time Master is present in this ECU.	

5.5.2.4.20. StbM_SetOffset

Purpose	Service to set the value of a offset time base.	
Synopsis	`Std_ReturnType` **`StbM_SetOffset`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_TimeStampType * timeStamp , const StbM_UserDataType * userData );`	
Service ID	0x0d	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
	`timeStamp`	- Time stamp containing the current offset time.
	`userData`	- User data of the time base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Allows the Customers and the Timebase Provider Modules to set the offset time that has to be valid for the system.	

5.5.2.4.21. StbM_SetRateCorrection

Purpose	Service to set the Rate Correction for a Time Base.	
Synopsis	`Std_ReturnType` **`StbM_SetRateCorrection`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_RateDeviationType rateDeviation );`	
Service ID	0x12	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- Time Base reference.
	`rateDeviation`	- Value of the applied rate deviation.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.

Description	Allows to set the rate of a Synchronized Time Base (being either a Pure Local Time Base or not).

5.5.2.4.22. StbM_SetUserData

Purpose	Service to set user specific data.	
Synopsis	`Std_ReturnType` **`StbM_SetUserData`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_UserDataType * userData );`	
Service ID	0x0c	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
	`userData`	- User data of the time base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Allows the Customers to set the new user data that has to be valid for the system, which will be sent to the busses.	

5.5.2.4.23. StbM_StartTimer

Purpose	Service to set a time value, which the Time Base value is compared against.	
Synopsis	`Std_ReturnType` **`StbM_StartTimer`** `( StbM_SynchronizedTimeBaseType timeBaseId , StbM_CustomerIdType customerId , const StbM_TimeStampType * expireTime );`	
Service ID	0x15	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- Time Base reference
	`customerId`	- Status of the Synchronized Time Base
	`expireTime`	- Time value relative to current Time Base value of the Notification Customer, when the Timer shall expire

Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Allows the Customers to set an expire time (through StartTimer C-S Interface), which the Time Base value is compared against.	

5.5.2.4.24. StbM_TriggerTimeTransmission

Purpose	Service to force the TimeSync Modules to transmit the current Time Base.	
Synopsis	`Std_ReturnType` **`StbM_TriggerTimeTransmission`** `( StbM_SynchronizedTimeBaseType timeBaseId );`	
Service ID	0x1c	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.
Return Value	the success/failure of the function call	
	`E_OK`	In case of successful call of the function.
	`E_NOT_OK`	In case of unsuccessful call of the function.
Description	Called by the [Upper Layer] to force the TimeSync Modules to transmit the current Time Base again due to an incremented timeBaseUpdateCounter[timeBaseId].	

5.5.2.4.25. StbM_UpdateGlobalTime

Purpose	Service to update global time.	
Synopsis	`Std_ReturnType` **`StbM_UpdateGlobalTime`** `( StbM_SynchronizedTimeBaseType timeBaseId , const StbM_TimeStampType * timeStamp , const StbM_UserDataType * userData );`	
Service ID	0x10	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	`timeBaseId`	- ID of the synchronized time-base.

	timeStamp	- Time stamp containing the current time.
	userData	- User data of the time base.
Return Value	the success/failure of the function call	
	E_OK	In case of successful call of the function.
	E_NOT_OK	In case of unsuccessful call of the function.
Description	Allows the Customers to set the Global Time that will be sent to the buses. This function will be used if a Time Master is present in this ECU. Using UpdateGlobalTime will not lead to an immediate transmission of the Global Time.	

5.5.3. Integration notes

5.5.3.1. Exclusive areas

This section describes the exclusive areas used by the StbM module.

5.5.3.1.1. SCHM_STBM_EXCLUSIVE_AREA_0

Protected data structures	All shared data that shall be protected from mutual access.
Recommended locking mechanism	The exclusive area shall be protected by a locking mechanism. The options for locking are described in the EB tresos AutoCore Generic documentation. Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details. This exclusive area shall be configured to a spin-lock if StbM is used in a multicore environment with cross-core calls.

