Issue Report

License Number	Customer
CBD1400346	Nexteer Automotive Corporation Package: GMLAN 3.1 Micro: R7F701308 Compiler: GHS 2013.5.5

Maintenance Expiry Date

2014-08-01

SIP Delivery Date	SIP Version
2014-07-03	01.01.24

SLP	Delivery Number
GMLAN 3.1	D00

Report Creation Date

2014-07-11

Contact

In case of questions or the need for an update of the basic software delivery, please contact GMSupport@us.vector.com or your Vector contact person.

Table of Contents

1. Introduction
   - 1.1 Resolving Issues
   - 1.2 Issue Classification
2. New Issues
   - 2.1 Runtime Issues without Workaround: 2
   - 2.2 Runtime Issues with Workaround: 6
   - 2.3 Apparent Issues: 15
   - 2.4 Compiler Warnings: 10
3. New Issues for Information: 0
4. Report Legend
5. Quality Management Contact

1. Introduction

1.1 Resolving Issues

Reported issues are not necessarily fixed automatically by the next update delivery. If some of the reported issues shall be fixed, please contact Vector to establish an agreement about issues that shall be fixed in upcoming deliveries. Please note that Vector may fix additional issues without explicit request.

1.2 Issue Classification

This Issue Report provides issues that have been detected since the last report. The issues have been classified to facilitate the assessment of their impact:

The chapter 'New Issues' lists issues that have been detected since the last report and which could not be excluded based on the use-case defined in the questionnaire. The issues are classified as follows:

- **Runtime Issues without Workaround:** Runtime issues without a workaround require an update of the basic software delivery in case the issue affects the ECU overall functionality. The effect of an issue to the ECU functionality has to be analyzed by the customer as the basic software usage and its configuration is not known by Vector. The risk of change has also to be taken into account.

- **Runtime Issues with Workaround:** It is not recommended to update a delivery due to a runtime issue with a documented workaround. The effect of an issue to the ECU functionality has to be analyzed by the customer as the basic software usage and its configuration is not known by Vector. The risk of change has also to be taken into account.
- **Compiler Warnings:** As a service we report the known compiler warnings. The occurrence of a compiler warning may depend on the used configuration and compiler settings.
- **Apparent Issues:** Apparent issues are detected immediately when using the basic software. If an issue does not show up while working with the basic software the ECU project is not affected by the issue. Apparent issues may or may not have workarounds.

The chapter 'New Issues for Information' lists issues that are not relevant for the use case that has been documented in the questionnaire provided to Vector. The issues may, however, be relevant for other use cases. Additionally, issues that have been accepted or are tolerated by the OEM (as defined in the questionnaire) are reported here.

2. New Issues

2.1 Runtime Issues without Workaround

The lists contain issues that have been detected since the last report and which could not be excluded based on the use-cases defined in the questionnaire (see chapter ‘New Issues for Information’).

ESCAN00074189	Service 0x2C: A PID declined by the application via pre-handler is skipped for the DPID definition
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	6.06.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

During the definition of a DPID with service 0x2C, when the application declines a PID in its pre-handler (by setting a NRC) a positive response is sent instead of a negative response with NRC 0x31.

When does this happen:

During runtime.

In which configuration does this happen:

Protocol == KWP
AND
Service 0x2C is used
AND
Pre-Handler for PIDs are used

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00076278	StateOn flag return value may be incorrect
Component@Subcomponent:	Il_Vector_Gm@Implementation
First affected version:	1.00.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The value reported by the StateOn flag (IlGet<signal>RxStateOn) may be incorrect.

- It may return true (non-zero) even though no relevant VN is active

When does this happen:

The issue is observed at runtime, but it is caused by an incorrectly generated macro. If the macro is generated incorrectly, the runtime behavior is consistent and reproducible (it occurs each and every time).

In which configuration does this happen:

In MultiChannel Configurations with State On Flags.

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

2.2 Runtime Issues with Workaround

It is not recommended to update a delivery due to a runtime issue with a documented workaround. The effect of an issue to the ECU functionality has to be analyzed by the customer as the basic software usage and its configuration is not known by Vector. Thereby the risk of change has also to be taken into account.

Index

ESCAN00027894	Memory is overwritten when initializing the CANBedded Stack Nm_Gmlan_Gm@Implementation
ESCAN00045854	An incorrect timeout is issued for Flow Control and Consecutive Frame timing supervision. Tp_Iso15765@GenTool_Geny
ESCAN00067827	Rh850Rscan only: CAN communication shows undefined behavior DrvCan__baseRi14Hll@GenTool_Geny
ESCAN00068912	Positive response to service $A5 03 not suppressed Diag_CanDesc__coreBase@Implementation
ESCAN00073999	Signal handler names have wrong names after deletion of some signals of a DID Diag_CanDesc__coreBase@GenTool_Geny_CANdesc
ESCAN00076096	Periodic response of a DPID requested with service $AA return wrong data Diag_CanDesc__coreBase@Implementation

ESCAN00027894	Memory is overwritten when initializing the CANBedded Stack
Component@Subcomponent:	Nm_Gmlan_Gm@Implementation
First affected version:	3.30.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Memory is overwritten when initializing the CANBedded Stack.

