

License Number

CBD1500834

Customer

Huizhou Desay SV Automotive Co., Ltd.
Package: MSR_Vector_SLP4
Micro: R7F701432
Compiler: Green Hills 6.14

Maintenance Expiry Date

2016-03-01

SIP Delivery Date

2016-02-19

SIP Version

14.00.01

SLP

MSR_Vector_SLP4

Delivery Number

D00

Report Creation Date

2016-02-04

Contact

In case of questions or the need for an update of the basic software delivery, please contact support@cn.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: 0](#)
 - [2.2 Runtime Issues with Workaround: 1](#)
 - [2.3 Apparent Issues: 14](#)
 - [2.4 Not Released Functionality: 0](#)
 - [2.5 Compiler Warnings: 2](#)
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.
- **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.
- **Not Released Functionality:** Not released functionalities are modules and features that have not yet passed a complete development cycle (they are e.g. not or only partly tested). For serial production projects the integrator has to ensure that all BETA features are disabled as indicated. If a ESCAN affects a complete BSW module, the module must not be used for serial production.
- **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.

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').

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

ESCAN00085191	EIINT exception table is not aligned on 512Byte boundary Os_PlatformRh850Gen6@Generator
-------------------------------	--

ESCAN00085191 EIINT exception table is not aligned on 512Byte boundary

Component@Subcomponent: Os_PlatformRh850Gen6@Generator

First affected version: 1.00.00

Fixed in versions: 1.10.00, 1.81.00

Problem Description:

What happens (symptoms):

-
- 1) An unhandled core exception is called instead of the configured core exception handler
 - 2) The CPU branches to program code which is mapped before the core exception table and the result is unpredictable
 - 3) An unhandled EIINT exception is called instead of the configured EIINT interrupt handler
 - 4) The CPU branches to program code which is mapped before the EIINT exception table and the result is unpredictable

When does this happen:

This happens during runtime when a core exception or an EIINT interrupt occurs.

In which configuration does this happen:

This happens in all configurations.

Resolution Description:

Workaround:

Add 512Byte alignment to linker file:

Example

```
SECTIONS
{
  .ROM_start :>ROM_0000

  .osExceptionVectorTable_c0 (align 512) :>.
  .osEIINTVectorTable_c0 (align 512) :>.
  .osExceptionVectorTable_c1 (align 512) :>.
  .osEIINTVectorTable_c1 : (align 512) >.
```

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

ESCAN00075933	Generator crashes when parsing an OIL file with an include directive with nonexistent absolute path Os_CoreGen6@Generator
ESCAN00076481	Debugger refuses to read ORTI file Os_CoreGen6@Generator
ESCAN00078961	Generator does not state an error message when two ISRs of different categories are using the same function name, if ISR hooks are activated Os_CoreGen6@Generator
ESCAN00079209	Counter macros <UNIT>*2TICKS and TICKS2*<UNIT> and TimerConstants use floating point operations with incorrect rounding Os_CoreGen6@Generator
ESCAN00079317	Generator does not check whether at least one APPMODE is assigned to each container TASK/AUTOSTART=TRUE Os_CoreGen6@Generator
ESCAN00079977	TechnicalReference does not help to solve installation problem Os_CoreGen6@Doc_TechRef
ESCAN00080172	Compiler error: missing API SyncScheduleTable, StartScheduleTableSynchron and/or SetScheduleTableAsync Os_CoreGen6@Generator
ESCAN00080742	OSTICKDURATION not defined Os_CoreGen6@Generator
ESCAN00080817	Illegal API optimization for SuspendOSInterrupts/ResumeOSInterrupts Os_CoreGen6@Generator
ESCAN00080956	stack sharing for internal resources not working Os_CoreGen6@Generator
ESCAN00083045	The oil parser does not work properly with the unsigned 64 bit integer Os_CoreGen6@Generator
ESCAN00085663	Unexpected Windows Error Message in case of wrongly configured Linked Resource Os_CoreGen6@Generator
ESCAN00086301	TimingMeasurement and Synchronisation is not supported in SC2 Os_CoreGen6@Generator
ESCAN00086809	Generator Output differs on different PCs Os_CoreGen6@Generator

ESCAN00075933 Generator crashes when parsing an OIL file with an include directive with nonexistent absolute path

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

The generate crashes with an exception.

When does this happen:

during parsing the file.

In which configuration does this happen:

ALL

(identical to ESOS00004007)

Resolution Description:

Workaround:

If this happens (rare case), please open the OIL file with an editor and correct the path to the correct include path. The include file should typically be found in the generator path.