5.5.3.1.2. SCHM_STBM_EXCLUSIVE_AREA_1

Protected data structures	StbM requires this exclusive area in order to provide enhanced precision of synchronizing of an OsScheduleTable.
Recommended locking mechanism	This exclusive area must always be assigned to an Os resource that is shared with a high priority task. The options for locking are described in the EB tresos AutoCore

	Generic documentation. Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details. This exclusive area is not available if Tiggered Costumers are not used by StbM.

5.5.3.1.3. SCHM_STBM_EXCLUSIVE_AREA_2

Protected data structures	StbM requires this exclusive area in order to encapsulate the Time Recording feature.
Recommended locking mechanism	This exclusive area shall be protected by a locking mechanism. The options for locking are described in the EB tresos AutoCore Generic documentation. Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details. This exclusive area shall be configured to a spin-lock if StbM is used in a multicore environment with cross-core calls. If the slave of a time base is on the same core(StbM_BusSetGlobalTime is called on the same core) as the StbM_MainFunction is, this exclusive area can be configured to another locking mechanism than spin-lock so resources are not wasted. This exclusive area is not available if Time Recording feature is not used by StbM.

5.5.3.2. Production errors

Production errors are not reported by the StbM module.

5.5.3.3. Memory mapping

General information about memory mapping is provided in the EB tresos AutoCore Generic documentation. Refer to the section Memory mapping and compiler abstraction in the Integration notes section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section

CODE
VAR_INIT_8
VAR_CLEARED_32
VAR_CLEARED_8
VAR_CLEARED_UNSPECIFIED
CONST_UNSPECIFIED
CONFIG_DATA_8
CONFIG_DATA_UNSPECIFIED
VAR_SHARED_TIME_DATA

5.5.3.4. Integration requirements

WARNING 

Integration requirements list is not exhaustive

The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user's guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

5.5.3.4.1. lim.StbM.EB_INTREQ_StbM_0001

Description	StbM needs module property files in adjacent modules. For auto detection and full control of adjacent modules, the StbM depends on the existence of a adequate module property file in this adjacent module. A module without module property file suffers the following limitations: - StbM can not control a master time domain of this BSW module. - StbM can not use a possible hardware timestamp capability of this BSW module. - The StbM configuration must contain a reserved time base for this BSW module. - Parameter StbMDevErrorDetect must be disabled (if at least one such module is used
Rationale	The mechanism of module property files allows the StbM to handle adjacent modules in a generic way. Adjacent modules often do not follow a common guideline for their module configuration and interfaces. In addition the StbM does not need to know how many and which modules shall be handled at configuration time.

5.5.3.4.2. lim.StbM.EB_INTREQ_StbM_0002

Description	Limitations on synchronization of schedule tables. To guarantee fault-free synchronization of schedule tables it is necessary to consider the following configuration constraints: Os counter interval - The Os counter interval (depends on OsCounterMaxAllowedValue and the tick time) must be greater than the MainFunction period of StbM. Schedule Table duration - The duration of the synchronized schedule table (depends on OsScheduleTableDuration and the tick time) must be smaller than one second. - In addition the value of OsScheduleTableDuration must not exceed 4294967296. - In addition one second must be an integer multiple of the duration of the synchronized schedule table.

5.5.3.4.3. lim.StbM.EB_INTREQ_StbM_0003

Description	Call to StbM_BusSetGlobalTime() is mandatory. The StbM requires that any adjacent bus specific time synchronization module (e.g. EthTSyn) notifies updates of the global time by using the API StbM_BusSetGlobalTime() when acting as time consumer (time slave) or time distributor (time gateway). - The StbM does not obtain the time by itself for any time base with an assigned slave time domain.
Rationale	This mechanism allows a common implementation for different configurations when acting as time distributor (time gateway). StbM only needs to supply the time in the context of StbM_BusSetGlobalTime(). Without this limitation StbM needs to supply the time in the context of: - StbM_BusSetGlobalTime() if the slave does not support hardware timestamping. - StbM_GetCurrentTime() if the master(s) does not

	support hardware timestamping. - StbM_MainFunction() if both the slave and the master(s) support hardware timestamping.