When does this happen:

The issue occurs always and immediately if CclInitPowerOn or IlInitPowerOn is called with the configuration mentioned below.

In which configuration does this happen:

Any configuration, where the number of Nm Channels differs from the number of Can Channels.

Hint: The generated define in kCanNumberOfChannels in can_cfg.h differs from kNmNumberOfChannels generated to nm_cfg.h

Resolution Description:

Workaround:

Do not call IlInitPowerOn in the application. Instead, call IlInit for each channel which uses the Interaction Layer.

Note for GM ECUs: If the Interaction Layer is not used on the first channel in GENy (channel index 0), the application must additionally call CanInitPowerOn before IlInit.

Example:
The ECU has three CAN channels, where the Interaction Layer is only used on the first two.

```
IlInit(0); /* Initialize the IL on CAN channel 0 */
IlInit(1); /* Initialize the IL on CAN channel 1 */
```

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00045854	An incorrect timeout is issued for Flow Control and Consecutive Frame timing supervision.
Component@Subcomponent:	Tp_Iso15765@GenTool_Geny
First affected version:	2.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

An incorrect timeout is issued for Flow Control (TX) and Consecutive Frame (RX) timing supervision in case of large timeouts.

When does this happen:

During runtime at transmission and/or reception of multi frames.

In which configuration does this happen:

This can only appear if channel specific timing is activated (#if defined TP_CHANNEL_SPECIFIC_TIMING)
AND
the configured timeout values are greater than 255 "ticks".
Please note that the number of "ticks" is calculated by dividing the configured timeout value by the configured periodic cycle time of the TP.

Resolution Description:

Workaround:

Use smaller timeouts or increase the call-cycle of the TP task functions.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00067827	Rh850Rscan only: CAN communication shows undefined behavior
Component@Subcomponent:	DrvCan__baseRi14Hll@GenTool_Geny
First affected version:	1.00.00
Fixed in versions:	2.06.00

Problem Description:

What happens (symptoms):

CAN communication shows undefined behavior

When does this happen:

Anytime during runtime (init, rx, tx, ..)

In which configuration does this happen:

Only for platform Rh850 with Rscan cell
All configurations that use any of the physical channels CAN5, CAN6 or CAN7

Resolution Description:

Workaround:

Do not use physical channels CAN5, CAN6 or CAN7.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00068912	Positive response to service $A5 03 not suppressed
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	6.12.01
Fixed in versions:	

Problem Description:

What happens (symptoms):

The positive response to service $A5 03 is not suppressed.
AND possibly
The following compiler warning occurs:

```
 static void DescOemEnableProgrammingMode(DescMsgContext *pMsgContext)
 ^
"desc.c", Warning[Pe177]:
 function "DescOemEnableProgrammingMode" was declared but never referenced
```

When does this happen:

At runtime/compile time.

In which configuration does this happen:

Configurations created in an older delivery affected by ESCAN00061312:
"Not possible to support negative responses while suppressing positive response for service $A5 03"
AND
The 'Reload all description files' button on the 'Diag_CanDesc_Kwp' page in GENy has NOT been pressed at least once since moving from the older delivery.

Hint:

This issue has been analyzed thoroughly and will not be fixed due to the following reasons:
a) In GENy there is no reasonable possibility to detect that an old configuration has been loaded which would require a reload of the CDD.
b) The issue only occurs in a migration scenario with an old GENy configuration from a previous delivery and a new delivery using that configuration.

Resolution Description:

Workaround:

Press the 'Reload all description files' button on the 'Diag_CanDesc_Kwp' page in GENy and then save the configuration. This process only needs to be done once.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00073999	Signal handler names have wrong names after deletion of some signals of a DID
Component@Subcomponent:	Diag_CanDesc__coreBase@GenTool_Geny_CANdesc
First affected version:	6.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

After reloading a modified CDD-file, where signals of a DID were deleted, the signal handler the DID have the names of the deleted signals.

When does this happen:

After deleting some signals within a signal list of a DID in CANdela-Studio and reloading the modified CDD-File
within GENy.

In which configuration does this happen:

Signal handlers are used

Resolution Description:

Workaround:

Empty the field "CANdela document name" and click "Reload all description files". The configuration is now empty and no old names are stored. Afterwards import the CDD file again.

