


SCOPE OF APPLICATION All Project/Engineering		SHT/SHTS 1 / 41
Responsibility: Classic AUTOSAR Team	AUTOSAR CanNm Manual	DOC. NO
AUTOSAR CanNm User Manual		

Document Change History			
Date (YYYY-MM-DD)	Ver.	Editor	Content
2020-12-04	1.0.0.0	SHKim	<ul style="list-style-type: none"> CanNm UM Initial Version
2022-03-09	1.0.1.0	ThuanVT	<ul style="list-style-type: none"> Update Scope of the Release Update Change Log
2022-05-09	1.0.2.0	ThuanVT	<ul style="list-style-type: none"> Update Scope of the Release Update Change Log Add new parameter CanNmForcePnAvailabilityConfEnabled
2022-06-20	1.0.2.1	SMKwon	<ul style="list-style-type: none"> Update Change Log
2022-08-16	1.1.0.0	SMKwon	<ul style="list-style-type: none"> Update Scope of the Release Update Change Log Add new parameter for NM State and Network Request Reason field. Change the values of CanNmPdulInfoLength and CanNmPdulInfoOffset Change the values of CanNmFilterMaskByteIndex and CanNmFilterMasByteValue
2022-10-07	1.1.1.0	NamNT1	<ul style="list-style-type: none"> Update Scope of the Release Update Change Log
2022-10-31	1.1.2.0	ThuanVT	<ul style="list-style-type: none"> Update Scope of the Release Update Change Log Update Error Messages
2022-12-16	1.1.3.0	NamNT1	<ul style="list-style-type: none"> Update Scope of the Release Update Change Log

사내한(Restricted)/제호 본 문서는 Hyundai Autoever 의 정보자산이므로 무단으로 전제 및 복제할 수 없으며, 이를 무단으로 이용하는 자는 법적 제재에 의해 제재를 받을 수 있습니다. <div> Edition Date: 2024/03/08 </div>	File Name CanNm_UM.pdf	Creation JH Yang	Check Hoimin Kim	Approval JINSO O Jang
		2024/03/08	2024/03/08	2024/03/08

2023-01-31	1.1.4.0	NamNT1	<ul style="list-style-type: none"> •Update Scope of the Release •Update Change Log •Delete redundant validation (ERR31023)
2023-04-28	1.1.5.0	NamNT1	<ul style="list-style-type: none"> •Update Scope of the Release •Update Change Log
2023-06-16	1.1.6.0	HieuTM8	<ul style="list-style-type: none"> •Update Scope of the Release •Update Change Log •CanNmChannelConfig Settings
2024-02-19	1.2.0.0	JHKim	<ul style="list-style-type: none"> •Update Scope of the Release •Update Change Log •Add Deviations •Add new parameter •Update Error message
2024-03-08	1.2.0.0_HF1	JH Yang	<ul style="list-style-type: none"> •Update Error Messages

Table of Contents

1 Overview	5
2 Reference	5
3 AUTOSAR System	6
3.1 CanNm Module	6
4 Limitations and Deviations	7
4.1 Limitations	7
4.2 Deviations	8
5 Configuration Guide	8
5.1 CanNmGlobalConfig Settings	8
5.2 CanNmChannelConfig Settings	11
5.3 CanNmRxPdu Settings	17
5.4 CanNmTxPdu Settings	18
5.5 CanNmUserDataTxPdu Settings	18
5.6 CanNmPnInfo Settings	18
5.7 CanNmPnFilterMaskByte Settings	19
6 Application Programming Interface (API)	19
6.1 Type Definitions	19
6.2 Macro Constants	19
6.3 Functions	19
6.3.1 Initialization	20
6.3.2 Network Control	20
6.3.3 Communication Control	22
6.3.4 Information Services	23
6.3.5 Scheduled functions	32
7 Generator	32
7.1 Generator Option	32

7.2 Generator Message	33
7.2.1 Error Messages.....	33
7.2.2 Warning Messages	41
7.2.3 Information Messages	41
8 SWP Error Code	41
9 Appendix	41

1 Overview

It is written based on AUTOSAR standard SRS / SWS. If a more detailed functional explanation is needed when using the module, see the Reference Manual. The interpretation of setting related category is as follows:

- Changeable (C): Items that can be set by the user
- Fixed (F): Items that cannot be changed by the user.
- Not Supported (N): Deprecated item

2 Reference

Sl. No.	Title	Version
1	AUTOSAR_SWS_CanNetworkManagement.pdf	4.4.0

3 AUTOSAR System

3.1 CanNm Module

The AUTOSAR CAN Network Management is a hardware-independent protocol that can only be used on CAN. Its main purpose is to coordinate the transition between normal operation and bus-sleep mode of the network. In addition to the core functionality, configurable features are provided e.g. to implement a service to detect all present nodes or to detect if all other nodes are ready to sleep.

The CAN Network Management (CanNm) function provides an adaptation between Network Management Interface (NmIf) and CAN Interface (CanIf) module.

The CanNm module performs the following operations to synchronize controllers entering SLEEP on the same network based on the Autosar CanNm specification:

- Network Request and Release request processing of controller.
- Remote or Local Wake-up request processing.
- Network Configuration function provided.
- Perform procedures for CAN Bus Sleep synchronization.

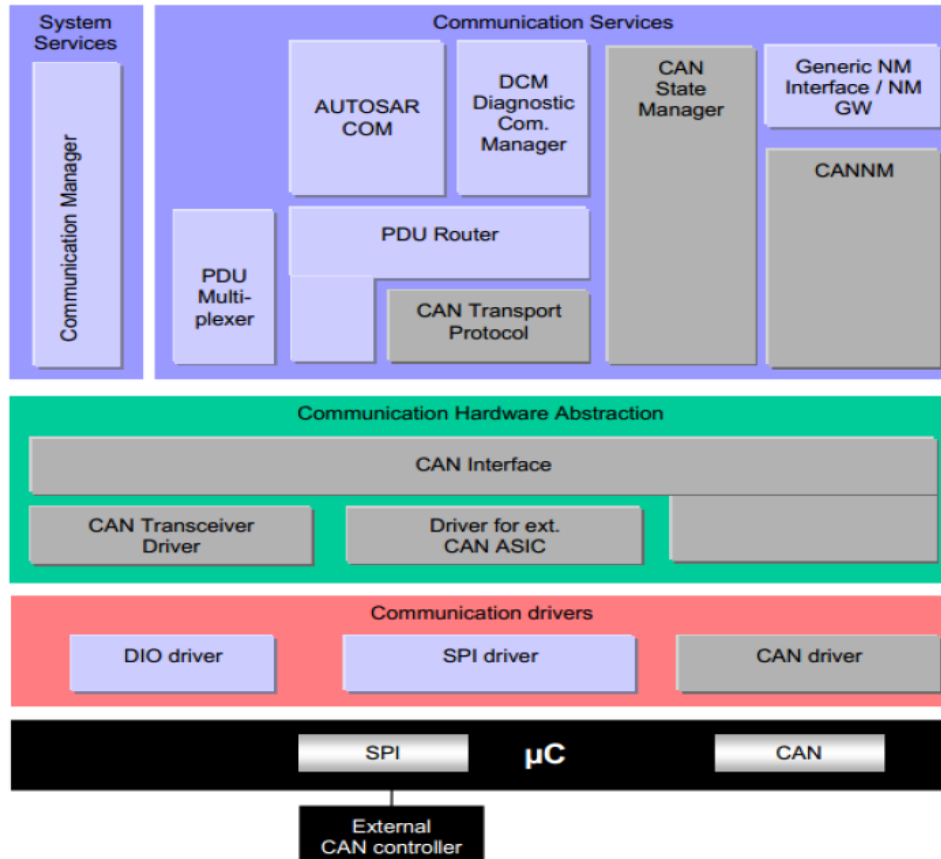


Figure 1

4 Limitations and Deviations

4.1 Limitations

- One channel of CanNm is associated with only one network management cluster in one network. One network management cluster can have only one channel of CanNm in one node.
- One channel of CanNm is associated with only one network within the same ECU.
- CanNm is only applicable for CAN systems.