Resolution:

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

ESCAN00076481 Debugger refuses to read ORTI file

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 6.06.00

Fixed in versions:

Problem Description:

What happens (symptoms):

The debugger is unable to provide information about the currently active task, its priority, the currently running Cat2 ISR, the currently performed OS API service and the last occurred errors (we currently do not know whether the issue is detected at parsing the ORTI file or later).

When does this happen:

This happens always and immediately at debug time when the customer tries to trace the currently active task or ISR.

In which configuration does this happen:

This happens only in configurations with ORTI enabled and depends on no further configuration details. This may happen only with a debugger from a certain vendor.

(this issue is identical to ESOS00004101)

Resolution Description:

Workaround:

On single core systems perform the following substitution manually in the ORTI file:

Replace:

"osConfigBlock.CcbAddress->"

By:

"osCtrlVarsCore0."

On multicore systems, there is currently no workaround.

Resolution:

The described behavior is to be corrected by modification of the code.

ESCAN00078961 Generator does not state an error message when two ISRs of different categories are using the same function name, if ISR hooks are activated

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

Undetected erroneous configuration leads to undefined behavior

In which configuration does this happen:

ISR hooks are activated

A Cat1 ISR and a Cat2 ISR are mapped to the same function using the configuration parameter "UseSpecialFunctionName"

When does this happen:

This happens immediately at the return from a user defined ISR function, dependent on the configuration of that function:

- If the user defined ISR function is configured to end with a 'return from interrupt' instruction and the cat2 ISR has occurred.
- If the user defined ISR function is configured to end with a normal 'return' instruction and the cat1 ISR has occurred.

Resolution Description:

Workaround:

Prevent to configure two ISRs of different categories which use the same same function as a special function name.

Resolution:

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

ESCAN00079209 Counter macros <UNIT>*2TICKS and TICKS2*<UNIT> and TimerConstants use floating point operations with incorrect rounding

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 8.00.00

Fixed in versions: 9.01.06

Problem Description:

What happens (symptoms):

Two symptoms:

- (1) A floating point library needs to be linked additionally (code size increase).
- (2) The result value is always rounded downwards, no mathematical rounding.

When does this happen:

- (1) In addition to the configurational preconditions, this happens only if the macros are used with variables as input values.
- (2) See the description of configuration below.

In which configuration does this happen:

- (1) This happens only if no hardware floating point support is used
- (2) This happens only if the macro/TimerConstants computations lead to non-integer values.

Resolution Description:

Workaround:

Consider that rounding is not done mathematical.

Resolution:

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

ESCAN00079317 Generator does not check whether at least one APPMODE is assigned to each container TASK/ AUTOSTART=TRUE

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

A task, which is configured with AUTOSTART=TRUE, is not started. If this task is used as initial task (single entry point), the whole application will not start.

When does this happen:

This happens always and immediately when the OS is started.

In which configuration does this happen:

A task is configured with AUTOSTART=TRUE and no APPMODE is assigned.

Resolution Description:

Workaround:

The issue is simply that a wrong configuration is stays undetected. to work around this issue, simply correct the configuration.

Resolution:

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

ESCAN00079977 TechnicalReference does not help to solve installation problem

Component@Subcomponent: Os_CoreGen6@Doc_TechRef

First affected version: 6.16.01

Fixed in versions:

Problem Description:

What happens (symptoms):

The OS Generator refuses to generate files with the following message:
[Fatal Error: 0021] XML write error: Failed to CreateInstance on an XML DOM

Neither the message itself, nor the TechnicalReference help to solve the issue.

When does this happen:

This happens during OS generation.

In which configuration does this happen:

This issue is configuration independent.

Resolution Description:

Workaround:

Install MSXML 4.0 Service Pack 3.

Resolution:

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

ESCAN00080172 Compiler error: missing API SyncScheduleTable, StartScheduleTableSynchron and/or SetScheduleTableAsync

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 8.00.01

Fixed in versions:

Problem Description:

What happens (symptoms):

The compiler/linker issues an error that one or more of the following functions are implicitly defined, do not have a prototype, or cannot be linked:

SyncScheduleTable

StartScheduleTableSynchron

SetScheduleTableAsync

When does this happen:

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

In which configuration does this happen:

This happens if in an OS SC1 or SC3 at least one schedule table is configured with OsScheduleTableSyncStrategy other than NONE.

Resolution Description:

Workaround:

No workaround necessary. The issue occurs only in cases where a non-supported or non-licensed feature is enabled in the configuration.

Resolution:

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

ESCAN00080742 OSTICKDURATION not defined

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 8.00.00

Fixed in versions:

Problem Description:

What happens (symptoms):

Compilation error is reported as there has no OSTICKDURATION been defined but used,

When does this happen:

The error is issued by the compiler during compilation of the code if the code uses OSTICKDURATION.