OR

For the affected signals change the Signal Handler Type to "Direct Access" and back to "Signal Handler" again

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00076096	Periodic response of a DPID requested with service $AA return wrong data
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.02.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The periodic response for a DPID requested with service $AA contains wrong data

When does this happen:

During runtime when the following situation occurs:
- One or more DPIDs are requested with service $AA in a specific rate
- For one DPID the data of it's source PIDs is currently read from the application
- During the read out process the reporting of all DPIDs is stopped
- Before the reading is finished the reporting of a DPID is requested again

In which configuration does this happen:

Service $AA is used
AND
((Service $2C is used) OR (The source PIDs of a DPID are asynchronous))

Resolution Description:
Workaround:

Use the High-Performance Mode in CANdesc. The utilization is described in the TechnicalReference in the API description of "DescMayCallStateTaskAgain".

In case at least one source PID used for DPIDs is asynchronous (that means the application can't always provide the data for the PID in the first function call to the application), change the application so the data is always available on the first call of the application function (make the PID synchronous).

Resolution:

The described issue is corrected by modification of all affected work-products.

2.3 Apparent Issues

Apparent issues are detected immediately when using the basic software. If an issue does not show up while working with the basic software the ECU project is not affected by the issue. Apparent issues may or may not have workarounds.

Index

ESCAN00049589	Compile error: direct signal access feature in CANdesc does not consider far memory pointers Diag_CanDesc__coreBase@Implementation
ESCAN00055957	appdesc.c missing line feed (LF) after carraige return (CR) on some lines Diag_CanDesc__coreBase@Implementation
ESCAN00069542	Missing description that initially active VNs are no more activated upon power on Nm_Gmlan_Gm@Doc_TechRef
ESCAN00069732	Objects are not handled by interrupt or polling as configured in the Individual Polling view DrvCan__baseRi15@GenTool_Geny
ESCAN00069876	Incorrect Description for Calibration Attribute nmMaxApplShutDownDenyCnt CBD_TechRef_GmlanCalibration@Doc_TechRef
ESCAN00070445	The P2 timings can be changed in the tool GUI but the new values have no effect on CANdesc code generation Diag_CanDesc__coreBase@GenTool_Geny_CANdesc
ESCAN00070517	Compiler error: missing constant kDescStateSessionDefault Diag_CanDesc__coreBase@Implementation
ESCAN00071069	The description of service 0x12 is out-dated Diag_CanDesc_Oem@Doc_TechRef_Kwp_Gm
ESCAN00071131	Wrong API name "DescApplSendSpontaneousResponse" in Technical Reference Diag_CanDesc__coreBase@Doc_TechRef
ESCAN00073848	Tester present timeout occurs too early after a 0x28 request Diag_CanDesc__coreBase@Implementation
ESCAN00074878	a2l error: The parameter SAMPLE_RATE is always 0 Cp_XcpOnCan@GenTool_Geny
ESCAN00075459	TpRxGetAssignedDestination returns wrong destination Tp_Iso15765@Implementation
ESCAN00076138	GENy does not respond when importing CDD files Diag_Geny_coreBase@GenTool_Geny
ESCAN00076840	Missing pre-conditions in the API description of DescApplSendSpontaneousResponse Diag_CanDesc__coreBase@Doc_TechRef
ESCAN00076879	Comment for Messages wrong - Generated code is not affected DrvCan__base@GenTool_Geny

ESCAN00049589	Compile error: direct signal access feature in CANdesc does not consider far memory pointers
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	1.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compile error for mismatching pointer type assignment.

When does this happen:

At compile time.

In which configuration does this happen:

- CANdesc
AND
- Direct signal access to RAM/ROM objects is used.
AND
- FAR memory

Some services such as the UDS ones 0x22/0x2A and 0x2E, can be processed on signal level. If they are processed on signal level it is possible to choose "Direct Access" as Signal Handler Type. In this case, CANdesc reads or writes the value of signal direct of/to a variable. (The name of the variable is configured in the cdd file or GENy.) If this variable is located in FAR memory a Compiler/Linker warning or error will occur.

Resolution Description:

Workaround:

Avoid direct signal access to such objects and implement the main-handler within the application code. (Choose "Signal Handler" for the Signal Handler Type and copy the data that is located in the FAR memory in the application callback for this signal.)

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00055957	appdesc.c missing line feed (LF) after carraige return (CR) on some lines
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.07.26
Fixed in versions:	

Problem Description:

What happens (symptoms):

The appdesc.c file is missing the line feed (LF) character at the end of certain lines. It should follow the carraige return (CR) character. This will cause compilers and debuggers to display the incorrect line of source code. Additionally, some IDEs will complain that the line feed character is missing.

When does this happen:

At generation time.

In which configuration does this happen:

All configurations.

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00069542	Missing description that initially active VNs are no more activated upon power on
Component@Subcomponent:	Nm_Gmlan_Gm@Doc_TechRef
First affected version:	2.01.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Chapters 3.3 "VN Concept" and 4.2 "Normal Operation" states that initially active VNs are activated upon power on.