4.2 Deviations

- Non-sent Nm messages from Normal Operation State when using ES Passive Node
Although it is not described in the AUTOSAR CanNm specification, it is a passiveNode feature within ES95480-03E that supports sending NmMessages in the corresponding controller Normal Operation State when the CanNmGlobalConfig/CanNmPassiveNodeEnabled setting is true.

5 Configuration Guide

5.1 CanNmGlobalConfig Settings

Parameter Name	Value	Category
CanNmBusLoadReductionEnabled	True/False	C
CanNmBusSynchronizationEnabled	True/False	C
CanNmComControlEnabled	True/False	C
CanNmComUserDataSupport	True/False	C
CanNmCoordinatorSyncSupport	True/False	C
CanNmDevErrorDetect	True/False	C
CanNmGlobalPnSupport	True/False	C
CanNmImmediateRestartEnabled	True/False	C
CanNmImmediateTxconfEnabled	True/False	C
CanNmMainFunctionPeriod	0 ... INF	C
CanNmPassiveModeEnabled	True/False	C
CanNmPduRxIndicationEnabled	True/False	C
CanNmPnEiraCalcEnabled	True/False	C
CanNmPnResetTime	0.001 ... 65535	C
CanNmRemoteSleepIndEnabled	True/False	C
CanNmStateChangeIndEnabled	True/False	C
CanNmUserDataEnabled	True/False	C
CanNmVersionInfoApi	True/False	C
CanNmPnEiraRxNSduRef	Reference to [Pdu]	C
CanNmPassiveNodeEnabled	True/False	C

1) CanNmBusLoadReductionEnabled:

Switch for enabling busload reduction support.

Dependency: If CanNmPassiveModeEnabled or CanNmGlobalPnSupport is true, then CanNmBusLoad-ReductionEnabled is False.

2) **CanNmBusSynchronizationEnabled:**

Switch for enabling bus synchronization support.

Dependency: If CanNmPassiveModeEnabled is False, then CanNmComControlEnabled equals NmBus-SynchronizationEnabled else False.

3) **CanNmComControlEnabled:**

Switch for enabling the Communication Control support.

Dependency: If CanNmPassiveModeEnabled is False, then CanNmComControlEnabled equals NmCom-ControlEnabled else False.

4) **CanNmComUserDataSupport:**

Switch for enabling the Tx path of Com User Data.

Dependency: If CanNmPassiveModeEnabled is True then CanNmComUserDataSupport is False.

5) **CanNmCoordinatorSyncSupport:**

Enables/disables the coordinator synchronization support.

Dependency: CanNmCoordinatorSyncSupport has to be false if CanNmPasiveModeEnabled is true.

6) **CanNmDevErrorDetect:**

Switches the development error detection and notification on or off.

- true: detection and notification are enabled.
- false: detection and notification are disabled.

7) **CanNmGlobalPnSupport:**

Switch for enabling partial networking support globally.

8) **CanNmImmediateRestartEnabled:**

Switch for enabling the immediate transmission of an NM PDU upon bus-communication request in Prepare-Bus-Sleep mode.

Dependency: Only valid if CanNmPasiveModeEnabled is false.

9) **CanNmImmediateTxconfEnabled:**

Enable/disable the immediate tx confirmation.

Dependency: Only valid if CanNmPasiveModeEnabled is false.

10) **CanNmMainFunctionPeriod:**

Call cycle in seconds of CanNm_MainFunction

11) **CanNmPassiveModeEnabled:**

Switch for enabling support of the Passive Mode.

12) **CanNmPduRxIndicationEnabled:**

Switch for enabling the PDU Rx Indication.

Dependency: It is the same value as NmPduRxIndicationEnabled.

13) **CanNmPnEiraCalcEnabled:**

Specifies if CanNm calculates the PN request information for internal and external requests.
(EIRA)

Dependency: Only valid if CanNmGlobalPnSupport is true.

14) **CanNmPnResetTime:**

Specifies the runtime of the reset timer in seconds. This reset time is valid for the reset of PN requests in the EIRA and the ERA.

Dependency:

- CanNmMsgCycleTime < CanNmPnResetTime < CanNmTimeoutTime
- Only valid if CanNmGlobalPnSupport is true.

15) **CanNmRemoteSleepIndEnabled:**

Switch for enabling remote sleep indication support. This feature is required for gateway nodes only.

Dependency: It is AND bitwise between CanNmPassiveModeEnabled and NmRemoteSleepIndEnabled.

16) **CanNmStateChangeIndEnabled:**

Switch for enabling the CAN NM state change notification.

Dependency: It is the same value as NmStateChangeIndEnabled.

17) **CanNmUserDataEnabled:**

Switch for enabling user data support.

Dependency: It is the same value as NmUserDataEnabled.

18) **CanNmVersionInfoApi:**

Switch for enabling version info API support.

19) CanNmPnEiraRxNSduRef:

Reference to a Pdu in the COM-Stack.

Dependency: Only valid if CanNmPnEiraCalcEnabled is true.

20) CanNmPassiveNodeEnabled:

Switch for enabling support of the Passive Node.

Dependency:

- If CanNmPassiveNodeEnabled is true, then CanNmBusSynchronization, CanNmCoordinatorSyncSupport, CanNmRemoteSleepIndEnabled, CanNmPassiveModeEnabled should be false.
- If this parameter setting is set to Unset, a generation error is set to occur.

5.2 CanNmChannelConfig Settings

Parameter Name	Value	Category
CanNmActiveWakeupBitEnabled	True/False	C
CanNmAllNmMessagesKeepAwake	True/False	C
CanNmBusLoadReductionActive	True/False	C
CanNmCarWakeUpBitPosition	True/False	C
CanNmCarWakeUpBytePosition	0 ... 7	C
CanNmCarWakeUpFilterEnabled	True/False	C
CanNmCarWakeUpFilterNodeId	0...255	C
CanNmCarWakeUpRxEnabled	True/False	C
CanNmImmediateNmCycleTime	0.001...65535	C
CanNmImmediateNmTransmissions	0 ... 255	C
CanNmMsgCycleOffset	0.001...65535	C
CanNmMsgCycleTime	0.001...65535	C
CanNmMsgReducedTime	0.001...65535	C
CanNmMsgTimeoutTime	0.001...65535	C
CanNmNodeDetectionEnabled	True/False	C
CanNmNodeId	0 ... 255	C
CanNmNodeIdEnabled	True/False	C
CanNmPduCbvPosition	CANNM_PDU_BYTE_0, CANNM_PDU_BYTE_1,	C

	CANNM_PDU_OFF.	
CanNmPduNidPosition	CANNM_PDU_BYTE_0, CANNM_PDU_BYTE_1, CANNM_PDU_OFF.	C
CanNmPnEnabled	True/False	C
CanNmPnEraCalcEnabled	True/False	C
CanNmPnHandleMultipleNetworkRequests	True/False	C
CanNmRemoteSleepIndTime	0.001...65535	C
CanNmRepeatMessageTime	0...65535	C
CanNmRepeatMsgIndEnabled	True/False	C
CanNmTimeoutTime	0.002...65535	C
CanNmWaitBusSleepTime	0.001...65535	C
CanNmComMNetworkHandleRef	Symbolic name reference to [ComMChannel]	C
CanNmPnEraRxNSduRef	Reference to [Pdu]	C
CanNmForcePnAvailabilityConfEnabled	True/False	C
CanNmPduNetworkRequestReasonPosition	CANNM_PDU_BYTE_2, CANNM_PDU_OFF	C
CanNmPduNmStatePosition	CANNM_PDU_BYTE_3, CANNM_PDU_OFF	C
CanNmUserDataInitValue	0 ... 255	C

1) CanNmActiveWakeupBitEnabled:

Enables/Disables the handling of the Active Wakeup Bit in the CanNm module.
Dependency: Only valid if CanNmPassiveModeEnabled is false.