In which configuration does this happen:

This happens independent from the configuration.

Resolution Description:

Workaround:

Two alternatives:

1. Use the new macros OSSECONDSPERTICK_<CounterName> instead. These define the time of a counter tick as a floating point value in seconds while OSTICKDURATION defined it as an integer value in nanoseconds.
2. Create define macro OSTICKDURATION manually as an integer value which is equal to the return value of OS_TICKS2NS_<CounterName>(1). Mind that this value depends on the configuration parameter OsCounter/OsSecondsPerTick and needs adaptation after each change of that parameter. The macro OS_TICKS2NS_<CounterName> may be found in the generated file tcbpost.h. Direct usage of OS_TICKS2NS_<CounterName> for the definition of OSTICKDURATION could cause compilation problems of the RTE if the RTE uses OSTICKDURATION in a preprocessor condition where floating point operations are not allowed.

Resolution:

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

ESCAN00080817 Illegal API optimization for SuspendOSInterrupts/ResumeOSInterrupts

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

API functions SuspendOSInterrupts and ResumeOSInterrupts are not available, even with API optimization set to Manual and corresponding API functions enabled.

When does this happen:

Functions are not available at link time

In which configuration does this happen:

configurations without Cat2 ISRs

Resolution Description:

Workaround:

Simply configure one ISR of category 2. Alternatively, configure a hardware counter with an alarm using that counter.

Resolution:

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

ESCAN00080956 stack sharing for internal resources not working

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

if you configure tasks with same internal resources, they don't share a stack

When does this happen:

always

In which configuration does this happen:

ASR 4
Non SafeContext

Resolution Description:

Workaround:

the customer has to patch osStacks.h and osStacks.c (manual stack sharing)

Resolution:

remove the filter to RESOURCE[APPLICATION in function void VCalculator::addTaskStacks(string taskStackPrefix)

ESCAN00083045 The oil parser does not work properly with the unsigned 64 bit integer

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

If an unsigned 64 bit integer (e.g. TIMEFRAME) is used in oil file, the value is not properly recognized by the oil parser.

When does this happen:

This happens at generation time.

In which configuration does this happen:

This happens only in configurations which contain 64 bit unsigned integer values bigger than 0x7ffffffffff (decimal: 9223372036854775807).

Resolution Description:

Workaround:

No workaround available.

Resolution:

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

ESCAN00085663 Unexpected Windows Error Message in case of wrongly configured Linked Resource

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

Windows states that the MICROSAR OS generator (gen<PlatformName>.exe) does not work anymore.

When does this happen:

This happens always and immediately at generation time when the configuration is erroneous as described below.

In which configuration does this happen:

The configuration contains a linked resource which has no link to another resource. This is an invalid configuration for linked resources.

Resolution Description:

Workaround:

Add a link to a valid resource to all resources with type 'LINKED'

Resolution:

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

ESCAN00086301 TimingMeasurement and Synchronisation is not supported in SC2

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 7.00.00

Fixed in versions: 9.01.07

Problem Description:

What happens (symptoms):

TimingMeasurement and SchedT Synchronization Tests fail.
In TM not all OS API functions were available (compiler error).
For Synchronization the wrong SC was calculated.

When does this happen:

At Generation / Compile Time

In which configuration does this happen:

SC2, non-SafeContext

Resolution Description:

Workaround:

No workaround necessary as systems with this issue will never be delivered.

Resolution:

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

ESCAN00086809 Generator Output differs on different PCs

Component@Subcomponent: Os_CoreGen6@Generator

First affected version: 5.28.80

Fixed in versions:

Problem Description:

What happens (symptoms):

Even with same input data (OIL) generated data differs in its content if it is generated on different PC (maybe also on the same?)

In the specific case event masks and arrays `oskResCeilingPrio` `oskResCeilingPrioMask` are affected.

The full system needs to be compiled again, although the configuration has not changed.

When does this happen:

During generation

In which configuration does this happen:

Potentially all where multiplicity of a OS configuration element is greater 1.

Resolution Description:

Workaround:

No workaround necessary as the described behaviour causes no malfunction of the embedded code (only just potential confusion of customer).

Resolution:

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

2.4 Not Released Functionality

Not released functionalities are modules and features that have not yet passed a complete development cycle (they are e.g. not or only partly tested). For serial production projects the integrator has to ensure that all BETA features are disabled as indicated. If a ESCAN affects a complete BSW module, the module must not be used for serial production.

No issue to be reported.