This is not correct. Since implementation version 4.02.00 initially active VNs are only activated upon reception or transmission of a HLVW message.

When does this happen:

When reading the technical reference.

In which configuration does this happen:

In configurations with initially active VNs.

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00069732	Objects are not handled by interrupt or polling as configured in the Individual Polling view
Component@Subcomponent:	DrvCan__baseRi15@GenTool_Geny
First affected version:	1.00.00
Fixed in versions:	1.05.01

Problem Description:

What happens (symptoms):

Not all configured BasicCAN objects are visible in the Individual Polling view of GENy.
AND
Any objects are not handled by interrupt or polling as configured in the Individual Polling view.

When does this happen:

Always after adding a further CAN channel while configuring and then anytime during runtime.

In which configuration does this happen:

Individual Polling is enabled
AND
Multiple BasicCAN is enabled
AND
more than one BasicCAN object is configured
AND
more than one CAN channel is configured

Resolution Description:

API Extensions:

No extension of the API.

API Changes:

No modification of the API.

Module handling changes:

No modification of the module handling.

For a detailed description of the API and the handling of the module refer to the Technical Reference.

ESCAN00069876	Incorrect Description for Calibration Attribute nmMaxApplShutDownDenyCnt
Component@Subcomponent:	CBD_TechRef_GmlanCalibration@Doc_TechRef
First affected version:	2.01.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The description in chapter 6.7 for calibration attribute nmMaxApplShutDownDenyCnt suggests that this attribute can be modified in the GENy GUI, but the selection doesn't exist. The documentation will be updated to remove this suggestion.

nmMaxApplShutDownDenyCnt can be modified in the generated handler calibrations file gmlcal.c.

When does this happen:

Always

In which configuration does this happen:

All

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00070445	The P2 timings can be changed in the tool GUI but the new values have no effect on CANdesc code generation
Component@Subcomponent:	Diag_CanDesc__coreBase@GenTool_Geny_CANdesc
First affected version:	6.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The user is able to modify the default P2 timings in the GENtool GUI, but the new values are not used during the CANdesc code generation. As a result the default P2 timings are only applicable.

When does this happen:

At CANdesc configuration resp. code generation time.

In which configuration does this happen:

Any KWP2000 configuration that shall use P2 timings other than the default ones.

Resolution Description:

Workaround:

The P2/P2* timings are set automatically according to the GM specification to the values of 75 ms/ 5000 ms. Usually, there should be no need to change the P2/P2* timings in GENy to a different value.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00070517	Compiler error: missing constant kDescStateSessionDefault
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	1.00.05
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compile error for missing constant kDescStateSessionDefault

When does this happen:

At compile time.

In which configuration does this happen:

CANdesc Full is used
AND
In the used CDD the name of the default session state is different from "Default".

Resolution Description:

Workaround:

Rename the default session state in the CDD to "Default"

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00071069	The description of service 0x12 is out-dated
Component@Subcomponent:	Diag_CanDesc_Oem@Doc_TechRef_Kwp_Gm
First affected version:	3.00.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The description of service 0x12 doesn't consider the changes made with CANdesc 6.
Only one application callback on SID level is generated instead of two, for each sub-function.

When does this happen:

Always.

In which configuration does this happen:

Any.

Resolution Description:
Workaround:

6.2 Service ReadFailureRecordData ($12)
CANdesc generates only one function callback (main-handler) for all service $12 requests and does not offer any special support for this service. Therefore all dispatching and validation steps (e.g. dispatching on sub-function level, check the request length or validate the PID parameter if applicable), as well as the assembly of the response message (including the sub-function byte) have to be performed by the application.

6.2.1 Service ReadFailureRecordIdentifiers ($12 $01)
Depending on the report type requested (PID or DPID) the application must place one of the following values into the first data byte of the response message:
0x00 - for report by PID
0x01 - for report by DPID

Note
The ECU can support either reports by PID or DPID, but not both.

6.2.2 Service ReadFailureRecordParameters ($12 $02)
CANdesc does not automatically include the PID parameter in the response message for service $12 $02. The application must perform this task.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00071131	Wrong API name "DescApplSendSpontaneousResponse" in Technical Reference
Component@Subcomponent:	Diag_CanDesc__coreBase@Doc_TechRef
First affected version:	3.05.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Wrong API name in Technical Reference:

DescApplSendSpontaneousResponse is stated in the Technical Reference but the correct name is DescSendApplSpontaneousResponse

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00073848	Tester present timeout occurs too early after a 0x28 request
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.05.04
Fixed in versions:	

Problem Description:
What happens (symptoms):

The positive response $60 after a tester present timeout is sent too early (<5000 ms).

