

Author(s)	Sven Hesselmann	
Restrictions	Customer confidential - Vector decides	
Abstract	Introduction how to integrate 3rd partly modules into the MICROSAR4 stack	
ADSTRACT	Introduction now to integrate 3rd partiy modules into the MICROSAR4 stack	

### **Table of Contents**

1.0	Overview	1
2.0	Integration in DaVinci Configurator 5	2
2.1	Configuration With CFG5	2
2.1.1	Adding of BSWMD File	2
2.1.2	Adding a Module to the Current Project	3
2.2	External Generation Step	4
2.2.1	Manual Set-Up	5
2.2.2	Automatic Set-Up	5
2.3	Internal Behavior Description	5
2.3.1	Example for Internal Behavior Description	5
2.3.2	Templates for Internal Behavior Description	6
2.4	RTE configuration	7
2.5	CDD Configuration	7
3.0	Settings.xml	8
4.0	Integration Into the Build Project	8
4.1	Compiler_Cfg.h	9
4.2	MemMap.h	9
5.0	Additional Resources	10
6.0	Contacts	10

### 1.0 Overview

This application note describes integration of third party modules (i.e. MCAL) into DaVinci Configurator Pro 5 (CFG5 for short) and the MICROSAR stack for AUTOSAR Release 4.x.

This application note uses the MCU module as an example.



<sup>™</sup> Vector Eparm. <sup>™</sup> <sup>™</sup> <sup>™</sup> <sup>™</sup>						
Project Sample_Name.epa	Module: Mcu					
<enter filter="" here=""> Filter Reset Live</enter>	BSW-IMPLEMENTA	TION				
⊡ Mcu_bswmd.arxml	Short name:	Mcu_Impl		i		
Mcu_Impl	Module Def Ref:	/VendorX/Mcu		i	3	
⊡…i Mcu ⊨i McuGeneralConfiguration	Preconfigured config:			i	2	
McuDevErrorDetect	Recommended config:			i	8	
McuGietRamStateApi	AR release version:	i	Vendor Id:	i		
McuNoPII	SW version number:	i	Vendor Api Infix:	i		
····· · · · · · · · · · · · · · · · ·	Use Code generator:			i		
<ul> <li>McuMainFunctionPeriod</li> <li>         ⊕ -          ⊕ McuModuleConfiguration ⊕ -          ⊕ McuPublishedInformation         </li> </ul>	Internal Behavior Ref:	/VendorX/Mcu_ib_bswmd/BswModu	lleDescriptions/Mcu/McuBehavior	i		
Mcu_bswmd.arxml /VendorX/Mcu_Imp	bl					
AUTOSAR Version 4.0.3					:	

Figure 1 – Overview of BSWMD file of MCU

# 2.0 Integration in DaVinci Configurator 5

For integration of a module into a MICROSAR stack, different things have to be done.

If the module fulfills one of the following list points, check this chapter for the description.

The module:

- has parts, generated based on the configuration (i.e. ECUC file)
- requires the SCHM API for Exclusive Area handling
- has cyclic MainFunction calls
- needs access to communication PDUs

### 2.1 Configuration With CFG5

If the module shall be configured within the CFG5, the tool requires its modules description in a BSWMD file (basis software module description). Also the module has to be added to the configuration.

### 2.1.1 Adding of BSWMD File

Provide an additional search path for BSWMD files within your configuration. The BSWMD file must end with **.arxml** to be noticed by CFG5.

Open **Project|Project Setting|Modules|Additional Definitions** and **Add** the path to the module's BSWMD file as shown in the following screenshots.



🚰 Project Settings Editor 🛛		
🔄 < 🤟 Modules 🕨 😤 Additional	Definitions	
🔍 <filter></filter>	Search Paths for Additional Module Definitions	
Code Generation External Generation Steps Custom Workflow Custom Workflow Steps Modules Additional Definitions Project Settings	The search paths are scanned on project load only. Please close and reopen the project after changing this list.	

Figure 2 – Modules Additional Definitions

🚰 Project Settings Editor 🛛				
🔚 < 🐈 Modules 🕨 📸 Additional Definitions				
🔍 <filter></filter>	Search Paths for Additional Module Definitions			
Code Generation External Generation Steps Custom Workflow Custom Workflow Steps Modules Additional Definitions Project Settings	The search paths are scanned on project load only. Please close and reopen the project after changing this list. 국가 [C:\Vector\CBD1200333_D01_V85x_AddOn\Mcu\bswmd 《Add ②			

Figure 3 – BSWMD added

Now the CFG5 knows the module and it can be added to the configuration. But before the configuration has to be closed and opened again.

