Document Title	Specification of Vehicle-2-X Geo Networking
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	793

Document Status	Final
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	4.3.1

Document Change History			
Date	**Release**	**Changed by**	**Change Description**
2017-12-08	4.3.1	AUTOSAR Release Management	• Editorial changes
2016-11-30	4.3.0	AUTOSAR Release Management	• Initial Release

Disclaimer

Table of Contents

1 Introduction and functional overview 5
1.1 Architectural overview 5
1.2 Functional overview 6
2 Acronyms and abbreviations 7
3 Related documentation 8
3.1 Input documents 8
3.2 Related standards and norms 8
3.3 Related specification 9
4 Constraints and assumptions 10
4.1 Limitations 10
4.2 Applicability to car domains 10
5 Dependencies to other modules 11
5.1 AUTOSAR DET (Default Error Tracer) 11
5.2 AUTOSAR EcuM (Ecu State Manager) 11
5.3 AUTOSAR Ethernet Interface (EthIf) 11
5.4 AUTOSAR Vehicle-2-X Basic Transport Protocol (V2xBtp) 11
5.5 AUTOSAR Vehicle-2-X Management (V2xM) 11
5.6 File structure 11
5.6.1 Code file structure 11
5.6.2 Header file structure 11
6 Requirements traceability 13
7 Functional specification 16
7.1 General Functionality 16
7.2 GeoNetworking Packet Structure and Format 16
7.3 GeoNetworking Protocol Operations 17
7.3.1 Network Management 17
7.3.2 Security Mechanisms 17
7.4 Message Forwarding 18
7.5 Message Transmission 18
7.6 Message Reception 19
7.7 Error classification 20
7.7.1 Development Errors 20
7.7.2 Runtime Errors 20
7.7.3 Transient Faults 20
7.7.4 Production Errors 20
7.7.5 Extended Production Errors 20
8 API specification 21
8.1 Imported types 21
8.2 Type definitions 21
8.2.1 V2xGn_ConfigType 21
8.2.2 V2xGn_TxParamsType 21

8.3] () Function definitions 22
8.3.1 V2xGn_Init 22
8.3.2 V2xGn_GetVersionInfo 22
8.3.3 V2xGn_V2xM_PreparePseudonymChange 23
8.3.4 V2xGn_V2xM_CommitPseudonymChange 24
8.3.5 V2xGn_V2xM_AbortPseudonymChange 24
8.3.6 V2xGn_Transmit 25
8.4 Call-back notifications 26
8.4.1 V2xGn_V2xM_EncapConfirmation 26
8.4.2 V2xGn_V2xM_DecapConfirmation 27
8.4.3 V2xGn_RxIndication 27
8.4.4 V2xGn_TxConfirmation 28
8.5 Scheduled functions 29
8.5.1 V2xGn_MainFunction 29
8.6 Expected Interfaces 29
8.6.1 Mandatory Interfaces 29
8.6.2 Optional Interfaces 30
9 Sequence diagrams 31
9.1 V2xGn_RxIndication 31
9.2 V2xGn_Transmit 32
9.3 V2xGn_V2xM_UpdatePseudonym 33
9.4 V2xGn_MainFunction 33
10 Configuration specification 34
10.1 Containers and configuration parameters 34
10.1.1 Variants 34
10.1.2 V2xGn 34
10.1.3 V2xGn 34
10.1.4 V2xGnGeneral 34
10.1.5 V2xGnGeneral 34
10.1.6 V2xGnBeaconService 39
10.1.7 V2xGnBeaconService 39
10.1.8 V2xGnPacketForwarding 40
10.1.9 V2xGnPacketForwarding 40
10.2 Published Information 44
11 Not applicable requirements 45

1 Introduction and functional overview

This specification specifies the functionality, API and the configuration of the AUTOSAR Basic Software module Vehicle-2-X GeoNetworking (V2xGn).

V2xGn together with Vehicle-2-X Facilities (V2xFac) [8], Vehicle-2-X Basic Transport (V2xBtp) [9], Vehicle-2-X Management (V2xM) [10] and AUTOSAR BSW modules Ethernet Interface (EthIf) [5], Wireless Ethernet Driver (WEth) [11] and Wireless Ethernet Transceiver Driver (WEthTrcv) [12] forms the V2X stack within the AUTOSAR architecture.

The base for this document is the GeoNetworking specification [18][19]. It is assumed that the reader is familiar with this specification.

1.1 Architectural overview

V2xGn provides services to and is dependent on the upper V2xBtp module and uses the services of and gets services from the lower EthIf module to realize its functions explained in sections 1.2 and chapter 7 of this document.

Positioning of the V2xGn module within the AUTOSAR BSW and the Layered Software architecture [1] is shown in Figure 1.

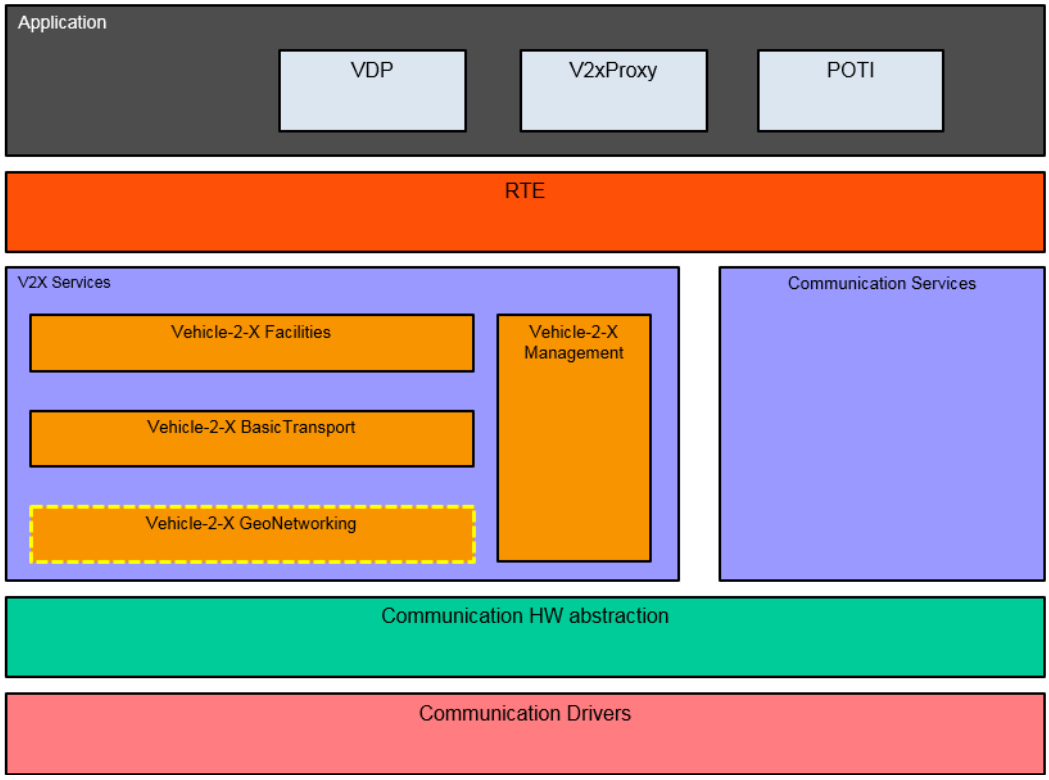

Figure 1 – AUTOSAR BSW software architecture – V2xGn module scope

1.2 Functional overview

The internal functionality of the V2xGn module should comply to the medium independent specification of the GeoNetworking protocol [18] and the medium dependent specification of the GeoNetworking protocol [19], relying on ETSI ITS-G5 technology as medium. The module provides services to the upper V2xBtp module specified in [9] and in order to provide its packet transport services, it relies on the lower EthIf module [5]. Vehicle-2-X specific data is also exchanged with the V2xM module.

GeoNetworking protocol is a set of network layer functionalities that enables ad hoc communication without infrastructure support using geographical positions of the communicating entities. It supports communication among individual Intelligent Transport System (ITS) station and distribution of packets in geographical areas. As GeoNetworking can be executed over different ITS technologies such as ITS-G5 and infrared, GeoNetworking specification consists of a standard for media-independent functionality [18] which specifies all functions that are common to all ITS access technologies and one or more media-dependent specifications [19] which includes extensions for a specific ITS technology.

2 Acronyms and abbreviations

The following acronyms and abbreviations have a local scope and are therefore not contained in the AUTOSAR glossary [4].