When does this happen:

- Send service 0x28 (DisableNormalCommunication) request
- Do not send a valid TesterPresent request
- Wait for the tester present timeout to occur (>5000 ms)

In which configuration does this happen:

DiagProtocol == KWP
AND
Service 0x28 DisableNormalCommunication is active

Resolution Description:
Workaround:

1.) Increase the tester present timeout slightly by creating a UserConfig.cfg-file with the following content:

```
----------------
#ifdef kDescS1TimerTicks
# undef kDescS1TimerTicks
# define kDescS1TimerTicks 503
#endif
----------------
```

2.) Load the UserConfig.cfg-file into GENy

3.) Generate the source files

After these steps, the tester present timeout should be between the required timespan 5000 ms to 5100 ms.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00074878	a2l error: The parameter SAMPLE_RATE is always 0
Component@Subcomponent:	Cp_XcpOnCan@GenTool_Geny
First affected version:	1.07.00
Fixed in versions:	1.08.03

Problem Description:

What happens (symptoms):

The parameter SAMPLE_RATE in the generated CanXcp.a2l file is generated with a fixed value of 0 and does not reflect the parameter as configured in the bus timing configuration.

When does this happen:

Always and immediately

In which configuration does this happen:

---

all configurations

Resolution Description:

Workaround:

No workaround available.
The file CanXcp.a2l is only used ECU externally for the XCP Master and can be patched manually.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00075459	TpRxGetAssignedDestination returns wrong destination
Component@Subcomponent:	Tp_Iso15765@Implementation
First affected version:	2.73.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

TpRxGetAssignedDestination returns kTpRequestDiagPhysical, although the current connection is functional.

When does this happen:

During reception of a functional request with normal fixed, extended or mixed29 addressing

In which configuration does this happen:

- multiple addressing AND
- application precopy callback must be enabled (TP_USE_APPL_PRECOPY == kTpOn)
- fast Precopy is disabled
- new ApplTpCheckTA API is used (introduced in TP version 2.73.00)

Resolution Description:

Workaround:

The upper layer which implements the CheckTA / ApplPrecopy callback have to store the information if the reception is functional or physical.
This information have to be used by the code which want to use TpRxGetAssignedDestination().

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00076138	GENy does not respond when importing CDD files
Component@Subcomponent:	Diag_Geny_coreBase@GenTool_Geny
First affected version:	6.12.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

GENy shows 'busy icon' and will not respond for a long time

When does this happen:

Always when importing a CDD file

In which configuration does this happen:

All with periodic DIDs, delay increases with number of periodic DIDs

Resolution Description:

Workaround:

The tool is not crashing, so an unreasonable amount of patience can help.
Larger CDD files can take more than a day to load, so this workaround is not possible for all use-cases.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00076840	Missing pre-conditions in the API description of DescApplSendSpontaneousResponse
Component@Subcomponent:	Diag_CanDesc__coreBase@Doc_TechRef
First affected version:	3.05.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

Only service 0x86 is listed as pre-condition in the description of the API DescApplSendSpontaneousResponse.
However there are additional conditions for that API.

When does this happen:

Always

In which configuration does this happen:

When Service 0x86 is not supported or the feature to send a spontaneous response is disabled in GENy
AND
FBL behavior according to HIS is supported by CANdesc

Resolution Description:
Workaround:

Additional pre-conditions:
The API is also available for specific OEMs in case FBL behavior according to HIS is required (please refer to chapter 11.8).

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00076879	Comment for Messages wrong - Generated code is not affected
Component@Subcomponent:	DrvCan__base@GenTool_Geny
First affected version:	2.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The comment if a message is a BasicCan or a FullCan message is wrong in certain cases

When does this happen:

During configuration time

In which configuration does this happen:

- A message is sent and received (RX and TX message) AND
- FullCan Flag configured for Rx OR Tx message

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

2.4 Compiler Warnings

As a service we also provide the known compiler warnings. The occurrence of a compiler warning may depend on the used basic software configuration and compiler settings.

Index

ESCAN00027751	Compiler warning for cast to smaller type for "failedByteMask" Diag_CanDesc__coreBase@Implementation
ESCAN00029697	Compiler warning for useless assignment on API DescPmClientCheckPid Diag_CanDesc__coreBase@Implementation
ESCAN00031035	Compiler Warning: variable "timer" in "DescRdpiDeletePid" is possibly uninitialized Diag_CanDesc__coreBase@Implementation
ESCAN00032552	Compiler warning statement not reached in Internal Assertion DrvCan__coreHll@Implementation
ESCAN00044044	Compiler Warning: condition is always false Cp_Xcp@Implementation
ESCAN00047283	IL flags are declared without the "volatile" keyword. Il_Vector@Implementation
ESCAN00058378	Compiler warning: narrowing or signed-to-unsigned type conversion found Diag_CanDescGgdaExt_Gm@Implementation
ESCAN00059701	Compiler warning: condition is always true" in the IlTxTimerTask, IlTxStateTask and IlTxRepetitionsAreActive Il_Vector@Implementation
ESCAN00066833	Compiler warning: Redefined macro name when compiling a main GENy project with a sub-project GenTool_GenyDriverBase@GenTool_Geny
ESCAN00067350	Compiler warning: Redefined macro name when compiling a main GENy project with a sub-project GenTool_GenyVcfgNameDecorator@GenTool_Geny