2) CanNmAllNmMessagesKeepAwake:

Specifies if CanNm drops irrelevant NM PDUs.

- false: Only NM PDUs with a PNI bit are true and containing a PN request for this ECU triggers the standard RX indication handling.
- true: Every NM PDU triggers the standard RX indication handling.

Dependency: Only valid if CanNmPnEraCalcEnabled or CanNmPnEraCalcEnabled is true.

3) CanNmBusLoadReductionActive:

This parameter defines if bus load reduction for the respective NM channel is active or not.
Dependency: CanNmBusLoadReductionActive is false if CanNmBusLoadReductionEnabled is false.

4) CanNmCarWakeUpBitPosition:

Specifies the bit position of the CWU within the NM PDU.

Dependency: Only valid if CanNmCarWakeUpRxEnabled is true.

5) CanNmCarWakeUpBytePosition:

Specifies the Byte position of the CWU within the NM PDU.

Dependency:

- Only valid if CanNmCarWakeUpRxEnabled is true.
- Its value should be greater than or equal to a number of enable system bytes (CBV, NID).

6) CanNmCarWakeUpFilterEnabled:

If CWU filtering is supported, only the CWU bit within the NM PDU with source node identifier CanNmCarWakeUpFilterNodeId is considered as a CWU request.

- False: CWU filtering is not supported
- True: CWU filtering is supported

Dependency: Only valid if CanNmCarWakeUpRxEnabled is true.

7) CanNmCarWakeUpFilterNodeId:

It is used to verify the received Nm PDUs are processed CarWakeUp feature or not.

Dependency: Only valid if CanNmCarWakeUpFilterEnabled is true.

8) CanNmCarWakeUpRxEnabled:

Enables or disables support of CarWakeUp bit evaluation in received NM PDUs.

FALSE: CarWakeUp is not supported.

TRUE: CarWakeUp supported.

9) CanNmImmediateNmCycleTime:

Defines the immediate NM PDU cycle time in seconds which is used for CanNmImmediateNm-Transmissions NM PDU transmissions.

Dependency: Only valid if CanNmImmediateNmTransmissions are greater one.

10) CanNmImmediateNmTransmissions:

Defines the number of immediate NM PDUs which shall be transmitted. If the value is zero no immediate NM PDUs are transmitted.

Dependency:

- If CanNmImmediateRestartEnabled is True then CanNmImmediateNmTransmissions is 0
- If CanNmPnHandleMultipleNetworkRequests is True then CanNmImmediateNmTransmissions is greater than 0.

11) CanNmMsgCycleOffset:

Time offset in the periodic transmission node. It determines the start delay of the transmission. Specified in seconds.

Dependency:

- It shall be less than CanNmMsgCycleTime.
- Only valid if CanNmPassiveModeEnabled is False.

12) CanNmMsgCycleTime:

It determines the periodic rate in the "periodic transmission mode with busload reduction" and is the basis for transmitting scheduling in the "periodic transmission mode without bus load reduction".

Dependency:

- Only valid if CanNmPassiveModeEnabled is False.

13) CanNmMsgReducedTime:

Node specific bus cycle time in the periodic transmission mode with busload reduction. Specified in seconds.

Dependency:

- The value is equal to or greater than $0,5 * \text{CanNmMsgCycleTime}$ and less than CanNmMsgCycleTime.
- Only valid if CanNmBusLoadReductionEnabled is True and CanNmBusLoadReductionActive is True and CanNmPassiveModeEnabled is False.

14) CanNmMsgTimeoutTime:

This timeout is defined then CanNm monitors that an NM-PDU is transmitted successfully within this Transmission Timeout Time and provides an error notification

Dependency:

- The value is less than CanNmMsgCycleTime.
- Only valid if CanNmPassiveModeEnabled and CanNmImmediateTxConfEnabled are set to FALSE and CanNmPnEnabled is set to TRUE.

15) CanNmNodeDetectionEnabled:

Precompile time switch to enable the node detection feature.

Dependency:

- Only valid if CanNmNodeIdEnabled is set to TRUE.
- If CanNmPassiveModeEnabled is True then CanNmNodeDetection will be False.

16) CanNmNodeId:

Node identifier of the local node.

Dependency:

- Only valid if CanNmNodeIdEnabled is True.

17) CanNmNodeIdEnabled:

Pre-processor switch for enabling the source node identifier.

It is always True because NmNodeIdEnable is always True.

18) CanNmPduCbvPosition:

Defines the position of the control bit vector within the NM PDU.

Dependency:

- The value is not the same as CanNmPduNidPosition.
- If CanNmNodeDetectionEnabled is True then CanNmPduCbvPosition won't be CANNM_PDU_OFF.
- If CanNmPduNidPosition is CANNM_PDU_OFF and CanNmPduCbvPosition is not CANNM_PDU_OFF then CanNmPduCbvPosition shall be CANNM_PDU_BYTE_0.

19) CanNmPduNidPosition:

Defines the position of the source node identifier within the NM PDU.

Dependency:

- If CanNmNodeIdEnabled is True then CanNmPduNidPosition is not CANNM_PDU_OFF.
- The value is not the same with CanNmPduCbvPosition.
- If CanNmPduCbvPosition is CANNM_PDU_OFF and CanNmPduNidPosition is not CANNM_PDU_OFF then CanNmPduNidPosition shall be CANNM_PDU_BYTE_0.

20) CanNmPnEnabled:

It is used to enable or disable the support of partial networking.

Dependency:

- Only valid if CanNmGlobalPnSupport is True.

21) CanNmPnEraCalcEnabled:

Specifies if CanNm calculates the PN request information for external requests. (ERA)

Dependency:

- Only valid if CanNmGlobalPnSupport is True.

22) CanNmPnHandleMultipleNetworkRequests:

Specifies if CanNm performs an additional transition from Network Mode to Repeat Message State.

Dependency:

- Only valid if CanNmGlobalPnSupport is True.

23) CanNmRemoteSleepIndTime:

It defines the time in seconds how long it shall take to recognize that all other nodes are ready to sleep.

Dependency:

- CanNmRemoteSleepIndTime is equal to or greater than CanNmMsgCycleTime.
- CanNmRemoteSleepIndTime is only required if CanNmRemoteSleepIndEnabled is True.

24) CanNmRepeatMessageTime:

It defines the time in seconds how long the NM shall stay in the Repeat Message State.

Dependency:

- CanNmRepeatMessageTime is multiple of CanNmMsgCycleTime.
- It is greater than multiple of CanNmImmediateNmTransmissions and CanNmImmediateNmCycleTime.

25) CanNmRepeatMsgIndEnabled:

Enable or disable to call Nm_RepeatMessageIndication to notify reception of Repeat Message Request bit.

Dependency:

- If CanNmPassiveModeEnabled is False, it is equal to NmRepeatMsgIndEnabled and it is False if not.