Abbreviation / Acronym:	***Description:***
BTP	Basic Transport Protocol
CBF	Contention-Based Forwarding
DET	Default Error Tracer
GAC	GeoAnycast
GBC	GeoBroadcast
GN	GeoNetworking
GN-SDU	GeoNetworking Service Data Unit
ITS	Intelligent Transport System
MAC	Medium Access Control
SHB	Single Hop Broadcast
TC	Traffic Class
TSB	Topologically Scoped Broadcast

3 Related documentation

3.1 Input documents

[1] AUTOSAR Layered Software Architecture
AUTOSAR_EXP_LayeredSoftwareArchitecture.pdf

[2] AUTOSAR General Requirements on Basic Software Modules
AUTOSAR_SRS_BSWGeneral.pdf

[3] AUTOSAR General Specification for Basic Software Modules
AUTOSAR_SWS_BSWGeneral.pdf

[4] Glossary
AUTOSAR_TR_Glossary

[5] Specification of Ethernet Interface
AUTOSAR_SWS_EthernetInterface.pdf

[6] Specification of ECU State Manager
AUTOSAR_SWS_ECUStateManager.pdf

[7] Specification of Default Error Tracer
AUTOSAR_SWS_DefaultErrorTracer.pdf

[8] Specification of Vehicle-2-X Facilities
AUTOSAR_SWS_V2XFacilities.pdf

[9] Specification of Vehicle-2-X Basic Transport
AUTOSAR_SWS_V2XBasicTransport.pdf

[10] Specification of Vehicle-2-X Management
AUTOSAR_SWS_V2XManagement.pdf

[11] Specification of Wireless Ethernet Driver
AUTOSAR_SWS_WirelessEthernetDriver.pdf

[12] Specification of Wireless Ethernet Transceiver Driver
AUTOSAR_SWS_WirelessEthernetTransceiverDriver.pdf

3.2 Related standards and norms

[13] Intelligent Transport Systems (ITS); Communications Architecture
ETSI EN 302 665 V1.1.1 (2010-09)

[14] Intelligent Transport Systems (ITS); Vehicular Communications;
GeoNetworking; Part 1: Requirements
ETSI EN 302 636-1 V1.2.1 (2014-04)

[15] Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 2: Scenarios
ETSI EN 302 636-2 V1.2.1 (2013-11)

[16] Intelligent Transport Systems (ITS); Vehicular Communications GeoNetworking Part 3: Network Architecture
ETSI EN 302 636-3 V1.2.1 (2014-12)

[17] Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol
ETSI EN 302 636-5-1 V1.2.1 (2014-08)

[18] Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality
ETSI EN 302 636-4-1 V1.2.1 (2014-07)

[19] Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 2: Media-dependent functionalities for ITS-G5
ETSI TS 102 636-4-2 V1.1.1 (2013-10)

[20] Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 8: Interface between security entity and network and transport layer
ETSI TS 102 723-8

[21] Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 10: Interface between access layer and network and transport layer
ETSI TS 102 723-10 V1.1.1 (2012-11)

[22] C2C-CC BSP Requirement
C2CCC_RS_2037_BSP_Requirements.docx

[23] Intelligent Transport Systems (ITS); Harmonized Channel Specifications for Intelligent Transport Systems operating in the 5 GHz frequency band between access layer and network and transport layer
ETSI TS 102 724 V1.1.1 (2012-10)

[24] List of EtherTypes by IEEE
http://standards.ieee.org/develop/regauth/ethertype/eth.txt

3.3 Related specification

AUTOSAR provides a General Specification on Basic Software (SWS BSW General) [3] which is also valid for V2xGn.

Thus, the specification SWS BSW General [3] shall be considered as additional and required specification for V2xGn.

4 Constraints and assumptions

4.1 Limitations

- The GeoNetworking protocol and therefore the V2xGn module requires a broadcast capable access layer in order to provide transmit services.
- Wireless Communication supports IEEE 802.11p only. Other 802.11 standards (e.g. for infrastructure networks and integration with TCP/IP) can be extended in future releases of the AUTOSAR standard.
- The V2X modules follow the guidance regarding the Day-1 scenarios defined by the Basic System Standards Profile from Car-2-Car-Consortium.
- AUTOSAR R4.3.0 only focuses on the European version of car-to-car communication as defined by ETSI. Extension to other regions are planned for future releases of the AUTOSAR standard.

4.2 Applicability to car domains

This specification is applicable to all car domains.

5 Dependencies to other modules

This section describes the relations of the V2xGn module to other modules within the AUTOSAR basic software architecture. It outlines the modules that are required or optional for the realization of the V2xGn module and the V2xGn services that these modules use.

5.1 AUTOSAR DET (Default Error Tracer)

In development mode, the V2xGn module reports errors through DET [7].

5.2 AUTOSAR EcuM (Ecu State Manager)

The EcuM [6] is responsible for the initialization of V2xGn.

5.3 AUTOSAR Ethernet Interface (EthIf)

The Ethernet Interface is the lower layer module of the V2xGn module.

5.4 AUTOSAR Vehicle-2-X Basic Transport Protocol (V2xBtp)

The V2xBtp is the upper layer module of the V2xGn module.

5.5 AUTOSAR Vehicle-2-X Management (V2xM)

V2xM is used for interchange of Data with other V2X-Modules. Security mechanisms are configure for the V2xM and are used by V2xGn.

5.6 File structure

5.6.1 Code file structure

For details refer to the chapter 5.1.6 “Code file structure” in *SWS_BSWGeneral* [3].

5.6.2 Header file structure

[SWS_V2xGn_00001] ⌈ The implementation header files shall include *V2x_GeneralTypes.h*. ⌋ (SRS_BSW_00301, SRS_BSW_00456)

[SWS_V2xGn_00006] ⌈ Generic type definitions of the V2xGn module which are described in section 8.2 shall be located in the header file V2xGn_Types.h. ⌋ ( )

Figure 2 shows the include hierarchy of the V2xGn module.