5.5.3.4.4. lim.StbM.EB_INTREQ_StbM_0004

Description	Limitation regarding the invocation of StbM_Init function. The invocation of StbM_Init, should be called after the EcuM module calls NvM_ReadAll.
Rationale	If the parameter StbMStoreTimebaseNonVolatile equals STORAGE_AT_SHUTDOWN and there exists a reference from a TimeBase to a NvMBlock, then whenever the TimeStamp will be changed, it will be stored in NvMBlock at SHUTDOWN phase and afterwards at INITIALIZATION phase (StbM_Init), the TimeStamp will be loaded from NvM module. If NvM_ReadAll is not called before StbM_Init, then StbM module will take as initial TimeStamp some garbage values.

5.5.3.4.5. lim.StbM.EB_INTREQ_StbM_0005

Description	Os counter with the highest resolution shall be used. The StbM provides the possibility to derive the current time from an Os counter if no hardware timestamp support is available. In this case it is recommended to select the Os counter via parameter StbMLocalTimeHardware with the highest resolution.
Rationale	The resolution of the Os counter determines the precision of the time base.

5.5.3.4.6. lim.StbM.EB_INTREQ_StbM_0006

Description	Exclusive access to Os counter used to derive local time. It shall be ensured that the Os counter used by StbM to derive the current time via parameter StbMLocalTimeHardware is not modified by any other basic software module or application.
Rationale	The StbM uses the Os counter to determine the current time if no hardware timestamp support is available. A leap in time of the respective Os counter results in a wrong time.

5.5.3.4.7. lim.StbM.EB_INTREQ_StbM_0007

Description	The exclusive areas which ensure that certain regions in the TSyn/StbM modules are not being preempted must not be configured to interrupt suspending/disabling in case OS counters are used as sources for the virtual local time.

Rationale	StbM cannot call GetCounterValue() with interrupts disabled/suspended (according to SWS_Os_00093)

5.5.3.4.8. lim.StbM.EB_INTREQ_StbM_0008

Description	StbM_MainFunction period must not exceed 3 sec.
Rationale	StbM needs to update the Main Tuple every 3 sec on the Main Function if no update happened before.

5.5.3.4.9. lim.StbM.EB_INTREQ_StbM_0009

Description	GptChannelMode of each Gpt Channel which is configured as Local hardware clock has to be configured to GPT_CH_MODE_CONTINUOUS.
Rationale	Gpt channel can be configured in "one-shot mode" or "continuous mode". Gpt Channel mode has to be configured to GPT_CH_MODE_CONTINUOUS, because when the timer reaches the target the timer shall continue running with the value "0" at next time tick, instead of stopping and maintaining its timer value unchanged.

5.5.3.4.10. lim.StbM.EB_INTREQ_StbM_0010

Description	GptChannelTickValueMax of each Gpt Channel which is configured as Local hardware clock has to be configured to the maximum value of UINT32.
Rationale	It determines the precision of the time base.

5.5.3.4.11. lim.StbM.EB_INTREQ_StbM_0011

Description	StbMClockFrequency and StbMClockPrescaler of each StbM TimeBase has to be different from zero and StbMClockFrequency/StbMClockPrescaler shall have a value equal with the value of the GptChannel Frequency from Gpt for a better precision.
Rationale	When calling Gpt_GetTimeElapsed() API, StbM obtains the time in GPT ticks. The time in Gpt ticks has to be converted into nanoseconds, using StbMClockFrequency/StbMClockPrescaler factor, that is why StbMClockFrequency and StbMClockPrescaler shall be configured to a value different from zero.

5.5.3.4.12. lim.StbM.EB_INTREQ_StbM_0012

Description	Limitation regarding the invocation of StbM_Init function. In case an StbM TimeBase uses Gpt as time source, the invocation of StbM_Init, should be called after the Gpt_Init. Please also make sure that Gpt is in NORMAL mode (not in SLEEP mode).
Rationale	If an StbM Time Base uses Gpt as time source for Virtual Local Time, the Gpt_Init function shall be called before StbM_Init, because Gpt module starts the hardware timers on the configured channels. The Gpt should be in NORMAL mode, because if Notification of customers is enabled, Gpt_EnableNotification() will be called.