26) CanNmTimeoutTime:

It is time for managing networks in the system.

Dependency:

- All nodes in the cluster shall be the same CanNmTimeoutTime and it shall be greater than CanNmMsgCycleTime.

27) CanNmWaitBusSleepTime:

It denotes the time in seconds how long the NM shall stay in the Prepare Bus-Sleep Mode before transitioning into Bus-Sleep Mode.

Dependency:

- All nodes in the cluster shall be the same CanNmWaitBusSleepTime.

28) CanNmComMNetworkHandleRef:

It is used to get ComMChannelId which the channel belongs to and pass arguments when calling API of Nm to report statistics of networks.

29) CanNmPnEraRxNSduRef:

It is used to get PduId in PduR to pass when calling PduR_CanNmRxIndication if the ERA feature is enabled.

30) CanNmForcePnAvailabilityConfEnabled

This parameter enables or disables confirmation of PN availability without PN transceiver.

31) CanNmPduNetworkRequestReasonPosition:

The value of the parameter represents the location of the Network Request Reason in the NM PDU (CANNM_PDU_BYTE_2 means 3rd byte of NM PDU, CANNM_PDU_BYTE_OFF means Network Request Reason field is not part of the NM PDU)

32) CanNmPduNmStatePosition:

The value of the parameter represents the location of the NM State in the NM PDU (CANNM_PDU_BYTE_3 means 4th byte of NM PDU, CANNM_PDU_BYTE_OFF means NM State field is not part of the NM PDU)

33) CanNmUserDataInitValue:

The value of the parameter represents the user data that will be initialed for each byte of the CanNm Tx channel.

5.3 CanNmRxPdu Settings

Parameter Name	Value	Category
CanNmRxPduId	0 ... 65535	C
CanNmRxPduRef	Reference to [Pdu]	C

1) CanNmRxPduId:

This parameter defines the Rx PDU ID of the CanIf L-PDU range that is associated with this CanNm channel.

2) CanNmRxPduRef:

Reference to the global PDU that is used by this CanNm channel.

5.4 CanNmTxPdu Settings

Parameter Name	Value	Category
CanNmTxConfirmationPduId	0 ... 65535	C
CanNmTxPduRef	Reference to [Pdu]	C

1) CanNmTxConfirmationPduId:

Handle Id to be used by the Lower Layer to confirm the transmission of the CanNmTxPdu to the LowerLayer.

2) CanNmTxPduRef:

The reference to the common PDU structure.

5.5 CanNmUserDataTxPdu Settings

Parameter Name	Value	Category
CanNmTxUserDataPduId	0 ... 65535	C
CanNmTxUserDataPduRef	Reference to [Pdu]	C

1) CanNmTxUserDataPduId:

This parameter defines the Handle ID of the NM User Data I-PDU.

2) CanNmTxUserDataPduRef:

Reference to the NM User Data I-PDU in the global PDU collection.

5.6 CanNmPnInfo Settings

Parameter Name	Value	Category
CanNmPnInfoLength	1 ... 63	C
CanNmPnInfoOffset	1 ... 63	C

1) CanNmPnInfoLength:

Specifies the length of the PN request information in the NM PDU.

Dependency:

- Only valid if CanNmGlobalPnSupport is True.

2) CanNmPnInfoOffset:

Specifies the offset of the PN request information in the NM PDU.

Dependency:

- Only valid if CanNmGlobalPnSupport is True.

5.7 CanNmPnFilterMaskByte Settings

Parameter Name	Value	Category
CanNmPnFilterMaskByteIndex	0 ... 62	C
CanNmPnFilterMaskByteValue	0 ... 255	C

1) CanNmPnFilterMaskByteIndex:

Index of the filter mask byte. Specifies the position within the filter mask byte array.

Dependency:

- Only valid if CanNmGlobalPnSupport is True and CanNmPnFilterMaskByteIndex is less than CanNmPnInfoLength.

2) CanNmPnFilterMaskByteValue:

Parameter to configure the filter mask byte.

Dependency:

- Only valid if CanNmGlobalPnSupport is True.

6 Application Programming Interface (API)

6.1 Type Definitions

<In this chapter all types of module are listed>

6.2 Macro Constants

None

6.3 Functions

<Describes all functionalities of modules, group by feature or listed following SWS chapter 8>

6.3.1 Initialization

Function Name	CanNm_Init	
Syntax	<i>void CanNm_Init (</i> <i>const CanNm_ConfigType* cannmConfigPtr)</i>	
Service ID [Hex]	0x00	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	cannmConfigPtr	Pointer to a selected configuration structure
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	Initialize the CanNm module. This function is used by BSW.	
Available via	CanIf.h	

Function Name	CanNm_Delnit	
Syntax	<i>void CanNm_Delnit (void)</i>	
Service ID [Hex]	0x10	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	None	
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	De-initializes the CanNm module. This function is used by BSW.	
Available via	CanNm.h	

6.3.2 Network Control

Function Name	CanNm_PassiveStartUp
----------------------	----------------------

Syntax	<i>Std_ReturnType CanNm_PassiveStartUp (</i> <i>NetworkHandleType nmChannelHandle)</i>	
Service ID [Hex]	0x01	
Sync/Async	Asynchronous	
Reentrancy	Reentrant (but not for the same NM-Channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Passive startup of network management has failed
Description	<p>Passive startup of the AUTOSAR CAN NM. It triggers the transition from Bus-Sleep Mode or Prepares Bus Sleep Mode to the Network Mode in Repeat Message State.</p> <p>Caveats: CanNm is initialized correctly. This function is used by user.</p>	
Available via	CanNm.h	

Function Name	CanNm_NetworkRequest	
Syntax	<i>Std_ReturnType CanNm_NetworkRequest (</i> <i>NetworkHandleType nmChannelHandle)</i>	
Service ID [Hex]	0x02	
Sync/Async	Asynchronous	
Reentrancy	Reentrant (but not for the same NM-Channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Requesting of network has failed

Description	Request the network, since ECU needs to communicate on the bus. This function is used by user. But it needs configuration. (It cannot be called directly by user)
Available via	CanNm.h

Function Name	CanNm_NetworkRelease	
Syntax	<i>Std_ReturnType CanNm_NetworkRelease (</i> <i>NetworkHandleType nmChannelHandle)</i>	
Service ID [Hex]	0x03	
Sync/Async	Asynchronous	
Reentrancy	Reentrant (but not for the same NM-Channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Releasing of network has failed
Description	Release the network, since ECU doesn't have to communicate on the bus. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

6.3.3 Communication Control

Function Name	CanNm_DisableCommunication	
Syntax	<i>Std_ReturnType CanNm_DisableCommunication (</i> <i>NetworkHandleType nmChannelHandle)</i>	
Service ID [Hex]	0x0c	
Sync/Async	Asynchronous	
Reentrancy	Reentrant (but not for the same NM-Channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	

Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Disabling of NM PDU transmission ability has failed
Description	Disable the NM PDU transmission ability due to a ISO14229 Communication Control (28hex) service This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_EnableCommunication	
Syntax	Std_ReturnType CanNm_EnableCommunication (NetworkHandleType nmChannelHandle)	
Service ID [Hex]	0x0d	
Sync/Async	Asynchronous	
Reentrancy	Reentrant (but not for the same NM-Channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Enabling of NM PDU transmission ability has failed
Description	Enable the NM PDU transmission ability due to a ISO14229 Communication Control (28hex) service This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

6.3.4 Information Services

Function Name	CanNm_SetUserData
Syntax	Std_ReturnType CanNm_SetUserData (

	<i>NetworkHandleType nmChannelHandle, const uint8* nmUserDataPtr</i>	
Service ID [Hex]	0x04	
Sync/Async	Synchronous	
Reentrancy	Reentrant (but not for the same NM-Channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
	nmUserDataPtr	Pointer where the user data for the next transmitted NM PDU shall be copied from
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error
		E_NOT_OK: Setting of user data has failed
Description	Set user data for NM PDUs transmitted next on the bus. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_GetUserData	
Syntax	<i>Std_ReturnType CanNm_GetUserData (</i> <i>NetworkHandleType nmChannelHandle, uint8* nmUserDataPtr)</i>	
Service ID [Hex]	0x05	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	nmUserDataPtr	Pointer where user data out of the most recently received NM PDU shall be copied to
Return Value	Std_ReturnType	E_OK: No error
		E_NOT_OK: Getting of user data has failed
Description	Get user data out of the most recently received NM PDU. This function is used by user.	