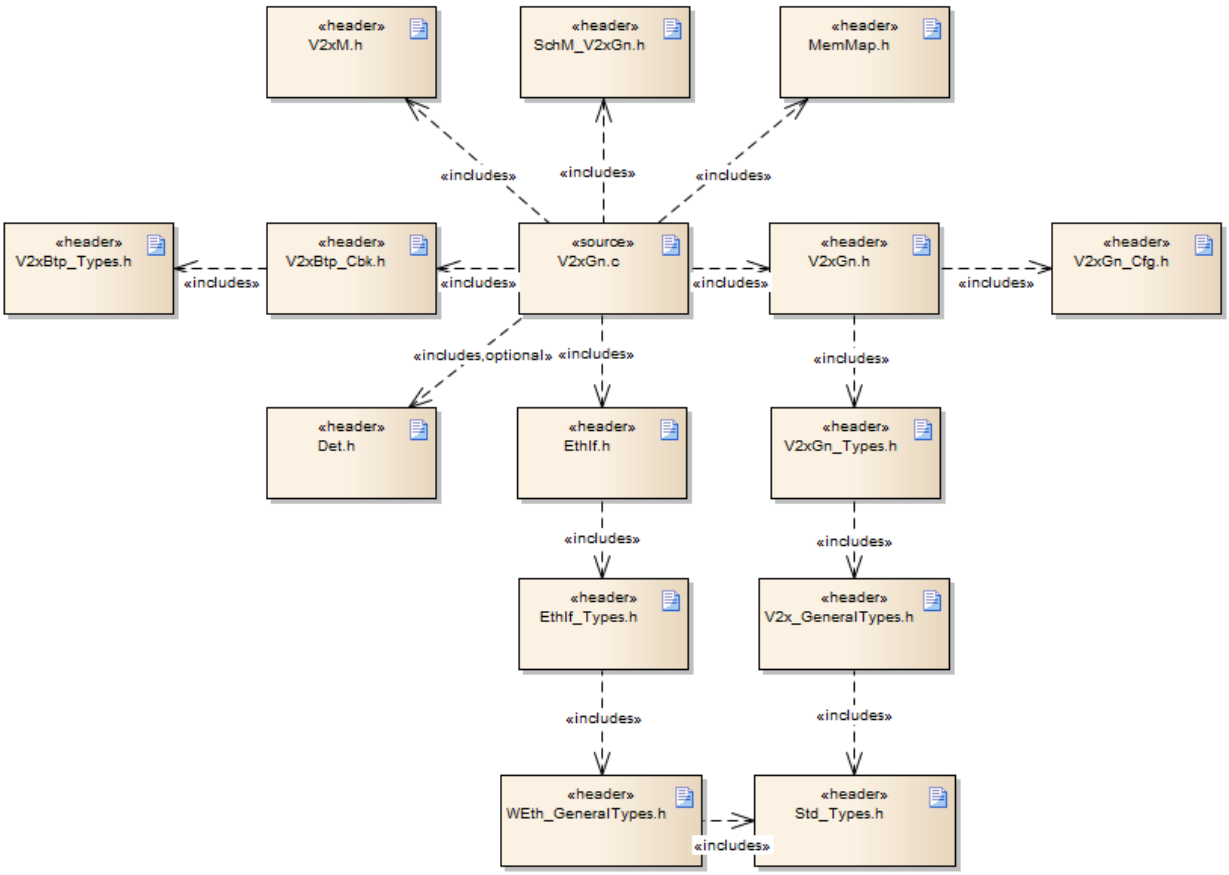

Figure 2 – Code and header file structure of the V2xGn Module

6 Requirements traceability

Requirement	Description	Satisfied by
SRS_BSW_00301	All AUTOSAR Basic Software Modules shall only import the necessary information	SWS_V2xGn_00001
SRS_BSW_00345	BSW Modules shall support pre-compile configuration	SWS_V2xGn_00078
SRS_BSW_00456	- A Header file shall be defined in order to harmonize BSW Modules	SWS_V2xGn_00001
SRS_V2X_00010	The implementation of the V2X system shall follow additional guidance given by C2C-CC requirements	SWS_V2xGn_20169, SWS_V2xGn_20181, SWS_V2xGn_20260, SWS_V2xGn_20262, SWS_V2xGn_20263, SWS_V2xGn_20264, SWS_V2xGn_20265, SWS_V2xGn_20266, SWS_V2xGn_20267, SWS_V2xGn_20268, SWS_V2xGn_20269, SWS_V2xGn_20270, SWS_V2xGn_20401
SRS_V2X_00160	The V2X system shall use end-to-end security for communication to external entities	SWS_V2xGn_00026, SWS_V2xGn_20251
SRS_V2X_00161	The V2X system shall employ the security envelope on its Network layer	SWS_V2xGn_00012, SWS_V2xGn_20251
SRS_V2X_00163	The "verification" of a message shall comprise at least cryptographic verification of the message's signature	SWS_V2xGn_NA_3
SRS_V2X_00164	The V2X system shall only forward verified messages	SWS_V2xGn_00026
SRS_V2X_00174	The V2X system shall support key origin authentication for the new (long-term or pseudonym) public keys that are provided in certificate signing requests	SWS_V2xGn_NA_3
SRS_V2X_00176	The V2X system shall change pseudonyms	SWS_V2xGn_00028, SWS_V2xGn_00091, SWS_V2xGn_00112, SWS_V2xGn_00115
SRS_V2X_00184	The V2X system shall allow applications to block the pseudonym change	SWS_V2xGn_NA_3
SRS_V2X_00189	The V2X system shall be able to estimate vehicle states	SWS_V2xGn_NA_4
SRS_V2X_00190	The V2X system shall handle vehicle states in a consistent manner	SWS_V2xGn_NA_4
SRS_V2X_00193	The V2X system shall use ITS time as time base	SWS_V2xGn_NA_4
SRS_V2X_00207	The difference between	SWS_V2xGn_NA_4

	Station clock and time base shall be estimated	
SRS_V2X_00214	The V2X system shall allow applications to deactivate transmission of CAMs	SWS_V2xGn_NA_4
SRS_V2X_00232	The V2X system shall cooperate with tolling zone stations in vicinity	SWS_V2xGn_NA_1
SRS_V2X_00239	The V2X DCC mechanism shall be configurable	SWS_V2xGn_NA_1
SRS_V2X_00242	The V2X system shall manage CAM generation such that no CAM messages will be dropped	SWS_V2xGn_NA_1
SRS_V2X_00245	The V2X system shall support per-packet transmission power control	SWS_V2xGn_NA_1
SRS_V2X_00259	The V2X system shall manage the life time of all DENM packets	SWS_V2xGn_20259
SRS_V2X_00291	The V2X system shall only send messages with valid postion and time	SWS_V2xGn_NA_2
SRS_V2X_00301	The V2X system's Facility Layer shall handle DENM repetition	SWS_V2xGn_NA_2
SRS_V2X_00318	The V2X system's Facility Layer shall generate traces and path histories	SWS_V2xGn_NA_2
SRS_V2X_00322	The V2X system shall provide services to avoid channel congestion of the shared media	SWS_V2xGn_NA_1
SRS_V2X_00323	The V2X system shall provide mitigation techniques to avoid disturbing other services operating at nearby frequencies	SWS_V2xGn_NA_4
SRS_V2X_00391	The V2X system's access layer shall be ITS-G5 compliant	SWS_V2xGn_20414, SWS_V2xGn_NA_1
SRS_V2X_00405	The V2X basic system shall support services for confidentiality	SWS_V2xGn_NA_3
SRS_V2X_00406	The end-to-end security envelope shall be generated depending on the message type	SWS_V2xGn_NA_3
SRS_V2X_00407	The signature in the end-to-end security envelope shall be generated using a private key corresponding to a valid	SWS_V2xGn_NA_3

	authorization ticket (pseudonym certificate)	
SRS_V2X_00412	The V2X system shall inform the driver about the expiration of the pseudonym certificates	SWS_V2xGn_NA_3
SRS_V2X_00413	The V2X system shall inform the driver about the expiration of the Long Term Certificates	SWS_V2xGn_NA_3
SRS_V2X_00451	The V2X system's access layer shall be compliant to the ETSI Harmonized Channel Specifications	SWS_V2xGn_NA_1
SRS_V2X_00511	The V2X system shall provide services for communication to multiple, geographically scattered and movable entities	SWS_V2xGn_NA_4
SRS_V2X_00531	The V2X system's Networking Layer shall support addressing based on geographic coordinates	SWS_V2xGn_20250, SWS_V2xGn_20251, SWS_V2xGn_20252, SWS_V2xGn_20255, SWS_V2xGn_20258, SWS_V2xGn_20414, SWS_V2xGn_20415
SRS_V2X_00693	The V2X system shall provide functionality for generating traces and path histories	SWS_V2xGn_NA_4
SRS_V2X_00711	The V2X system's CA basic service shall be compliant to ETSI Specification of Cooperative Awareness Basic Service	SWS_V2xGn_NA_2
SRS_V2X_00741	The V2X system's DEN basic service shall be compliant to ETSI Specifications of Decentralized Environmental Notification Basic Service	SWS_V2xGn_NA_2

Note:
Requirement IDs within this document have an encoding to state where each requirement has its origin:

- SWS items starting with a leading 0 (SWS_V2xM_0xxxx) are module specific and not inherited.
- SWS items starting with a leading 2 (SWS_V2xM_2xxxx) are inherited from C2C-CC Basic System Profile

7 Functional specification

7.1 General Functionality

[SWS_V2xGn_00012] [The V2xGn Module shall implement the GeoNetworking Protocol as defined in [18], [19] and [22] unless specified otherwise in this document.] (SRS_V2X_00161)

[SWS_V2xGn_00013] [The GeoNetworking Protocol shall support the GeoNetworking related requirements specified in [22]] ( )

[SWS_V2xGn_20250] [All default constants and parameters of the V2xGn module not defined or overwritten in the current document shall be set as specified in Annex G of [18].] (SRS_V2X_00531)

[SWS_V2xGn_20251] [The V2xGn module shall be implemented assuming the ETSI parameter itsGnSecurity is constantly set to ENABLED.] (SRS_V2X_00531, SRS_V2X_00160, SRS_V2X_00161)

[SWS_V2xGn_20252] [The V2xGn module shall only support anonymous address configuration mode.] (SRS_V2X_00531)

[SWS_V2xGn_20255] [The V2xGn module shall support geo-areas areas of up to 80 km2. In consequence, the itsGnMaxGeoAreaSize shall have a value of 80. It is configurable by the configuration option V2xGnItsGnMaxGeoAreaSize.] (SRS_V2X_00531)

[SWS_V2xGn_20414] [The V2xGn module shall be implemented with respect to the ETSI parameter itsGnIfType constantly set to ITS-G5.] (SRS_V2X_00531, SRS_V2X_00391)

[SWS_V2xGn_00130] [The V2xGn module shall get the pointer to the current time information via V2xM_GetRefTimePtr within the V2xGn_Init function.] ( )

7.2 GeoNetworking Packet Structure and Format

[SWS_V2xGn_00020] [The GeoNetworking protocol shall only support the packet header types Single Hop Broadcast packet header, GeoBroadcast packet headers and Beacon packet header.
] ( )

[SWS_V2xGn_20258] [The V2xGn module shall set the LifeTime field of all SHB packets to 1 second. Consequently, the multiplier bit of the LT field shall be set to 1 and the base bit of the LT field shall be set to 1.] (SRS_V2X_00531)

[SWS_V2xGn_20259] [The V2xGn module shall set the LifeTime field of all GBC packets to the value of the maxPacketLifetime from the transmit parameters

TxParams.The value of the LifeTime field shall not exceed the itsGnMaxPacketLifetime, specified in [18], Annex G.] (SRS_V2X_00259)

[SWS_V2xGn_20415] [itsGnMinPacketRepetitionInterval is not applicable (N/A).] (SRS_V2X_00531)

7.3 GeoNetworking Protocol Operations

7.3.1 Network Management

[SWS_V2xGn_00022] [The V2xGn module shall update the local position and time information. The minimum update frequency is configured by the configuration parameter `V2xGnItsGnMinUpdateFrequencyLPV`. The scheduled function `V2xGn_MainFunction()` shall be used for the cyclic update.] ( )

[SWS_V2xGn_00023] [The V2xGn module shall support GeoNetworking beaconing. The scheduled function `V2xGn_MainFunction()` shall be used for the cyclic beaconing.] ( )

[SWS_V2xGn_20269] [The V2xGn module shall only send beacons with the Position Accuracy Indicator (PAI) set to 1.] (SRS_V2X_00010)

[SWS_V2xGn_00081] [The V2xGn module shall support Location Table Maintainance. The scheduled function `V2xGn_MainFunction()` shall be used for the cyclic maintainance of the Location Table.] ( )

[SWS_V2xGn_00129] [The V2xGn module shall get the current position and time information via V2xM_GetPositionAndTime within the MainFunction.] ( )

7.3.2 Security Mechanisms

[SWS_V2xGn_00026] [The V2xGn module shall use security services provided by V2xM V2xM_V2xGn_ReqEncap and V2xM_V2xGn_ReqDecap.] (SRS_V2X_00160, SRS_V2X_00164)

[SWS_V2xGn_00028] [The V2xGn shall suspend transmission of messages and clear transmit buffers when a pseudonym changes is in preparation.] (SRS_V2X_00176)

Note: The V2xM will notify the V2xGn about pseudonym changes via V2xGn_V2xM_PreparePseudonymChange, V2xGn_V2xM_CommitPseudonymChange and V2xGn_V2xM_AbortPseudonymChange.