5.5.3.4.13. lim.StbM.EB_INTREQ_StbM_0013

Description	Limitation regarding the usage of EthTSyn as a StbMLocalTimeHardware reference. An StbM time base can have as StbMLocalTimeHardware reference an EthTSyn Time Domain, only if EthTSyn module has EthTSynHardwareTimestampSupport parameter set to TRUE. This limitation applies to configuration parameter StbMLocalTimeHardware.
Rationale	The StbM module will not get the current value of the free running HW counter from the corresponding Ethernet Controller via EthIf_GetCurrentTime(), because the EthIf_GetCurrentTime() call will not be performed, if EthTSynHardwareTimestampSupport parameter is set to FALSE.

5.5.3.4.14. lim.StbM.EB_INTREQ_StbM_0014

Description	The GptChannelMode of the GptChannelConfiguration channel, configured as StbMGptTimerRef, shall be configured to GPT_CH_MODE_ONESHOT.
Rationale	Gpt channel can be configured in "one-shot mode" or "continuous mode". Gpt Channel mode has to be configured to GPT_CH_MODE_ONESHOT, because when the timer reaches the target the timer will expire and maintain its timer value unchanged, until the next Gpt_StartTimer() function call.

5.5.3.4.15. lim.StbM.EB_INTREQ_StbM_0015

Description	The GptChannelTickValueMax of the GptChannelConfiguration channel, configured as StbMGptTimerRef, shall be configured to a value greater or equal to the value of StbMTimerStartThreshold converted in ticks.
Rationale	For each Gpt channel a GptChannelTickValueMax can be configured. GptChannelTickValueMax parameter shall be configured to a value greater or equal to StbM-

	TimerStartThreshold converted into ticks, because the Gpt module shall be able to count the ticks until expiration time. Expiration time shall not exceed the maximum timer resolution.

5.5.3.4.16. lim.StbM.EB_INTREQ_StbM_0016

Description	The GptChannelTickFrequency of the GptChannelConfiguration channel, configured as StbMGptTimerRef, shall be configured to have a tick duration of one micro second.
Rationale	For each Gpt channel a GptChannelTickFrequency can be configured. GptChannelTickFrequency parameter shall be configured to have a tick duration of one micro second. See ECUC_StbM_00039 requirement.

5.5.3.4.17. lim.StbM.EB_INTREQ_StbM_0017

Description	If Gpt_ValueType is defined to be of type uint16, StbMTimerStartThreshold shall be configured to a maximum value of 0.065535 seconds.
Rationale	If Gpt_ValueType is defined to uint16, StbMTimerStartThreshold is limited to 0.065535 seconds, in order for Gpt to be able to count the maximum difference of time with which it can be called (65535 ticks).

5.5.3.4.18. lim.StbM.EB_INTREQ_StbM_0018

Description	The channel reference of the GptChannelConfiguration channel, configured as StbMGptTimerRef, shall NOT be used for ANY other purposes within the ECU configuration!
Rationale	StbM uses this Gpt channel to monitor notification customers which can not be interrupted by any means.

5.5.3.4.19. lim.StbM.EB_INTREQ_StbM_0019

Description	The length of the NvM block used to store the StbM_TimeStamps[].globalTime variable should be double checked. The Service Needs Wizard automatically generates it to 16, which includes padding.
Rationale	Depending on compiler options, padding might not be included, so the length will differ.

5.5.3.4.20. lim.StbM.EB_INTREQ_StbM_0020

Description	If Offset Correction is used, StbMOffsetCorrectionAdaptionInterval should have a value which is smaller than the synchronization period, but big enough to achieve the desired smoothies of the time from the application point of view. Example: if the slave is synchronized every second, StbMOffsetCorrectionAdaptionInterval could be 0.7s.
Rationale	Depending on compiler options, padding might not be included, so the length will differ.