	But it needs configuration. (It cannot be called directly by user)
Available via	CanNm.h

Function Name	CanNm_GetNodeIdentifier	
Syntax	<i>Std_ReturnType CanNm_GetNodeIdentifier (</i> <i>NetworkHandleType nmChannelHandle, uint8* nmNodeIDPtr)</i>	
Service ID [Hex]	0x06	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	nmNodeIDPtr	Pointer where node identifier out of the most recently received NM PDU shall be copied to
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Getting of the node identifier out of the most recently received NM PDU has failed or is not configured for this network handle.
Description	Get node identifier out of the most recently received NM PDU. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_GetLocalNodeIdentifier	
Syntax	<i>Std_ReturnType CanNm_GetLocalNodeIdentifier (</i> <i>NetworkHandleType nmChannelHandle, uint8* nmNodeIDPtr)</i>	
Service ID [Hex]	0x07	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	

Parameters (Out)	nmNodeIdPtr	Pointer where node identifier of the local node shall be copied to
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Getting of the node identifier of the local node has failed or is not configured for this network handle.
Description	Get node identifier configured for the local node. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_RepeatMessageRequest	
Syntax	<i>Std_ReturnType CanNm_RepeatMessageRequest (</i> <i>NetworkHandleType nmChannelHandle)</i>	
Service ID [Hex]	0x08	
Sync/Async	Asynchronous	
Reentrancy	Reentrant (but not for the same NM-channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Setting of Repeat Message Request Bit has failed or is not configured for this network handle.
Description	Set Repeat Message Request Bit for NM PDUs transmitted next on the bus. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_GetPduData	
Syntax	<i>Std_ReturnType CanNm_GetPduData (</i>	

	<i>NetworkHandleType nmChannelHandle, uint8* nmPduDataPtr)</i>	
Service ID [Hex]	0x0a	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	nmPduDataPtr	Pointer where NM PDU shall be copied to
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Getting of NM PDU Data has failed or is not configured for this network handle.
Description	Get the whole PDU data out of the most recently received NM PDU. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_GetState	
Syntax	<i>Std_ReturnType CanNm_GetState (</i> <i>NetworkHandleType nmChannelHandle, Nm_StateType*</i> <i>nmStatePtr, Nm_ModeType* nmModePtr)</i>	
Service ID [Hex]	0x0b	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	nmStatePtr	Pointer where state of the network management shall be copied to
	nmModePtr	Pointer where the mode of the network management shall be copied to
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Getting of NM state has failed.
Description	Get the whole PDU data out of the most recently received NM PDU.	

	This function is used by user.
Available via	CanNm.h

Function Name	CanNm_GetVersionInfo	
Syntax	<i>void CanNm_GetVersionInfo (</i> <i>Std_VersionInfoType* versioninfo)</i>	
Service ID [Hex]	0xf1	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	None	
Parameters (Inout)	None	
Parameters (Out)	versioninfo	Pointer to where to store the version information of this module
Return Value	None	
Description	This service returns the version information of this module. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_RequestBusSynchronization	
Syntax	<i>Std_ReturnType CanNm_RequestBusSynchronization</i> <i>NetworkHandleType nmChannelHandle)</i>	
Service ID [Hex]	0xc0	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Requesting of bus synchronization has failed
Description	Request bus synchronization.	

	This function is used by user. But it needs configuration. (It cannot be called directly by user)
Available via	CanNm.h

Function Name	CanNm_CheckRemoteSleepIndication	
Syntax	<i>Std_ReturnType CanNm_CheckRemoteSleepIndication(</i> <i>NetworkHandleType nmChannelHandle, boolean*</i> <i>nmRemoteSleepIndPtr)</i>	
Service ID [Hex]	0xd0	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	nmRemoteSleepIndPtr	Pointer where check result of remote sleep indication shall be copied to
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Checking of remote sleep indication bits has failed
Description	Check if remote sleep indication takes place or not. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_SetSleepReadyBit	
Syntax	<i>Std_ReturnType CanNm_SetSleepReadyBit (</i> <i>NetworkHandleType nmChannelHandle, boolean nmSleepReadyBit)</i>	
Service ID [Hex]	0x17	
Sync/Async	Synchronous	
Reentrancy	Reentrant (but not for the same NM-channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
	nmSleepReadyBit	Value written to ReadySleep Bit in CBV
Parameters (Inout)	None	

Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: No error E_NOT_OK: Writing of remote sleep indication bit has failed
Description	Set the NM Coordinator Sleep Ready bit in the Control Bit Vector This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_TxConfirmation	
Syntax	void CanNm_TxConfirmation (PduIdType TxPduId, Std_ReturnType result)	
Service ID [Hex]	0x40	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (In)	TxPduId	ID of the PDU that has been transmitted.
	result	E_OK: The PDU was transmitted. E_NOT_OK: Transmission of the PDU failed.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	The lower layer communication interface module confirms the transmission of a PDU, or the failure to transmit a PDU. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

Function Name	CanNm_RxIndication	
Syntax	Void CanNm_RxIndication (PduIdType RxPduId, const PduInfoType* PduInfoPtr)	

Service ID [Hex]	0x42	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (In)	RxPduId	ID of the received PDU.
	PduInfoPtr	Contains the length (SduLength) of the received PDU, a pointer to a buffer (SduDataPtr) containing the PDU, and the MetaData related to this PDU.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	Indication of a received PDU from a lower layer communication interface module. This function is used by user.	
Available via	CanNm.h	

Function Name	CanNm_ConfirmPnAvailability	
Syntax	void CanNm_ConfirmPnAvailability (NetworkHandleType nmChannelHandle)	
Service ID [Hex]	0x16	
Sync/Async	Synchronous	
Reentrancy	Reentrant (but not for the same NM-channel)	
Parameters (In)	nmChannelHandle	Identification of the NM-channel
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	None	
Description	Enables the PN filter functionality on the indicated NM channel. Availability: The API is only available if CanNmGlobalPnSupport is TRUE. This function is used by user. But it needs configuration. (It cannot be called directly by user)	
Available via	CanNm.h	

6.3.5 Scheduled functions

Service name:	CanNm_MainFunction
Syntax:	void CanNm_MainFunction(void)
Service ID[hex]:	0x13
Description:	Main function of the CanNm which processes the algorithm describes in that document. This function is used by BSW.
Available via:	SchM_CanNm.h

7 Generator

7.1 Generator Option

Option	Description
-G,-- Generation	Symbolic parameters to be used for fore generation (skip validation).
-H,--Help	Display this help message.
-I,--Input <I>	ECU description file path of the module for which generation tool needs to run.
-L,--Log	Symbolic parameters to be used for the generation error log.
-M,--Module <M>	Specify module name and version to be generated code for.
-T,--Top_path <T>	Symbolic parameters to be used for the set path of the module.
-V,--Validate	Symbolic parameters to be used for invoking validation checks.
-O, --Output <O>	Project-relative path to the location where the generated code is to be placed.
-B,-- Tool_path	Location where file .jar (Nm_R44/generator/com.autron.odin.generator.bsw.nm_x.x.x.jar)

7.2 Generator Message

7.2.1 Error Messages

ERR0310001: In the container "CanNmChannelConfig" <index>, If "CanNmPnEraCalcEnabled" is set as TRUE, "CanNmPnEraRxNSduRef" shall be set.