7.4 Message Forwarding

[SWS_V2xGn_20266] [The V2xGn module shall only support GeoBroadcast forwarding algorithms specified in [18] Annex E.3.] (SRS_V2X_00010)

[SWS_V2xGn_20267] [When forwarding packets, the V2xGn module shall use the DCC profile DP3 as defined in [23].] (SRS_V2X_00010)

[SWS_V2xGn_20169] [The V2xGn module shall check the distance from the sender position - in the security envelope, if available - and forward only messages with a distance from the sender of equal or less than 6 km.] (SRS_V2X_00010)

7.5 Message Transmission

[SWS_V2xGn_00034] [The V2xGn module shall provide the API `V2xGn_Transmit()` to enable transmit requests from the V2xBtp Module.] ( )

[SWS_V2xGn_00082] [
The V2xGn module shall use EthIf_ProvideTxBuffer to aquire a buffer within the Wireless Ethernet Driver for a V2X Packet to be transmitted. This has to be done during the V2xGn_Transmit context.]()

[SWS_V2xGn_00083] [
The V2xGn module shall provide transmission parameters to the Wireless Ethernet Driver for a V2X Packet to be transmitted via an API call to EthIf_SetBufWTxParams. This has to be done during the V2xGn_Transmit context.]()

[SWS_V2xGn_00035] [The V2xGn module shall transmits packets using the `EthIf_Transmit()` API provided by the EthIf Module. This has to be done during the V2xGn_Transmit context.] ( )

[SWS_V2xGn_00036] [If the configuration parameter V2xGnTxConfirmation is enabled, the V2xGn module shall provide information about the status of the transmission with an associated ID (generated by the V2xFac module and handed down to track the status of the packet) to the V2xBtp Module via the `V2xBtp_TxConfirmation()` callback.] ( )

[SWS_V2xGn_20260] [The V2xGn module shall buffer GBC packets when no neighbours are available (store-carry-forward). Consequently, the SCF bit of the TC (Traffic Class) field of GBC packets shall be set to 1.] (SRS_V2X_00010)

[SWS_V2xGn_20262] [The V2xGn module is not required to offload packets to another channel. Consequently, the channel offload bit of the TC (Traffic Class) field should be set to 0.] (SRS_V2X_00010)

[SWS_V2xGn_20263] [The V2xGn module shall only use the DCC profiles specified in [SWS_WEth_20235]. Consequently, the DCC Profile ID bits of the TC (Traffic Class) field shall only use the DPID values defined in **[SWS_WEth_20235]**.
] (SRS_V2X_00010)

[SWS_V2xGn_20264] ⌈ The V2xGn shall set the itsGnIsMobile bit of the Flags field to 1. ⌋ (SRS_V2X_00010)

[SWS_V2xGn_20265] ⌈ The V2xGn shall set the Maximum Hop Limit (MHL) field to 10. ⌋ (SRS_V2X_00010)

[SWS_V2xGn_20270] ⌈ All GeoNetworking frames sent by the V2xGn module shall use the EtherType value 0x8947 as listed by the IEEE Registration Authority at [24]. ⌋ (SRS_V2X_00010)

[SWS_V2xGn_20401] ⌈ The GN Source Address shall be constructed as follows:

- Set the field M (bit 0) to 0.
- Set the field ST (bits 1 to 5) to the station type of the ITS-S. The station type in the GN source address shall be identical to the station type in CAMs/DENMs.
- Set all bits of the field SCC (bits 6 to 15) to 0.
- Set the field MID (bits 16 to 63) to the value of the MAC address.

⌋ (SRS_V2X_00010)

[SWS_V2xGn_00128] ⌈ The V2xGn module shall call V2xM_GetGlobalTxParams that delivers own channel CBR information set in the GeoNetworking header to be transmitted. ⌋ ( )

7.6 Message Reception

[SWS_V2xGn_00038] ⌈ The V2xGn module shall create a unique TransactionId for each received packet. This TransactionId is handed up to track the received packets and is used for verification on demand. ⌋ ( )

[SWS_V2xGn_00039] ⌈ The V2xGn module shall indicate received packets via the `V2xBtp_RxIndication()` callback to the BTP module. ⌋ ( )

[SWS_V2xGn_00084] ⌈ The V2xGn module shall get the reception status of a received packet during the `V2xGn_RxIndication()` from the EthIf module with a call to `EthIf_GetBufWRxParams()`. ⌋ ( )

[SWS_V2xGn_20268] ⌈ The V2xGn module shall only use duplicate packet detection as specified in [18] Annex A.2 and A.3. ⌋ (SRS_V2X_00010)

[SWS_V2xGn_20181] ⌈ If the V2xGn module detects a collision of the least significant 32 bit of the pseudonym identifier of type HashedId8 with the pseudonym identifier of another ITS station, it shall change the pseudonym if the certificate corresponding to the other pseudonym ID is valid. ⌋ (SRS_V2X_00010)

[SWS_V2xGn_00127] ⌈ The V2xGn module shall call V2xM_SetGlobalRxParams with CBR information extracted from the GeoNetworking header. ⌋ ( )

[SWS_V2xGn_00131] [The V2xGn module shall use V2xM_CalcDistance when calculations of geographical distances are necessary for the V2xGn protocol operations.] ( )

7.7 Error classification

This chapter lists and classifies all errors that can be detected within this software module. Each error is classified according to relevance (development / production) and related error code. For development errors, a value is defined.

7.7.1 Development Errors

[SWS_V2xGn_00041] The following table lists development errors that shall be distinguished by the V2xGn module. V2xGn shall report them to the DET, if development error detection is enabled.
[

Type of error	*Related error code*	*Value [hex]*
API service called with invalid parameter	V2XGN_E_PARAM	0x01
API service called with invalid pointer	V2XGN_E_PARAM_POINTER	0x02
API service used without module initialization	V2XGN_E_UNINIT	0x03
API service called with invalid configuration pointer	V2XGN_E_INIT_FAILED	0x04

] ( )

7.7.2 Runtime Errors

There are no runtime errors.

7.7.3 Transient Faults

There are no transient faults.

7.7.4 Production Errors

There are no production errors.

7.7.5 Extended Production Errors

There are no extended production errors.

8 API specification

8.1 Imported types

In this chapter all types included from the following files are listed:

[SWS_V2xGn_00042] [

Module	*Imported Type*
ComStack_Types	BufReq_ReturnType
Eth_GeneralTypes	Eth_BufIdxType
	Eth_FrameType
Std_Types	Std_ReturnType
	Std_VersionInfoType
V2xBtp	V2xBtp_RxParamsType
V2x_GeneralTypes	V2x_ChanType
	V2x_GnAddressType
	V2x_GnDestinationAreaType
	V2x_GnDestinationType
	V2x_GnLocalPositionVectorType
	V2x_GnPacketTransportType
	V2x_GnTxResultType
	V2x_GnUpperProtocolType
	V2x_PseudonymType
	V2x_SecProfileType
	V2x_SecReportType
	V2x_SecReturnType
	V2x_TrafficClassIdType
WEth_GeneralTypes	WEth_BufWRxParamIdType
	WEth_BufWTxParamIdType

] ()

8.2 Type definitions

8.2.1 V2xGn_ConfigType

[SWS_V2xGn_00043] [

Name:	`V2xGn_ConfigType`	
Type:	`Structure`	
Range:	`implementation specific`	The content of the configuration data structure is implementation specific.
Description:	Configuration data structure of the V2xGn module.	

] ()

8.2.2 V2xGn_TxParamsType

[SWS_V2xGn_00063] [

Name:	`V2xGn_TxParamsType`		
Type:	`Structure`		
Element:	`V2x_GnUpperProtocolType`	`upperProtocol`	The protocol which

			triggered the request. (e.g. BTP-A or BTP-B)
	V2x_GnPacketTransportType	transportType	Specifies the packet transportation type
	V2x_GnAddressType	destinationAddress	Destination address for GeoUnicast packet
	V2x_GnDestinationAreaType	destinationArea	Destination area for GeoBroadcast/GeoAnycast packet.
	V2x_GnDestinationType	destinationType	Select which destination type (destinationAddress or destinationArea is used for this packet).
	V2x_SecProfileType	secProfile	Parameters depending on the security service.
	uint16	maxPacketLifetime	Time a packet can be buffered until it reaches the destination.
	V2x_TrafficClassIdType	trafficClassId	Set of parameter specifying the traffic class for the message.
Description:	Structure containing parameters for the V2xGn_Transmit() API.		

8.3] () Function definitions

8.3.1 V2xGn_Init

[SWS_V2xGn_00068] [

Service name:	V2xGn_Init

Syntax:	`void V2xGn_Init(`
	`  void`
	`)`

Service ID[hex]:	0x01
Sync/Async:	Synchronous
Reentrancy:	Non Reentrant
Parameters (in):	None
Parameters (inout):	None
Parameters (out):	None
Return value:	None
Description:	Initializes the V2xGn module.

] ()

8.3.2 V2xGn_GetVersionInfo

[SWS_V2xGn_00069] [

Service name:	V2xGn_GetVersionInfo

Syntax:	`void V2xGn_GetVersionInfo(`
	`  Std_VersionInfoType* VersionInfoPtr`
	`)`

Service ID[hex]:	0x02
Sync/Async:	Synchronous

Reentrancy:	Reentrant	
Parameters (in):	None	
Parameters (inout):	None	
Parameters (out):	VersionInfoPtr	Pointer to where to store the version information of this module.
Return value:	None	
Description:	Returns the version information of this module.	