2.5 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

[ESCAN00081825](#) Compiler warning: non-portable OS code compiled with `--asm_warnings`
Os_PlatformRh850Gen6@Implementation

[ESCAN00085294](#) `osAmITrusted()` used, even if `SysCallIsFastAgainstDecisionAndCall=TRUE`
Os_CoreGen6@Implementation

ESCAN00081825 Compiler warning: non-portable OS code compiled with --asm_warnings

Component@Subcomponent: Os_PlatformRh850Gen6@Implementation

First affected version: 1.00.00

Fixed in versions:

Problem Description:

What happens (symptoms):

When compiling the OS files, the compiler issues several warnings when compiling with the option --asm_warnings.

The warning is: "warning #1546-D: asm constructs are non-portable"

Those warnings can safely be ignored. The OS has non portable code inside and in case of inline assembly the compiler issues warnings if used with --asm_warnings

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:

the warnings will be issued when compiling the OS with the compiler option --asm_warnings.

Hint:

The compiler warning is known and has been analyzed thoroughly for its impact on the code. Nevertheless it will not be fixed as the OS is non-portable.

Resolution Description:

Workaround:

It is possible to compile the OS without this option and to compile all other sources / modules with this option.

If OS is compiled with this option, then it is safe to ignore the issued warnings.

ESCAN00085294		osAmITrusted() used, even if SysCallIsFastAgainstDecisionAndCall=TRUE	
Component@Subcomponent:	Os_CoreGen6@Implementation		
First affected version:	9.01.00		
Fixed in versions:	9.01.06		
Problem Description:			
What was done:			

Embedded code change.			
Why was it done:			

- The if case is never reached on some platforms.			
What is the impact on the application:			

- Maybe compiler warning.			
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.			

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

Issue Report	
Report Creation Date 2011-02-25	
Index ESCAN0002257 Headline describes symptoms and consequences of the Issue in one sentence DrvCan_baseAsr@GenTool_GeneratorGeny	
ESCAN0002257 Headline describes symptoms and consequences of the Issue in one sentence	
Component@Subcomponent: DrvCan_baseAsr@GenTool_GeneratorGeny First affected version: _____ Version fixed: _____ Problem Description: What happens (symptoms): ----- // to be removed: Describe FROM CUSTOMERS NON TECHNICAL POINT OF VIEW, - which symptoms one will get if this issue occurs? - How can the issue be seen? - if it cannot be seen, how can the customer detect it? - what happens AFTER the issue occurred? - What is the consequence, the implication? Consider the following questions: If the issue is TEMPORARY: Does the issue cause the malfunction once but after that ECU continues to work and probably works correctly? In which situation (ECU reset / wakeup) does the ECU recover? If the issue is PERMANENT: ECU is blocked until Watch-Dog reset. ECU blocked forever and Watch-Dog cannot help. When does this happen: ----- // to be removed: Describe FROM CUSTOMERS NON TECHNICAL POINT OF VIEW, which circumstances, operational situations, API function calls lead to the issue. With this information the customer wants to find out, whether he is affected by this issue or not. Consider the following questions: When (during runtime) does the issue occur and how can the customer find the issue? (1) Always and immediately (2) Only under specific circumstances (describe them) (3) Rarely, very rarely or unlikely Can the probability of occurrence of the issue be estimated? In which configuration does this happen: ----- // to be removed: Describe FROM CUSTOMERS POINT OF VIEW, which configurations of e.g. GenTool, database (attributes), OEM, compiler, components, ... lead to the issue. Resolution Description: Workaround: ----- No workaround available. // to be removed: If there is a workaround available, please replace the default text. Describe FROM CUSTOMERS POINT OF VIEW, what has to be done to avoid this issue. Resolution: ----- The described issue is corrected by modification of all affected workproducts. // to be removed: technical resolution: e.g. error is resolved in file "xyz" function "opq"	Component@Subcomponent describes the group of workproducts which are composed of the source code, project documentation, User Manual and Generation Tool. The Subcomponent describes the certain affected work-product in which part of the Component the issue appears. e.g. inside of the source code (e.g. Implementation) or inside of the User Manual (e.g. Documentation) or inside of the concerning Generation Tool code. The First affected Version describes in which version of the Component the Issue appears first and the Version fixed describes the corrected version of the Component in which the Issue does not appear anymore. The Problem description expresses the Issue content, eventually impact, etc. What happens: Symptoms, consequences and/or the detection way is described. When does it happen: Ignition, trigger point of the Issue In which configuration does this happen: Dependencies to a certain functionality or another component The Resolution description describes a workaround, if available and the resolution of the Issue.

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