This error occurs, when parameter "CanNmPnEraCalcEnabled" is set to TRUE and parameter "CanNmPnEraRxNSduRef" isn't configured.

ERR0310002: In the container "CanNmChannelConfig" <index>, If parameter "CanNmPnEraRxNSduRef" is set, parameter "PduRSrcPdu" in the module PduR shall be set.

This error occurs, when the parameter "CanNmPnEraRxNSduRef" has configuration and "PduRSrcPdu" isn't configured.

ERR0310003: "CanNmPnEraCalcEnabled" in "CanNmChannelConfig" <index> is only valid when "CanNmGlobalPnSupport" in "CanNmGlobalConfig" is set to TRUE. "CanNmGlobalPnSupport" in "CanNmGlobalConfig" shall be set.

This error occurs when the parameter "CanNmGlobalPnSupport" is set to FALSE or isn't configured and CanNmPnEraCalcEnabled is set to TRUE.

ERR0310004: "CanNmPnResetTime" in "CanNmGlobalConfig" shall be greater than "CanNmMsgCycleTime" in "CanNmChannelConfig" <index>. And "CanNmPnResetTime" in "CanNmGlobalConfig" shall be less than "CanNmTimeoutTime" in "CanNmChannelConfig" <index>.

This error occurs when parameter "CanNmPnResetTime" less than equal "CanNmMsgCycleTime" or parameter "CanNmPnResetTime" greater than equal "CanNmTimeoutTime".

ERR0310005: "CanNmTimeoutTime" in "CanNmChannelConfig" shall be greater than "CanNmMsgCycleTime" in "CanNmChannelConfig" <index>.

This error occurs when parameter "CanNmTimeoutTime" less than equal "CanNmMsgCycleTime".

ERR0310006: "CanNmPnHandleMultipleNetworkRequests" in "CanNmChannelConfig" is only valid if "CanNmGlobalPnSupport" in "CanNmGlobalConfig" is set as TRUE.

This error occurs when the parameter "CanNmPnHandleMultipleNetworkRequests" is set to TRUE and "CanNm-GlobalPnSupport" is set to FALSE.

ERR0310007: If "ComMPncNmRequest" in "ComMNetworkManagement" is set TRUE, "CanNmPnHandleMultipleNetworkRequests" in "CanNmChannelConfig" <index> shall be set as TRUE.

This error occurs when the parameter “ComMPncNmRequest” is set to TRUE and “CanNmPnHandle-MultipleNetworkRequests” is set to FALSE.

ERR031008: "PduRSourcePduHandleId" in "PduRSrcPdu" shall be set.

This error occurs if the parameter “PduRSourcePduHandleId” is not configured.

ERR0310009: "CanNmComControlEnabled" in "CanNmGlobalConfig" shall equal FALSE if "CanNmPassiveMode-Enabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter “CanNmComControlEnabled” and “CanNmPassiveModeEnabled” is set to TRUE.

ERR0310010: "CanNmComControlEnabled" in CanNmGlobalConfig shall equal "NmComControlEnabled" in NmGlobalFeatures if "CanNmPassiveModeEnabled" in CanNmGlobalConfig equals FALSE.

This error occurs when the parameter state “CanNmComControlEnabled” different from the state of “NmComControlEnabled” and “CanNmPassiveModeEnabled” is set to FALSE.

ERR0310011: "CanNmComUserDataSupport" in “CanNmGlobalConfig” shall equal FALSE if “CanNmPassive-ModeEnabled ” in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter “CanNmComUserDataSupport” and “CanNmPassiveModeEnabled” is set to TRUE.

ERR0310012: Value of parameter "CanNmNodeIdEnabled " in "CanNmChannelConfig" shall be set as TRUE.

This error occurs when the parameter “CanNmNodeIdEnabled” isn’t set to TRUE.

ERR0310013: "CanNmPduCbvPosition" in "CanNmChannelConfig"<index> shall be set as CANNM_PDU_BYTE_0 or CANNM_PDU_BYTE_1 if "CanNmNodeDetectionEnabled" in "CanNmChannelConfig"<index> equals TRUE.

This error occurs when parameter “CanNmNodeDetectionEnabled” is set to TRUE and set parameter “CanNmPduCbvPosition” with a value equal to “CANNM_PDU_OFF”.

ERR0310014: Parameter "CanNmPduCbvPosition" and "CanNmPduNidPosition" in "CanNmChannelConfig" <index> shall not be set as the same value exclude CANNM_PDU_OFF.

This error occurs when parameters “CanNmPduCbvPosition” and “CanNmPduNidPosition” have a value different with value “CANNM_PDU_OFF” and parameter “CanNmPduCbvPosition” have the position of the control bit vector equal parameter “CanNmPduNidPosition”.

ERR0310015: \$ParameterName in "PduRBswModules" shall be set as TRUE.

This error occurs when the parameter “**CanNmComUserDataSupport**” is set to TRUE and “**\$ParameterName**” is set to FALSE.

\$ParameterName: PduRTriggerTransmit, PduRTxConfirmation.

ERR0310016: If “**CanNmImmediateNmTransmissions**” in “**CanNmChannelConfig**” is greater than 0, “**CanNm-ImmediateNmCycleTime**” in “**CanNmChannelConfig**” shall be set.

This error occurs when parameter value “**CanNmImmediateNmTransmissions**” greater than 0 and value of “**Can-NmImmediateNmCycleTime**” isn’t set.

ERR0310017: “**CanNmMsgTimeoutTime**” in “**CanNmChannelConfig**” <index> shall not be greater than **\$Param-eterName** in “**CanNmChannelConfig**” <index>.

This error occurs when the parameter value of “**CanNmMsgTimeoutTime**” higher parameter value of “**\$ParameterName**”.

\$ParameterName: CanNmMsgCycleTime.

ERR0310018: “**PduRDestPduHandleId**” in “**PduRDestPdu**” shall be set.

This error occurs when the parameter “**PduRDestPduHandleId**” is not set.

ERR0310019: Value of each parameter “**CanNmPnFilterMaskByteIndex**” in container “**CanNmPnFilter-MaskByte**” is not unique.

This error occurs when more than one parameter “**CanNmPnFilterMaskByteIndex**” has been set as the same value.

ERR0310020: “**PduRBswModules**” in PduR shall be set for the CanNm.

This error occurs when the parameter “**PduRBswModules**” is not set.

ERR0310021: In the container “**CanNmGlobalConfig**”, If “**CanNmPnEiraCalcEnabled**” is set as TRUE, “**CanNmPn-EiraRxNSduRef**” shall be set.

This error occurs when the parameter “**CanNmPnEiraCalcEnabled**” is set to TRUE and “**CanNmPnEiraRxNSduRef**” isn’t configured.

ERR0310022: In the container “**CanNmGlobalConfig**”, If “**CanNmGlobalPnSupport**” is set as TRUE, “**CanNmPnResetTime**” shall be set.