] () **[SWS_V2xGn_00090]** [
If development error detection is enabled: the function shall check the parameter VersionInfoPtr for being valid. If the check fails, the function shall raise the development error V2XGN_E_PARAM_POINTER.]()

8.3.3 V2xGn_V2xM_PreparePseudonymChange

[SWS_V2xGn_00072] [

Service name:	V2xGn_V2xM_PreparePseudonymChange	
Syntax:	`Std_ReturnType V2xGn_V2xM_PreparePseudonymChange( const V2x_PseudonymType* PseudonymPtr )`	
Service ID[hex]:	0x05	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	PseudonymPtr	The Pseudonym provided by V2xM
Parameters (inout):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: operation successful E_NOT_OK: operation failed
Description:	This function is called by the V2xM when a Pseudonym Change occurs to prepare the change in every module using it.	

] () **[SWS_V2xGn_00091]** [
The function V2xGn_V2xM_PreparePseudonymChange shall prepare the setting of the pseudonym specific part of the GeoNetworking Address being used for packet transmission.](SRS_V2X_00176)

[SWS_V2xGn_00092] [
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT otherwise (if DET is disabled) return E_NOT_OK.]()

[SWS_V2xGn_00093] [
If development error detection is enabled: the function shall check the parameter PseudonymPtr for being valid. If the check fails, the function shall raise the development error V2XGN_E_PARAM_POINTER otherwise (if DET is disabled) return E_NOT_OK.]()

Note: This starts a module internal transaction for the pseudonym change. The actual pseudonym change becomes effective only after an API call to V2xGn_V2xM_CommitPseudonymChange.

8.3.4 V2xGn_V2xM_CommitPseudonymChange

[SWS_V2xGn_00111] ⌈

Service name:	V2xGn_V2xM_CommitPseudonymChange	
Syntax:	`Std_ReturnType V2xGn_V2xM_CommitPseudonymChange( void )`	
Service ID[hex]:	0x09	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	None	
Parameters (inout):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: operation successful E_NOT_OK: operation failed
Description:	This function is called by the V2xM when all modules are OK with the pseudonym change and the change is to be committed.	

⌋ () **[SWS_V2xGn_00112]** ⌈
The function V2xGn_V2xM_CommitPseudonymChange shall update the pseudonym specific part of the module's GeoNetworking Address. ⌋(SRS_V2X_00176)

[SWS_V2xGn_00113] ⌈
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT otherwise (if DET is disabled) return E_NOT_OK. ⌋()

Note: The function requires previous preparation of the pseudonym via an API call to V2xGn_V2xM_PreparePseudonymChange.

8.3.5 V2xGn_V2xM_AbortPseudonymChange

[SWS_V2xGn_00126] ⌈

Service name:	V2xGn_V2xM_AbortPseudonymChange	
Syntax:	`Std_ReturnType V2xGn_V2xM_AbortPseudonymChange( void )`	
Service ID[hex]:	0x0a	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	None	
Parameters (inout):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: operation successful E_NOT_OK: operation failed
Description:	This function is called by the V2xM when not all modules are OK with the pseudonym change and the change is to be rolled back.	

⌋ () **[SWS_V2xGn_00115]** ⌈

The function V2xGn_V2xM_AbortPseudonymChange shall set the state of the module to the state before the pseudonym change has been prepared.](SRS_V2X_00176)

[SWS_V2xGn_00116] [
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT otherwise (if DET is disabled) return E_NOT_OK.]()

Note: The function requires previous preparation of the pseudonym via an API call to V2xGn_V2xM_PreparePseudonymChange.

8.3.6 V2xGn_Transmit

[SWS_V2xGn_00070] [

Service name:	V2xGn_Transmit	

Syntax:	`V2x_GnTxResultType V2xGn_Transmit(`
	`  uint16 TransactionId16,`
	`  const V2xGn_TxParamsType* TxParams,`
	`  uint16 Length`
	`)`	

Service ID[hex]:	0x03	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	TransactionId16	Transaction Id for the Packet
	TxParams	Additional transmission parameters
	Length	Length of the user data
Parameters (inout):	None	
Parameters (out):	None	
Return value:	V2x_GnTxResultType	Values specified in the Type could be returned. V2X_GNTX_ACCEPTED if no error occurred. V2X_GNTX_ACCEPTED transmit has been accepted V2X_GNTX_E_MAXSDUSIZEOVFL transmit has been rejected due to maximum length exceedance V2X_GNTX_E_MAXPACKETLIFETIME transmit has been rejected due to maximum lifetime exceedance V2X_GNTX_E_TCID transmit has been rejected due to unsupported Traffic Class ID V2X_GNTX_E_MAXGEOAREASIZE transmit has been rejected due to GeoArea exceeds max size V2X_GNTX_E_UNSPECIFIED transmit has been rejected due to unspecified reasons
Description:	Is called by V2x_Btp to send a message.	

] () **[SWS_V2xGn_00095]** [
The function V2xGn_Transmit shall transmit a V2X Packet.]()

[SWS_V2xGn_00096] [
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT otherwise (if DET is disabled) return V2X_GNTX_E_UNSPECIFIED.]()

[SWS_V2xGn_00098] ⌈
The function shall return V2X_GNTX_E_MAXSDUSIZEOVFL if the call to EthIf_ProvideTxBuffer returns BUFREQ_E_OVFL. ⌋()

[SWS_V2xGn_00099] ⌈
The function shall return V2X_GNTX_E_MAXPACKETLIFETIME if the parameter TxParams.maxPacketLifeTime is invalid. ⌋()

[SWS_V2xGn_00100] ⌈
The function shall return V2X_GNTX_E_TCID if the parameter TxParams.trafficClass is invalid. ⌋()

[SWS_V2xGn_00101] ⌈
The function shall return V2X_GNTX_E_MAXGEOAREASIZE if the parameter TxParams.destinationType is V2X_GNDESTINATION_AREA and the parameter TxParams.destinationArea is invalid. ⌋()

8.4 Call-back notifications

This is a list of functions provided for other modules. The function prototypes of the callback functions shall be provided in the file V2xGn_Cbk.h.

8.4.1 V2xGn_V2xM_EncapConfirmation

[SWS_V2xGn_00118] ⌈

Service name:	V2xGn_V2xM_EncapConfirmation	

Syntax:	`void V2xGn_V2xM_EncapConfirmation(`
	`  uint16 TransactionId16`
	`)`	

Service ID[hex]:	0x0b	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	TransactionId16	The TransactionId of the encapsulated packet
Parameters (inout):	None	
Parameters (out):	None	
Return value:	None	
Description:	This function is called by the V2xM when an encapsulation has been finished.	

⌋() **[SWS_V2xGn_00119]** ⌈
The function V2xGn_V2xM_EncapConfirmation shall finalize the packet transmission by transmitting the packet to the lower layer. ⌋()

[SWS_V2xGn_00120] ⌈
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT. ⌋()

Note: The function requires previous successful transmission request via the API V2xGn_Transmit.

8.4.2 V2xGn_V2xM_DecapConfirmation

[SWS_V2xGn_00122] ⌈

Service name:	V2xGn_V2xM_DecapConfirmation	

Syntax:	`void V2xGn_V2xM_DecapConfirmation(`
	`  uint32 TransactionId32,`
	`  V2x_SecReportType SecReport,`
	`  uint64 CertificateId,`
	`  uint32 ItsAid,`
	`  uint8 SspLength,`
	`  uint8* SspBits`
	`)`	

Service ID[hex]:	0x0c	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	TransactionId32	The TransactionId32 of the decapsulated packet
	SecReport	The security report.
	CertificateId	The identification of the used for verification (by certificate hash)
	ItsAid	The numerical value of the ITS-AID
	SspLength	The length (in octets, up to 31) of the SSP bits
	SspBits	The SSP bits
Parameters (inout):	None	
Parameters (out):	None	
Return value:	None	
Description:	This function is called by the V2xM when a decapsulation has been finished.	

⌋ () **[SWS_V2xGn_00123]** ⌈
The function V2xGn_V2xM_DecapConfirmation shall continue the processing of a received packet by proceeding with V2xGn protocol operations. ⌋()

[SWS_V2xGn_00124] ⌈
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT. ⌋()

Note: The function requires previous successful reception of a packet via the API V2xGn_RxIndication.

8.4.3 V2xGn_RxIndication

[SWS_V2xGn_00071] ⌈

Service name:	V2xGn_RxIndication

Syntax:	`void V2xGn_RxIndication(`
	`  uint8 CtrlIdx,`
	`  Eth_FrameType FrameType,`
	`  boolean IsBroadcast,`
	`  const uint8* PhysAddrPtr,`
	`  uint8* DataPtr,`
	`  uint16 LenByte`

	)	
Service ID[hex]:	0x04	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	CtrlIdx	Index of the Ethernet controller within the context of the Ethernet Interface
	FrameType	frame type of received Ethernet frame
	IsBroadcast	parameter to indicate a broadcast frame
	PhysAddrPtr	pointer to Physical source address (MAC address in network byte order) of received Ethernet frame
	DataPtr	Pointer to payload of the received Ethernet frame (i.e. Ethernet header is not provided).
	LenByte	Length of received data.
Parameters (inout):	None	
Parameters (out):	None	
Return value:	None	
Description:	Indicates the reception of an Ethernet frame	

] () **[SWS_V2xGn_00103]** [
The function V2xGn_RxIndication shall get reception parameters of the Wireless Ethernet Driver for a V2X Packet received via an API call to EthIf_GetBufWRxParams.]()
This is done to get access to the wireless specific reception parameters (e.g. the RSSI or the TransactionId32) of the packet that is not available through the V2xGn_RxIndication API.