ESCAN00027751	Compiler warning for cast to smaller type for "failedByteMask"
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	3.01.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

Compiler warning message for the assignment:

*failedByteMask = (vuint8)(0x02 << *failedByteMask);

But there is no real danger of losing information by casting down to a smaller type since the code generator does not allow more than 7 (seven) sub-service bytes in the request message. So skipping the SID (bit 0) does not lead to losing the MSB and the value of the failedByteMask cannot be greater than six.

When does this happen:

At compile time.

In which configuration does this happen:

-CANdesc/CANdescBasic

Resolution Description:
Workaround:

Ignore the warning

Resolution:

This ESCAN will not be resolved, since the fix might require more resources on the ECU. The code generator assures that there will be no overflow on the shift operation.

ESCAN00029697	Compiler warning for useless assignment on API DescPmClientCheckPid
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.02.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The compiler generates a warning message for useless assignment (a = a;) .

When does this happen:

At compile time.

In which configuration does this happen:

- CANdesc
AND
- Service 0x22 is supported with multiple DIDs in single request.
AND
- Service 0x2C is not supported.

Resolution Description:
Workaround:

Ignore this warning since it is only because of an useless assignment.

Resolution:

The described issue will not be corrected, since the solution will require more ROM resources.

ESCAN00031035	Compiler Warning: variable "timer" in "DescRdpiDeletePid" is possibly uninitialized
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler produces the warning "variable 'timer' is possibly uninitialized" when compiling desc.c.

When does this happen:

At compile time

In which configuration does this happen:

If service 0xAA supported and at least one periodic transmission mode is supported (i.e. not only "send one response" is supported).

Resolution Description:

Workaround:

No workaround available.

Resolution:

This warning is incorrect and can be safely ignored.

ESCAN00032552	Compiler warning statement not reached in Internal Assertion
Component@Subcomponent:	DrvCan__coreHll@Implementation
First affected version:	2.06.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compiler warning "statement not reached" may occur in the following line:
 assertInternal(((kCanNumberOfTxObjects + CanTxQueuePadBits[kCanNumberOfChannels-1]) <= 65536u), kCanAllChannels, kErrorTxQueueTooManyHandle)
or
 assertInternal(((kCanNumberOfTxObjects + CanTxQueuePadBits[kCanNumberOfChannels-1]) <= 256u), kCanAllChannels, kErrorTxQueueTooManyHandle)

When does this happen:

At compile time

In which configuration does this happen:

In the following circumstances:
- transmit queue (C_ENABLE_TRANSMIT_QUEUE is defined)
AND
- assertion check (C_ENABLE_INTERNAL_CHECK is defined)
AND
- multi channel (C_MULTIPLE_RECEIVE_CHANNEL is defined)
AND
- the CAN driver supports the Bit Queue algorithm. (DRVCAN__HLLTXQUEUEBIT_VERSION is defined in can_def.h and NOT in can_drv.c)

Resolution Description:

Workaround:

This warning can be ignored since there is no danger for the software normal functioning.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00044044	Compiler Warning: condition is always false
Component@Subcomponent:	Cp_Xcp@Implementation
First affected version:	1.26.02
Fixed in versions:	

Problem Description:

What happens (symptoms):

```
Compiling file: ../../BSW/Xcp/xcpProf.c
0 errors, 5 warnings
ctc W549: ["../../BSW/Xcp/xcpProf.c" 2518/22] condition is always false
ctc W549: ["../../BSW/Xcp/xcpProf.c" 2522/22] condition is always false
ctc W549: ["../../BSW/Xcp/xcpProf.c" 2526/22] condition is always false
ctc W549: ["../../BSW/Xcp/xcpProf.c" 2659/22] condition is always false
ctc W549: ["../../BSW/Xcp/xcpProf.c" 2663/22] condition is always false
0 errors, 5 warnings
```

When does this happen:

This happens when XCP_DISABLE_WRITE_PROTECTION and XCP_DISABLE_WRITE_EEPROM are defined.