This error occurs when the parameter “**CanNmGlobalPnSupport**” is set to TRUE and “**CanNmPnResetTime**” isn’t configured.

ERR0310024: Parameter “**CanNmPnInfoOffset**” in “**CanNmPnInfo**” container shall not be set less than number of enabled system bytes (CBV, NID).

This error occurs when parameter **"CanNmPduCbvPosition"** and **"CanNmPduNidPosition"** with a value different to **"CANNM_PDU_OFF"** and **"CanNmPnInfoOffset"** value range is less than 2 (in bytes) or parameter **"CanNmPduCbvPosition"** with a value different to **"CANNM_PDU_OFF"** and parameter **"CanNmPduNidPosition"** with a value equal to **"CANNM_PDU_OFF"** and **"CanNmPnInfoOffset"** value range is less than 1 (in bytes) or parameter **"CanNmPduCbvPosition"** with a value equal to **"CANNM_PDU_OFF"** and parameter **"CanNmPduNidPosition"** with a value different to **"CANNM_PDU_OFF"** and **"CanNmPnInfoOffset"** value range is less than 1 (in bytes).

ERR0310025: "CanNmPnFilterMaskByte" container in "CanNmPnInfo" must be set when "CanNmGlobalPnSupport" in "CanNmGlobalConfig" is set as TRUE.

This error occurs when the parameter **"CanNmGlobalPnSupport"** is set to TRUE and **"CanNmPnFilterMaskByte"** container in **"CanNmPnInfo"** is not configured.

ERR0310027: "CanNmRemoteSleepIndEnabled" in "CanNmGlobalConfig" shall equal "NmRemoteSleepInd-Enabled" in "NmGlobalFeatures" if "CanNmPassiveModeEnabled" in "CanNmGlobalConfig" equals FALSE.

This error occurs when value of parameter **"CanNmRemoteSleepIndEnabled"** different with value of parameter **"NmRemoteSleepIndEnabled"** and **"CanNmPassiveModeEnabled"** is set to FALSE.

ERR0310028: "CanNmStateChangeIndEnabled" in "CanNmGlobalConfig" shall equal "NmStateChangeIndEnabled" in "NmGlobalFeature".

This error occurs when value of parameter **"NmStateChangeIndEnabled"** different with value of parameter **"CanNmStateChangeIndEnabled"**.

ERR0310029: "CanNmUserDataEnabled" in "CanNmGlobalConfig" shall equal "NmUserDataEnabled" in "NmGlobalFeatures".

This error occurs when the state of parameter **"CanNmUserDataEnabled"** different from the state of parameter **"NmUserDataEnabled"**.

ERR0310030: "NmPduRxIndicationEnabled" in "CanNmGlobalConfig" shall equal "CanNmPduRxIndication-Enabled" in "NmGlobalFeatures".

This error occurs when the state of parameter **"CanNmPduRxIndicationEnabled"** different from the state of parameter **"NmPduRxIndicationEnabled"**.

ERR0310031: In the container "CanNmTxPdu" <index>, If parameter "CanNmTxPduRef" is set, parameter "PduRDestPdu" in the module PduR shall be set.

This error occurs when parameter **"CanNmTxPduRef"** is set a **"PduRDestPdu"** isn't configured.

ERR0310032: "\$ParameterName" in "CanNmTxPdu" in container CanNmChannelConfig shall be set.

This error occurs when parameter **"\$ParameterName"** is not set.

\$ParameterName: "CanNmTxPduRef", "CanNmTxConfirmationPduId".

ERR0310033: If parameter "CanNmComUserDataSupport" in container "CanNmGlobalConfig" is set as TRUE, container "CanNmUserDataTxPdu" shall be set.

This error occurs when the parameter "CanNmComUserDataSupport" is set to TRUE and the number of "CanNmUserDataTxPdu" different with value 1.

ERR0310034: In the container "CanNmGlobalConfig", If parameter "CanNmPnEiraRxNSduRef" is set, the parameter "PduRSrcPdu" in the module PduR shall be set.

This error occurs when parameter "CanNmPnEiraRxNSduRef" is set and "PduRSrcPdu" is not configured.

ERR0310035: In the container "CanNmRxPdu" <index>, If parameter "CanNmRxPduRef" is set, parameter "PduRSrcPdu" in the module PduR shall be set.

This error occurs when parameter "CanNmRxPduRef" is set and "PduRSrcPdu" is not configured.

ERR0310036: In the container "CanNmUserDataTxPdu" <index>, If parameter "CanNmTxUserDataPduRef" is set, parameter "PduRDestPdu" in the module PduR shall be set.

This error occurs when the parameter "CanNmTxUserDataPduRef" is set and "PduRDestPdu" is not configured.

ERR0310037: "CanNmPnEnabled" in "CanNmChannelConfig" is only valid when "CanNmGlobalPnSupport" in "CanNmGlobalConfig" is set as TRUE. "CanNmGlobalPnSupport" in "CanNmGlobalConfig" shall be set.

This error occurs when the parameter "CanNmGlobalPnSupport" is set to FALSE and "CanNmPnEnabled" is set to TRUE.

ERR0310038: "CanNmPnInfo" in "CanNmGlobalConfig" is only valid when "CanNmGlobalPnSupport" in "CanNmGlobalConfig" is set as TRUE. "CanNmGlobalPnSupport" in "CanNmGlobalConfig" shall be set as TRUE.

This error occurs when the parameter "CanNmGlobalPnSupport" is set to FALSE and "CanNmPnInfo" is configured.

ERR0310039: In container "CanNmGlobalConfig", If parameter "CanNmPnEiraCalcEnables" is set as TRUE, parameter "CanNmGlobalPnSupport" shall be set as TRUE.

This error occurs when parameter "CanNmPnEiraCalcEnabled" is set to TRUE and "CanNmGlobalPnSupport" is set to FALSE.

ERR0310040: "CanNmNodeDetectionEnabled" in container "CanNmChannelConfig" <index> shall equal FALSE if "CanNmPassiveModeEnabled" in container "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **"CanNmNodeDetectionEnabled"** and **"CanNmPassiveModeEnabled"** set to TRUE.

ERR0310041: Container "CanNmTxPdu" in "CanNmChannelConfig"<index> shall be set.

This error occurs if a number of parameters **"CanNmTxPdu"** can be set differently as to 1.

ERR0310042: "CanNmRepeatMsgIndEnabled" in "CanNmChannelConfig" <index> shall equal FALSE if "CanNmPassiveModeEnabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **"CanNmRepeatMsgIndEnabled"** and **"CanNmPassiveModeEnabled"** is set to TRUE.

ERR0310043: "CanNmImmediateNmTransmissions" in "CanNmChannelConfig"<index> shall not be set as 0 if "CanNmPnHandleMultipleNetworkRequests" in "CanNmChannelConfig"<index> equals TRUE.

IF ((CanNmPnHandleMultipleNetworkRequests == TRUE) && (CanNmImmediateNmTransmissions == 0))

This error occurs when parameter **"CanNmPnHandleMultipleNetworkRequests"** is set to TRUE and parameter **"CanNmImmediateNmTransmissions"** is set to 0.

ERR0310044: "CanNmImmediateNmTransmissions" in "CanNmChannelConfig" <index> shall be set as 0 if "CanNmImmediateRestartEnabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **"CanNmImmediateRestartEnabled"** is set to TRUE and value of parameter **"CanNmImmediateNmTransmissions"** is set different 0.

ERR0310045: "CanNmRepeatMessageTime" in "CanNmChannelConfig"<index> shall be set larger than "CanNmImmediateNmTransmissions * CanNmImmediateNmCycleTime" in "CanNmGlobalConfig" equals TRUE.