[SWS_V2xGn_00104] [
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT.]()

[SWS_V2xGn_00105] [
If development error detection is enabled: the function shall check the parameter DataPtr for being valid. If the check fails, the function shall raise the development error V2XGN_E_PARAM_POINTER.]()

8.4.4 V2xGn_TxConfirmation

[SWS_V2xGn_00074] [

Service name:	V2xGn_TxConfirmation

Syntax:	`void V2xGn_TxConfirmation(`
	`  uint8 CtrlIdx,`
	`  uint8 BufIdx`
	`)`

Service ID[hex]:	0x07
Sync/Async:	Synchronous
Reentrancy:	Non Reentrant
Parameters (in):	CtrlIdx: Index of the Ethernet controller within the context of the Ethernet Interface
	BufIdx: Index of the buffer resource
Parameters (inout):	None

Parameters (out):	None
Return value:	None
Description:	Confirms the transmission of an Ethernet frame

] () **[SWS_V2xGn_00107]** [
If development error detection is enabled: the function shall check that the service V2xGn_Init was previously called. If the check fails, the function shall raise the development error V2XGN_E_UNINIT.]()

8.5 Scheduled functions

8.5.1 V2xGn_MainFunction

[SWS_V2xGn_00075] [

Service name:	V2xGn_MainFunction

Syntax:	`void V2xGn_MainFunction(`
	`  void`
	`)`

Service ID[hex]:	0x08
Description:	Main function of the V2xGn module for periodical execution of protocol operations.

] ()

8.6 Expected Interfaces

In this chapter all external interfaces required from other modules are listed.

8.6.1 Mandatory Interfaces

This chapter defines all external interfaces which are required to fulfill the core functionality of the module.

[SWS_V2xGn_00076] [

API function	***Description***
EthIf_GetBufWRxParams	Read out values related to the receive direction of the transceiver for a received packet. For example, this could be RSSI or Channel belonging to one single packet.
EthIf_GetBufWTxParams	Read out values related to the transmit direction of the transceiver for a transmitted packet. For example, this could be transaction ID belonging to one single packet.
EthIf_ProvideTxBuffer	Provides access to a transmit buffer of the specified Ethernet controller.
EthIf_SetBufWTxParams	Set values related to the transmit direction of the transceiver for a specific buffer (packet to be sent). For example, this can be the desired transmit power or the channel belonging to one single packet.
EthIf_Transmit	Triggers transmission of a previously filled transmit buffer
V2xBtp_RxIndication	Via this API, the V2xBtp module gets the data (BTP-PDU), the GeoNetworking parameters and the Transaction ID of a received GeoNetworking packet.
V2xM_CalcDistance	Calculates the distance between two geographical points on earth

	with the assumption that they are on elevation 0.
V2xM_GetPositionAndTime	Provides the instantaneous position information.
V2xM_GetRefTimePtr	Provides a pointer to the time reference of the V2X-Stack.
V2xM_TriggerPseudonymChange	This function is called by the V2xFac, V2xGn or another entity to change the Pseudonym used by the V2X-Stack, e.g. due to a GeoNetworking address conflict.
V2xM_V2xGn_GetGlobalTxParams	This function is called by V2xGn to get the current channel busy percentage for the specified channel
V2xM_V2xGn_ReqDecap	This function is called by the V2xGn to decrypt and verify a message. An asynchronous V2xGn_V2xM_DecapConfirmation call will be used to notify V2xGn of the result.
V2xM_V2xGn_ReqEncap	This function is called by the V2xGn to sign and/or encrypt a message. An asynchronous V2xGn_V2xM_EncapConfirmation call will be used to notify V2xGn of the result.
V2xM_V2xGn_SetGlobalRxParams	This function is called by V2xGn to set the current channel busy percentage for the specified channel

] ()

8.6.2 Optional Interfaces

This chapter defines all external interfaces which are required to fulfill an optional functionality of the module.

[SWS_V2xGn_00077] [

API function	*Description*
Det_ReportError	Service to report development errors.
V2xBtp_TxConfirmation	By this API primitive, the V2xBtp module gets an indication from the V2xGn module about the status of the transmission of the data (FAC-PDU) with the associated ID.

] ()

9 Sequence diagrams

The following sequence diagrams show the interactions between the V2xGn module and its adjacent modules.

Please note that the sequence diagrams are an extension for illustrational purposes to ease understanding of the specification and to support the functional specification described in chapter 7 and API specification described in chapter 8.

Note that all parameters and return types are left out to make the diagrams easier to read and understand.