In which configuration does this happen:

---

see above

Resolution Description:

Workaround:

Enable
XCP_DISABLE_WRITE_PROTECTION or XCP_DISABLE_WRITE_EEPROM

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00047283	IL flags are declared without the "volatile" keyword.
Component@Subcomponent:	Il_Vector@Implementation
First affected version:	3.10.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

IL flags (Indication, FirstValue, Confirmation, Timeout) are declared without the "volatile" keyword. Read and Write access to IL flags has no effect due to a Read-Modify-Write problematic.

e.g.
FlagA and FlagB are in the same byte and set on interrupt level

this sequence is executed on task level:
disable int;
clear FlagA; /*1*/
enable int;
... /*3*/
disable int;
clear FlagB; /*2*/
enable int;

The compiler might optimize this sequence and the flag read and write ALWAYS fails:

read the byte at 1), modify the local copy and write the byte at 2)
if the byte is written on interrupt level at 3), the data is lost.

When does this happen:

At runtime (This Problem has been found by a review and has never been detected in a ECU)

In which configuration does this happen:

- This issue highly depends on the used compiler and compiler options.
- Preemptive IL flag access is used (e.g. interrupt system)

Resolution Description:

Workaround:

Review the optimization configuration of your compiler.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00058378	Compiler warning: narrowing or signed-to-unsigned type conversion found
Component@Subcomponent:	Diag_CanDescGgdaExt_Gm@Implementation
First affected version:	2.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The following compiler warning occurs:

warning (dcc:1643): narrowing or signed-to-unsigned type conversion found: unsigned int to unsigned short

This warning occurs for the following code in GgdaRdiRxTask:

```
 /* send the DTC number and the FTB */
 vuint16 DTCNr = ((vuint16) ggdaContexts[context].uudtPrimBuffer[1] << 8)
 | (vuint16) ggdaContexts[context].uudtPrimBuffer[2];
```

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Configurations which use service $A9.

Resolution Description:

Workaround:

The warning can be safely ignored.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00059701	Compiler warning: condition is always true" in the IlTxTimerTask, IlTxStateTask and IlTxRepetitionsAreActive
Component@Subcomponent:	Il_Vector@Implementation
First affected version:	2.42.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compiler warns for "condition is always true" in the IlTxTimerTask, IlTxStateTask and IlTxRepetitionsAreActive API. This may happen depending on the configuration.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

IlTxTimerTask, IlTxStateTask: Any configuration with exactly one tx message.
IlTxRepetitionsAreActive: Any configuration with exactly one tx message and the API is configured. (IL_ENABLE_SYS_TX_REPETITIONS_ARE_ACTIVE_FCT must be defined)

Hint:

The compiler warning is known and has been analyzed thoroughly for its impact on the code. Nevertheless it will not be fixed due to the rare configuration. The code uses a while loop with a counter and can probably replaced by a for loop, but other compilers or codeanalysers may warn about a useless loop. The code exists for about 15 Years and will not be changed.

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00066833	Compiler warning: Redefined macro name when compiling a main GENy project with a sub-project
Component@Subcomponent:	GenTool_GenyDriverBase@GenTool_Geny
First affected version:	2.00.00
Fixed in versions:	2.10.00

Problem Description:
What happens (symptoms):

Compiler generates warning on the following macro redefinitions:

v_cfg_1.h(180) : CC78K0R warning W0816: Redefined macro name 'V_ATOMIC_BIT_ACCESS_IN_BITFIELD'
v_cfg_1.h(181) : CC78K0R warning W0816: Redefined macro name 'V_ATOMIC_VARIABLE_ACCESS'
v_cfg_1.h(196) : CC78K0R warning W0816: Redefined macro name 'kComNumberOfNodes'
v_cfg_1.h(197) : CC78K0R warning W0816: Redefined macro name 'ComSetCurrentECU'
v_cfg_1.h(198) : CC78K0R warning W0816: Redefined macro name 'comMultipleECUCurrent'

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

When two GENy projects are compiled together (e.g. CAN, LIN), one is setup as "main project" and the other is setup as "sub-project"

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00067350	Compiler warning: Redefined macro name when compiling a main GENy project with a sub-project
Component@Subcomponent:	GenTool_GenyVcfgNameDecorator@GenTool_Geny
First affected version:	2.19.00
Fixed in versions:	2.26.01, 2.27.00

Problem Description:
What happens (symptoms):

Compiler generates warning on the following macro redefinitions:

v_cfg_1.h(180) : CC78K0R warning W0816: Redefined macro name 'V_ATOMIC_BIT_ACCESS_IN_BITFIELD'
v_cfg_1.h(181) : CC78K0R warning W0816: Redefined macro name 'V_ATOMIC_VARIABLE_ACCESS'
v_cfg_1.h(196) : CC78K0R warning W0816: Redefined macro name 'kComNumberOfNodes'
v_cfg_1.h(197) : CC78K0R warning W0816: Redefined macro name 'ComSetCurrentECU'
v_cfg_1.h(198) : CC78K0R warning W0816: Redefined macro name 'comMultipleECUCurrent'