This error occurs when value of parameter **"CanNmRepeatMessageTime"** less than or equal value of **"CanNmImmediateNmTransmissions" * "CanNmImmediateNmCycleTime"**.

ERR0310046: "CanNmComControlEnabled" in container "CanNmGlobalConfig" shall be set equal "NmComControlEnabled" in Nm module if "CanNmPassiveModeEnabled" in container "CanNmGlobalConfig" equal FALSE.

This error occurs when parameter **"CanNmPassiveModeEnabled"** is set to FALSE and the value of parameter **"CanNmComControlEnabled"** is set different with **"NmComControlEnabled"**.

ERR0310047: "CanNmImmediateTxconfEnabled" in container "CanNmGlobalConfig" shall be set as FALSE if "CanNmPassiveModeEnabled" in container "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter “CanNmImmediateTxconfEnabled” and “CanNmPasiveModeEnabled” is set to TRUE.

ERR0310048: Parameter "CanNmCarWakeUpBytePosition" in "CanNmChannelConfig" <index> shall not be set less than number of enabled system bytes (CBV, NID)

This error occurs when parameter “CanNmPduCbvPosition” and “CanNmPduNidPosition” with a value different to “CANNM_PDU_OFF” and “CanNmCarWakeUpBytePosition” value range is less than 2 (in bytes) or parameter “CanNmPduCbvPosition” with a value different to “CANNM_PDU_OFF” but parameter “CanNmPduNidPosition” with a value equal to “CANNM_PDU_OFF” and “CanNmCarWakeUpBytePosition” value range is less than 1 (in bytes) or parameter “CanNmPduCbvPosition” with a value equal to “CANNM_PDU_OFF” but parameter “CanNmPduNidPosition” with a value different to “CANNM_PDU_OFF” and “CanNmCarWakeUpBytePosition” value range is less than 1 (in bytes).

ERR0310049: Parameter "PduLength" in [Pdu] container which CanNmTxPduRef refer to should be set as 8

This error occurs when “PduLength” in [Pdu] container which “CanNmTxPduRef” refer to be set different as 8.

ERR0310050: Parameter "CanNmMsgReducedTime" in "CanNmChannelConfig"<index> container shall follow rule $0.5 * \text{CanNmMsgCycleTime} \leq \text{CanNmMsgReducedTime} < \text{CanNmMsgCycleTime}$ if "CanNmBusLoad-ReductionActive" equals TRUE.

This error occurs when parameter “CanNmBusLoadReductionActive” shall be call and value of “CanNmMsg-ReducedTime” less than $0.5 * \text{CanNmMsgCycleTime}$ or when value of “CanNmMsgCycleTime” less than “CanNmMsgReducedTime”.

ERR0310051: Parameter "CanNmBusLoadReductionEnabled" in "CanNmGlobalConfig" container shall be set as FALSE if "CanNmPassiveModeEnabled" or "CanNmGlobalPnSupport" in container "CanNmGlobalConfig" equals TRUE.

This error occurs when the parameter “CanNmBusLoadReductionEnabled” is set to TRUE and “CanNmPassive-ModeEnabled” is set to TRUE or “CanNmGlobalPnSupport” is set to TRUE.

ERR0310052: Parameter "CanNmBusLoadReductionActive" in "CanNmChannelConfig" <index> container shall be set as FALSE if "CanNmBusLoadReductionEnabled" in container "CanNmGlobalConfig" equals FALSE.

This error occurs when the parameter “CanNmBusLoadReductionEnabled” is set to FALSE and “CanNm-BusLoadReductionActive” is set to TRUE.

ERR0310053: Number of "CanNmPnFilterMaskByte" shall equals value of "CanNmPnInfoLength" in container "CanNmPnInfo"

This error occurs when the parameter **“CanNmPnInfoLength”** is not equal MAX of **“CanNmPnFilterMask-ByteIndex” + 1**

ERR0310054: "CanSMPncSupport " in "CanSMGeneral" container shall be set as TRUE when "CanNmGlobal-PnSupport" in "CanNmGlobalConfig" is set as TRUE.

This error occurs when the parameter **“CanNmGlobalPnSupport”** is set to TRUE and **“CanSMPncSupport”** is not set TRUE.

ERR0310055: "CanNmImmediateRestartEnabled" in "CanNmGlobalConfig" container shall be set as FALSE when "CanNmPassiveModeEnabled" in "CanNmGlobalConfig" is set as TRUE.

This error occurs when parameter **“CanNmImmediateRestartEnabled”** and **“CanNmPassiveModeEnabled ”** is set to TRUE.

ERR0310056: Please change "CanNmTxUserDataPduRef" of Channel 1 to the I-PDU with the suitable length. The suitable length, in this case, is 6.

This error occurs if the number of available user data bytes is not equal to the length of the referenced I-PDU in **“CanNmTxUserDataPduRef”**.

ERR0310057: CanNm module supported Post-Build but there is no variant configured in ECUC.

This error occurs if CanNm module supported Post-Build but there is no variant configured in ECUC.

ERR0310058: Mismatch post-build variant with EcuC module, the module's post-build variants should include all EcuC post-build variants.

This error occurs if CanNm module's post-build variants list does not include all EcuC post-build variants.

ERR0310059: "CanNmPassiveModeEnabled" in "CanNmGlobalConfig" shall equal FALSE if "CanNmPassiveNodeEnabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **“CanNmPassiveModeEnabled”** and **“CanNmPassiveNodeEnabled”** is set to TRUE.

ERR0310060: "CanNmBusSynchronizationEnabled" in "CanNmGlobalConfig" shall equal FALSE if "CanNmPassiveNodeEnabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **“CanNmBusSynchronizationEnabled”** and **“CanNmPassiveNodeEnabled”** is set to TRUE.

ERR0310061: "CanNmCoordinatorSyncSupport" in "CanNmGlobalConfig" shall equal FALSE if "CanNmPassiveNodeEnabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **“CanNmCoordinatorSyncSupport”** and **“CanNmPassiveNodeEnabled”** is set to TRUE.

ERR0310062: "CanNmRemoteSleepIndEnabled" in "CanNmGlobalConfig" shall equal FALSE if "CanNmPassiveNodeEnabled" in "CanNmGlobalConfig" equals TRUE.

This error occurs when parameter **"CanNmRemoteSleepIndEnabled"** and **"CanNmPassiveNodeEnabled"** is set to TRUE.

7.2.2 Warning Messages

WRN0310001: Parameter "CanNmPnEraRxNSduRef" in "CanNmChannelConfig"<index> is only valid when "CanNmPnEraCalcEnabled" in "CanNmGlobalConfig" is set as TRUE. Parameter "CanNmPnEraCalcEnabled" in "CanNmGlobalConfig" shall be set.

This information occurs if parameter **"CanNmPnEraCalcEnabled"** is set to FALSE and **"CanNmPnEraRxNSduRef"** is configured different null.

WRN0310002: Parameter "CanNmPnEiraRxNSduRef" in "CanNmGlobalConfig" is only valid when "CanNmPnEiraCalcEnabled" in "CanNmGlobalConfig" is set as TRUE. Parameter "CanNmPnEiraCalcEnabled" in "CanNmGlobalConfig" shall be set.

This information occurs if parameter **"CanNmPnEiraCalcEnabled"** is set to FALSE and **"CanNmPnEiraRxNSduRef"** is configured different null.

7.2.3 Information Messages

Module CanNm doesn't have Information Messages.

8 SWP Error Code

None

9 Appendix