9.1 V2xGn_RxIndication

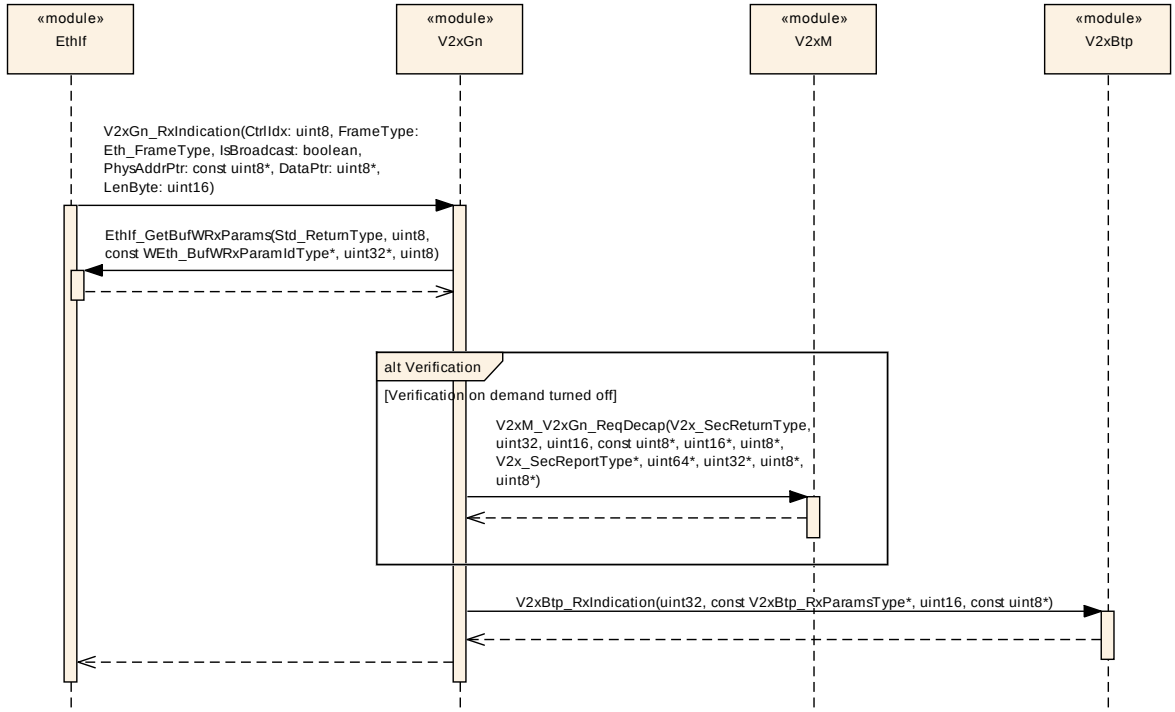

Figure 3 –V2xGn_RxIndication

9.2 V2xGn_Transmit

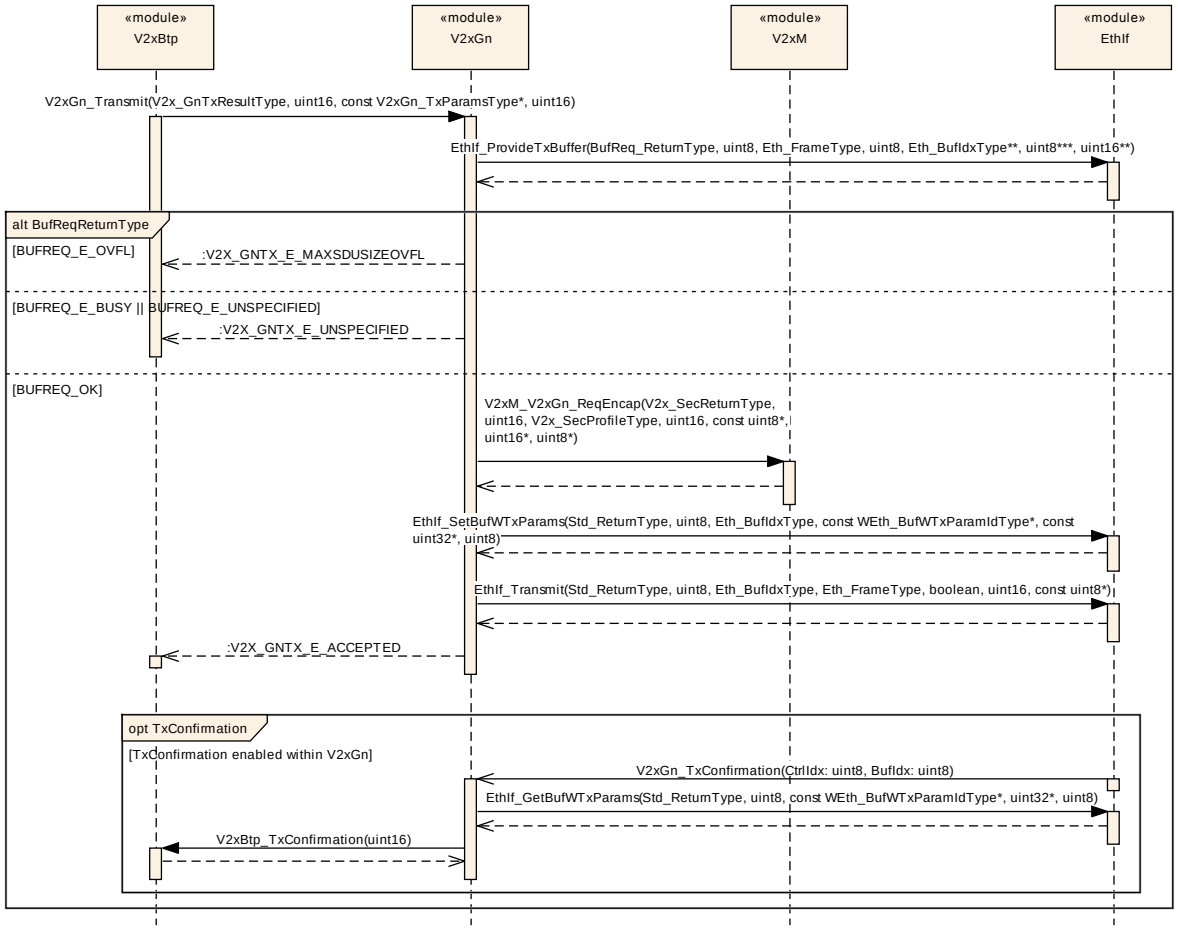

Figure 4 – V2xGn_Transmit

9.3 V2xGn_V2xM_UpdatePseudonym

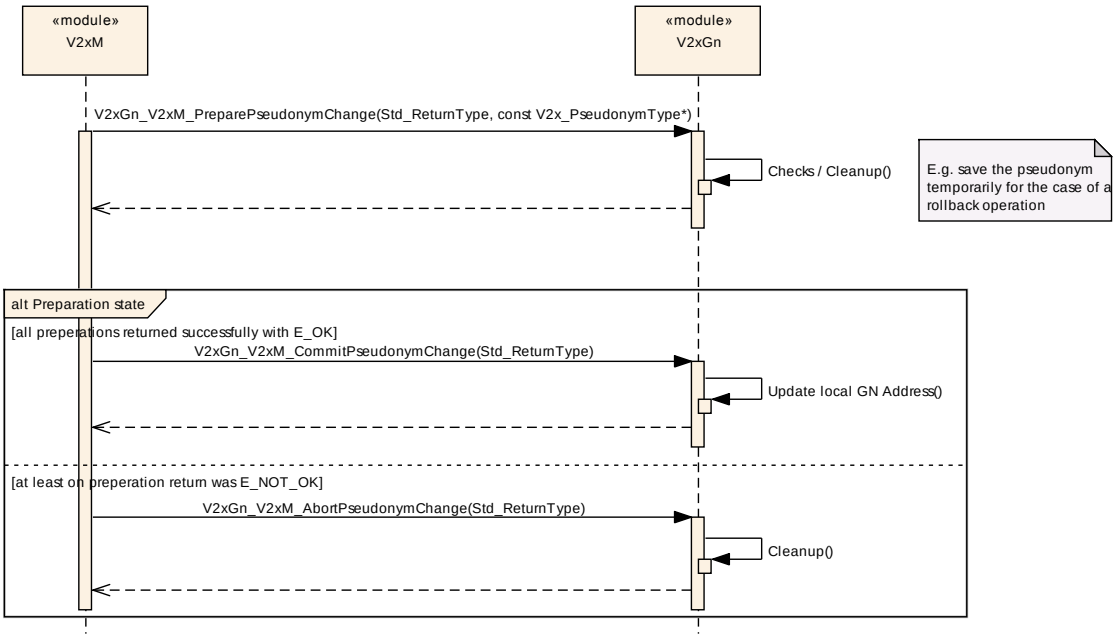

Figure 5 – V2xGn_V2xM_UpdatePseudonym

9.4 V2xGn_MainFunction

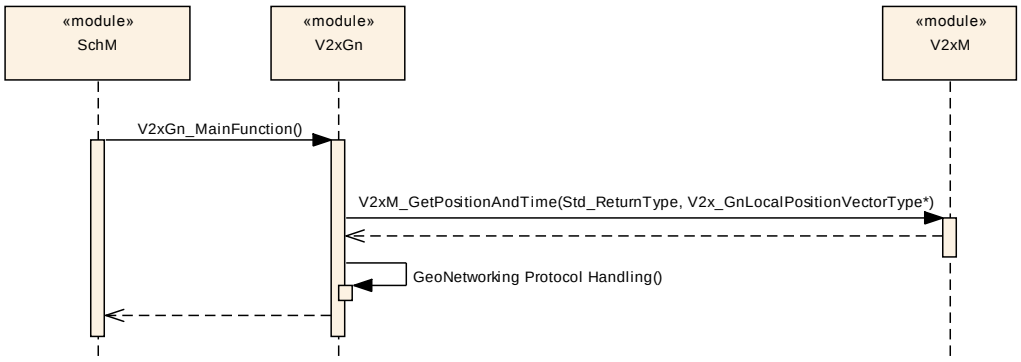

Figure 6 – V2xGn_MainFunction

10 Configuration specification

Chapter 10.1 specifies the structure (containers) and the parameters of the module V2xGn.

Chapter 10.2 specifies additionally published information of the module V2xGn.

10.1 Containers and configuration parameters

The following chapters summarize all configuration parameters. The detailed meanings of the parameters are described in chapter 7 and chapter 8.

10.1.1 Variants

[SWS_V2xGn_00078] [The V2xGn module only supports VARIANT-PRE-COMPILE] (SRS_BSW_00345)

10.1.2 V2xGn

10.1.3 V2xGn

SWS Item	**ECUC_V2xGn_00001 :**
Module Name	*V2xGn*
Module Description	Configuration of the V2xGn (Vehicle-2-X Geo Networking) module.
Post-Build Variant Support	false
Supported Config Variants	VARIANT-PRE-COMPILE

Included Containers		
Container Name	***Multiplicity***	***Scope / Dependency***
V2xGnBeaconService	1	This container contains the GeoNetworking configuration parameters related to the beacon service.
V2xGnGeneral	1	This container specifies the general configuration parameters of the V2xGn module.
V2xGnPacketForwarding	1	This container contains the GeoNetworking configuration parameters related to packet forwarding.

10.1.4 V2xGnGeneral

10.1.5 V2xGnGeneral

SWS Item	**ECUC_V2xGn_00002 :**
Container Name	V2xGnGeneral
Description	This container specifies the general configuration parameters of the V2xGn module.
Configuration Parameters	

SWS Item	**ECUC_V2xGn_00006 :**		
Name	V2xGnDevErrorDetect		
Parent Container	V2xGnGeneral		
Description	Switches the Default Error Tracer (Det) detection and notification ON or OFF. ▪ true: enabled (ON) ▪ false: disabled (OFF)		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00016 :**		
Name	V2xGnItsGnLifetimeLocTE		
Parent Container	V2xGnGeneral		
Description	Location table maintenance: Lifetime of an entry in the location table in [s]		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65535]		
Default value	20		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local dependency: shall be a multiple of the V2xGnMainFunctionPeriod		

SWS Item	**ECUC_V2xGn_00009 :**		
Name	V2xGnItsGnLocalGnAddr		
Parent Container	V2xGnGeneral		
Description	64bit GeoNetworking Address.		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 18446744073709551615		
Default value	1		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00014 :**		
Name	V2xGnItsGnMaxGeoNetworkingHeaderSize		
Parent Container	V2xGnGeneral		
Description	Maximum size of GeoNetworking header in [Byte].		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65535		
Default value	88		

Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00013 :**		
Name	V2xGnItsGnMaxSduSize		
Parent Container	V2xGnGeneral		
Description	Maximum size of GN-SDU in [Byte].		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65535		
Default value	1398		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00011 :**		
Name	V2xGnItsGnMinUpdateFrequencyLPV		
Parent Container	V2xGnGeneral		
Description	Minimum update frequency of local position vector (LPV) in [s].		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65535]		
Default value	--		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local dependency: shall be a multiple of the V2xGnMainFunctionPeriod		

SWS Item	**ECUC_V2xGn_00012 :**		
Name	V2xGnItsGnPaiInterval		
Parent Container	V2xGnGeneral		
Description	Distance related to the confidence interval for latitude and longitude [m]. Used to determine the PAI.		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 100		
Default value	80		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00008 :**
Name	V2xGnItsGnProtocolVersion
Parent Container	V2xGnGeneral
Description	GeoNetworking protocol version
Multiplicity	1
Type	EcucIntegerParamDef