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

When two GENy projects are compiled together (e.g. CAN, LIN), one is setup as "main project" and the other is setup as "sub-project"

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

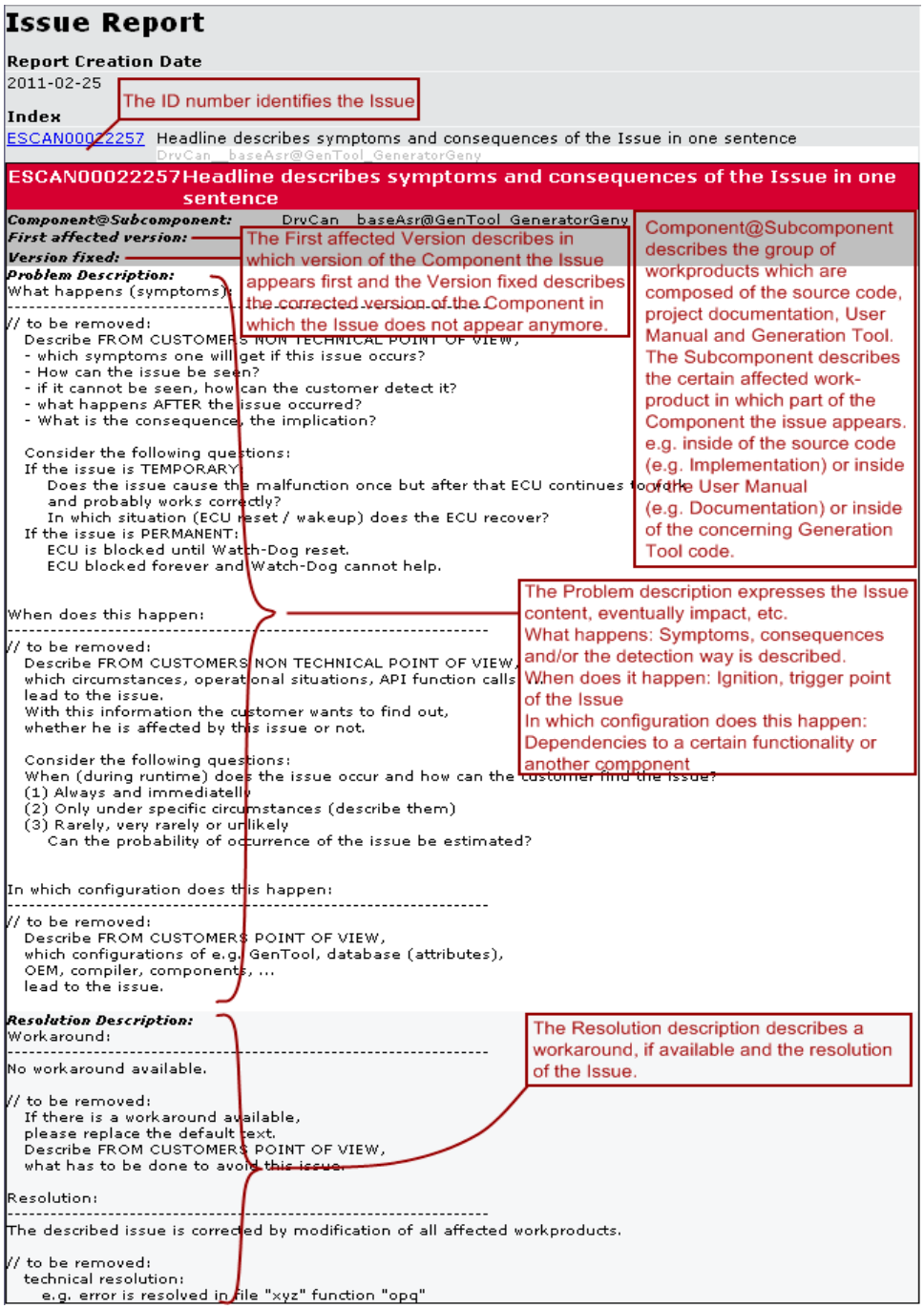
3. New Issues for Information

Issues which should not have an effect on the usage of the license as the issues are relevant for use cases other than those defined in the questionnaire. The list contains issues that have been detected since the last report.

Issues listed in this section are not relevant for the use case that has been documented in the questionnaire provided to Vector. However, the issues may be relevant for other use cases. Also issues that have been accepted or are tolerated by the OEM (as defined in the questionnaire) are reported here.

No issue to be reported.

4. Report Legend



5. Quality Management Contact

Diemo Krüger
PES Quality Management Engineer
Productline Embedded Software (PES)

Vector Informatik GmbH
Ingersheimer Str. 24
D-70499 Stuttgart

Phone: +49 711 80670-3477
Fax: +49 711 80670-399
eMail: diemo.krueger@vector.com