5.5.3.4.21. lim.StbM.EB_INTREQ_StbM_0021

Description	For the different Synchronized/Offset StbM time base roles, the following rules must be followed: Master: StbMAllowSystemWideGlobalTimeMaster should be set to TRUE Gateway: StbMAllowSystemWideGlobalTimeMaster should be set to TRUE and StbMIsSystemWideGlobalTimeMaster should be FALSE Slave: StbMAllowSystemWideGlobalTimeMaster should be set to FALSE and StbMIsSystemWideGlobalTimeMaster should be FALSE For Pure StbM time bases, the following rules must be followed: Master: StbMAllowSystemWideGlobalTimeMaster should be set to TRUE
Rationale	To assure that the correct Rte C-S Interfaces are generated with respect to the time base role.

5.5.3.4.22. lim.StbM.EB_INTREQ_StbM_0022

Description	The SCHM_STBM_EXCLUSIVE_AREA_2 exclusive area which encapsulates the Time Recording feature and ensures that the Time Recording blocks are sent to the Rte without interruption, must be configured as follows: If the call to StbM_BusSetGlobalTime API (which is where the time recording blocks are recorded) is performed in the context of that EcucPartition on which the MF is mapped to, then this exclusive area must be configured to "interrupts disabled/suspended" mechanism. If the call to StbM_BusSetGlobalTime API (which is where the time recording blocks are recorded) is performed in the context of an EcucPartition which is not mapped to the MF, then this exclusive area must be configured to "spin lock" mechanism.
Rationale	Only one core could be locked, and not all of them.

5.5.3.4.23. lim.StbM.EB_INTREQ_StbM_0023

Description	If Multicore Distribution is supported in StbM, please make sure that the EcucPartition on which StbM_MainFunction() is mapped, is the first entry in StbMEcucPartitionRefList list.

Rationale	In StbM module there are APIs which are not multicore capable and they need to be available only on that EcucPartition on which the StbM _MainFunction is mapped to.

5.5.3.4.24. lim.StbM.EB_INTREQ_StbM_0024

Description	StbM provides Multicore Distribution in 2 cases: 1. Clients from different COREs want to Set/Get data from StbM, by calling Client-Server Interfaces generated by Rte 2. [bus]TSyn modules from different COREs want to Set/Get data from StbM, by calling StbM's APIs For the 1'st case, Multicore Distribution is supported in StbM, only if StbMRteUsage is also enabled. For the 2'nd case, Multicore Distribution is supported in StbM, even if StbMRteUsage is disabled. Note: All mentioned above is true as long as the exclusive areas are appropriately configured.
Rationale	Rationale applying to 1'st case: In StbM there are APIs which can be reached through Client-Server Interfaces generated by RTE. Those interfaces are generated only if StbMRteUsage is enabled. Note: All mentioned above is true as long as the exclusive areas are appropriately configured. Rationale applying to 2'nd case: In StbM there are also APIs which do not need to be reached through any Client-Server Interfaces generated by RTE. Those interfaces are provided by StbM and can be called directly by [bus]TSyn modules, even if StbMRteUsage is disabled. Note: All mentioned above is true as long as the exclusive areas are appropriately configured.

5.5.3.4.25. lim.StbM.EB_INTREQ_StbM_0025

Description	In case of Multicore Distribution the StbM_MainFunction shall be mapped, only on the first EcucPartition from the StbMEcucPartitionRefList list. The rest of StbM_MainFunc-

	tion Dummies (that correspond to the other partitions), shall be ignored during configuration.
Rationale	StbM does not support multiple MainFunction usage.

5.5.3.4.26. lim.StbM.EB_INTREQ_StbM_0026

Description	Make sure that there are suficient stack resources for Time Recording blocks (configured in StbMSyncTimeRecordTableBlockCount) and for Time Validation blocks (configured in StbMTimeValidationRecordTableBlockCount).
Rationale	StbM copies Time Recording / Time Validation blocks from a global variable to a local one (on the stack) and passes the copied blocks (from the local variable) to the Application, thought C-S Interface.