Range	0 .. 255		
Default value	--		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00017 :**	
Name	V2xGnItsGnSnDecapResultHandling	
Parent Container	V2xGnGeneral	
Description	Indicates the handling of the V2xM_ReqDecap result code.	
Multiplicity	1	
Type	EcucEnumerationParamDef	
Range	V2XGN_NON_STRICT_SEC_HANDLING	GN packets that are not correctly verified and decrypted can be passed to the upper protocol entity for further processing.
	V2XGN_STRICT_SEC_HANDLING	Received GN packets that are not correctly verified and decrypted are always dropped.
Post-Build Variant Value	false	
Value Configuration Class	***Pre-compile time***	X All Variants
	Link time	--
	Post-build time	--
Scope / Dependency	scope: local	

SWS Item	**ECUC_V2xGn_00015 :**	
Name	V2xGnItsGnStationType	
Parent Container	V2xGnGeneral	
Description	Station Type used in GeoNetworking protocol, RoadSideUnit (15) not supported by AUTOSAR.	
Multiplicity	1	
Type	EcucEnumerationParamDef	
Range	V2XFAC_ST_BUS	--
	V2XFAC_ST_CYCLIST	--
	V2XFAC_ST_HEAVYTRUCK	--
	V2XFAC_ST_LIGHTTRUCK	--
	V2XFAC_ST_MOPED	--
	V2XFAC_ST_MOTORCYCLE	--
	V2XFAC_ST_PASSENGERCAR	--
	V2XFAC_ST_PEDESTRIAN	--
	V2XFAC_ST_SPECIALVEHICLES	--
	V2XFAC_ST_TRAILER	--
	V2XFAC_ST_TRAM	--
	V2XFAC_ST_UNKNOWN	--
Post-Build Variant Value	false	
Value Configuration Class	***Pre-compile time***	X All Variants
	Link time	--
	Post-build time	--
Scope / Dependency	scope: local	

SWS Item	**ECUC_V2xGn_00018 :**		
Name	V2xGnMainFunctionPeriod		
Parent Container	V2xGnGeneral		
Description	Specifies the period of main function V2xGn_MainFunction in seconds. V2xGn does not require this information but the BSW scheduler.		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. INF[		
Default value	0.001		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00007 :**		
Name	V2xGnTxConfirmation		
Parent Container	V2xGnGeneral		
Description	When enabled, transmission status information will be forwarded to the upper layer. ▪ true: enabled (ON) ▪ false: disable (OFF)		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: ECU		

SWS Item	**ECUC_V2xGn_00005 :**		
Name	V2xGnVersionInfoApi		
Parent Container	V2xGnGeneral		
Description	Enable/disables the API for reading the version information of the V2xGn Module. ▪ true: enabled (ON) ▪ false: disabled (OFF)		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00019 :**
Name	V2xGnEthIfRef
Parent Container	V2xGnGeneral
Description	This represents the reference to the Ethernet interface taken to transmit

	the V2X-Packets to.		
Multiplicity	1		
Type	Symbolic name reference to [EthIfController]		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

No Included Containers

10.1.6 V2xGnBeaconService

10.1.7 V2xGnBeaconService

SWS Item	**ECUC_V2xGn_00003 :**
Container Name	V2xGnBeaconService
Description	This container contains the GeoNetworking configuration parameters related to the beacon service.
Configuration Parameters	

SWS Item	**ECUC_V2xGn_00021 :**		
Name	V2xGnItsGnBeaconServiceMaxJitter		
Parent Container	V2xGnBeaconService		
Description	Maximum beacon jitter [s]. The Jitter is used for the beacon retransmission. The actual jitter value is a random number between 0 and V2xGnItsGnBeaconServiceMaxJitter. The function introduces a random component for the timer to avoid synchronization issues among GeoAdhoc routers.		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0.001 .. INF]		
Default value	0.75		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00020 :**		
Name	V2xGnItsGnBeaconServiceRetransmitTimer		
Parent Container	V2xGnBeaconService		
Description	Duration of Beacon service retransmit timer [s].		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0.001 .. INF]		
Default value	3		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

	dependency: shall be a multiple of the V2xGnMainFunctionPeriod.

No Included Containers

10.1.8 V2xGnPacketForwarding

10.1.9 V2xGnPacketForwarding

SWS Item	**ECUC_V2xGn_00004 :**
Container Name	V2xGnPacketForwarding
Description	This container contains the GeoNetworking configuration parameters related to packet forwarding.
Configuration Parameters	

SWS Item	**ECUC_V2xGn_00032 :**		
Name	V2xGnItsGnBcForwardingPacketBufferSize		
Parent Container	V2xGnPacketForwarding		
Description	Size of BC forwarding packet buffer [Byte].		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 18446744073709551615		
Default value	1024000		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00033 :**		
Name	V2xGnItsGnCbfPacketBufferSize		
Parent Container	V2xGnPacketForwarding		
Description	Size of CBF packet buffer [Byte]		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 18446744073709551615		
Default value	256000		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00022 :**	
Name	V2xGnItsGnDefaultHopLimit	
Parent Container	V2xGnPacketForwarding	
Description	Default hop limit indicating the maximum number of hops a packet travels.	
Multiplicity	1	
Type	EcucIntegerParamDef	
Range	0 .. 255	

Default value	10		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00030 :**		
Name	V2xGnItsGnDefaultMaxCommunicationRange		
Parent Container	V2xGnPacketForwarding		
Description	Default theoretical maximum communication range [m]		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65535		
Default value	1000		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00024 :**		
Name	V2xGnItsGnDefaultPacketLifetime		
Parent Container	V2xGnPacketForwarding		
Description	Default packet lifetime [s].		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 6300]		
Default value	60		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00034 :**		
Name	V2xGnItsGnDefaultTrafficClass		
Parent Container	V2xGnPacketForwarding		
Description	Forwarding: Default traffic class		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 255		
Default value	0		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00031 :**
Name	V2xGnItsGnGeoAreaLineForwardingUsed
Parent Container	V2xGnPacketForwarding
Description	Forwarding of GBC/GAC packet if GeoAdhoc router is located outside the destination GeoArea.

	▪ true: enabled (ON) ▪ false: disabled (OFF)		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00029 :**		
Name	V2xGnItsGnGeoBroadcastCbfMaxTime		
Parent Container	V2xGnPacketForwarding		
Description	Maximum duration a GeoBroadcast packet shall be buffered in the CBF packet buffer [s]		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. INF[		
Default value	0.001		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00028 :**		
Name	V2xGnItsGnGeoBroadcastCbfMinTime		
Parent Container	V2xGnPacketForwarding		
Description	Minimum duration a GeoBroadcast packet shall be buffered in the CBF packet buffer [s]		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. INF[		
Default value	0.001		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00027 :**	
Name	V2xGnItsGnMaxGeoAreaSize	
Parent Container	V2xGnPacketForwarding	
Description	Maximum size of the geographical area for a GBC and GAC packet [km2]. If the geographical area size exceeds the maximum value, the GeoNetworking packet shall not be sent (source) and not be forwarded (forwarder).	
Multiplicity	1	
Type	EcucIntegerParamDef	
Range	0 .. 18446744073709551615	
Default value	80	
Post-Build Variant Value	false	

Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00025 :**		
Name	V2xGnItsGnMaxPacketDataRate		
Parent Container	V2xGnPacketForwarding		
Description	Maximum packet data rate for a GeoAdhoc router [Byte/s]. If the mean (EMA) packet data rate a of a GeoAdhoc router exceeds the value, packets from this GeoAdhoc router (source or sender) are not forwarded.		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 4294967295		
Default value	100000		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00026 :**		
Name	V2xGnItsGnMaxPacketDataRateEmaBeta		
Parent Container	V2xGnPacketForwarding		
Description	Weight factor for the Exponential Moving Average of the packet data rate PDR in percent.		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. 1]		
Default value	0.9		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

SWS Item	**ECUC_V2xGn_00023 :**		
Name	V2xGnItsGnMaxPacketLifetime		
Parent Container	V2xGnPacketForwarding		
Description	Upper limit of the maximum lifetime [s]		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 6300]		
Default value	600		
Post-Build Variant Value	false		
Value Configuration Class	***Pre-compile time***	X	All Variants
	Link time	--	
	Post-build time	--	
Scope / Dependency	scope: local		

No Included Containers

10.2 Published Information

For details refer to the chapter 10.3 “Published Information” in [3].

11 Not applicable requirements

[SWS_V2xGn_NA_1] ⌈ This requirement references all not applicable access layer requirements ⌋ (SRS_V2X_00451, SRS_V2X_00322, SRS_V2X_00242, SRS_V2X_00391, SRS_V2X_00232, SRS_V2X_00245, SRS_V2X_00239)

[SWS_V2xGn_NA_2] ⌈ This requirement references all not applicable facility layer requirements ⌋ (SRS_V2X_00711, SRS_V2X_00291, SRS_V2X_00318, SRS_V2X_00741, SRS_V2X_00301)

[SWS_V2xGn_NA_3] ⌈ This requirement references all not applicable security requirements ⌋ (SRS_V2X_00405, SRS_V2X_00413, SRS_V2X_00163, SRS_V2X_00412, SRS_V2X_00407, SRS_V2X_00406, SRS_V2X_00184, SRS_V2X_00174)

[SWS_V2xGn_NA_4] ⌈ This requirement references all not applicable other requirements from SRS V2X ⌋ (SRS_V2X_00190, SRS_V2X_00193, SRS_V2X_00207, SRS_V2X_00214, SRS_V2X_00693, SRS_V2X_00189, SRS_V2X_00323, SRS_V2X_00511)