### 2.1.2 Adding a Module to the Current Project

For adding the module to the current configuration, open **Project|Project Settings|Modules** and **Add** it with the blue plus. If the path to the module is within the delivered SIP, you will find it in **Select from SIP** otherwise in **Select additional definition** (see screenshots below).



🚰 Project Settings Editor 🛛		
🔚 < 🤟 Modules 🕨 Addition	al Definitions	
Code Generation	▼ ⊕ ps X	
Custom Workflow	Modules Assistant	
Modules CAdditional Definitions	Source Select the source of the module definitions.	<b>**</b>
	Select from Software Integration Package (SIP)	
	Select from AUTOSAR Standard Definition	
	Select additional definition	
	< <u>B</u> ack <u>Next &gt;</u> <u>F</u> inish	Cancel

Figure 4 – Module Assistant

🚟 Modules	Ass	istant					
Module Definitions Select the definition of the modules to be added.							
Show or instantia	nly t ated	hose definitions, which multiple times).	are	not yet instantiated	in the ECU configuration (	or which can be	
Module	- 	Definition Reference	7	Domain <sub>v</sub>			
Mcu		/VendorX/Mcu		Base Services			
This list dis search path	play	rs all module definitions r additional module de	s fou finit	und in the AUTOSAR ions (see Project Set < <u>B</u> ack <u>h</u>	files (extension .axxml), wh tings Editor   Modules   Ad lext >	iich are located in the ditional Definitions). Cancel	

Figure 5 – Module Definitions

Now the module is within your project, configure it using the **Basic Editor**.

### 2.2 External Generation Step

If the module has parts generated based on the configuration of the ECUC file and the generation shall be started from the CFG5, the generation list has to be extended. The configuration of the generation steps for third-party modules can either be done manually or by a configuration file, making it easier to reuse your module for further projects.

# vector

# 2.2.1 Manual Set-Up

Open **Project|Project Settings|Code Generation|External Generation Steps** and **Add** the generation settings using the blue plus.

🚰 Project Settings Editor 🛛					- 0
🔚 🕊 🧠 Code Generation 🕨 🖏 External Generation Steps					0
<pre>     </pre> <pre>         <pre></pre></pre>	• +	ExtGen_DrvAdc ExtGen_DrvPort	Name:	ExtGenManually	
🖷 External Generation Steps & Custom Workflow	×	ExtGen_DrvWdg ExtGen_DrvSpi	Command line executable:	C:\Vector\CBD1200333_D01_V85x_AddOn\Mcu\Mcu.exe Prets <ctr.+smce> for tuggetions.</ctr.+smce>	
Custom Workflow Steps		ExtGen_DrvPwm ExtGen_DrvEls	Execution type:		•
Kadditional Definitions		ExtGen_DrvFee	Parameters:	\$(GenDataFolder) \$(ProcessingEcuCFile)generate	^
Project Settings		ExtGen_DrvGpt			
		ExtGen/DrvIcu			
		Excellinationly			-
				Press <ctrl+space> for suggestions.</ctrl+space>	1 🛌 🗸
			Working folder:	Press <ctrl+space> for suggestions.</ctrl+space>	] 🚄 -
			Comment:		
			Supports validation		
			Parameters:	\$(GenDataFolder) \$(ProcessingEcuCFile)validate	*
					-
				Press <ctrl+space> for suggestions.</ctrl+space>	
			Use specific AUTOSAR for	rmat 🕅	
			AUTOSAR format:	4.0.3	-
			Use specific transformation	on file 🔳	
			Transformation file:	Press <ctrl+space> for suggestions.</ctrl+space>	<u>ب</u> ق
			Output file:	Deer (TTE) (DATE: for constitue	<u> </u>
				ness summannaak jarsayyesaans.	

Figure 6 – External Generation Steps

Specify the module generator settings (i.e. parameters to be handed over). If the module generator also supports validation or requires a transformation of the input file, this can also be configured. For further information also see the Help Content of CFG5.

# 2.2.2 Automatic Set-Up

The settings described in 2.2.1 can also be done automatically by a so-called **Settings.xml**. For configuration options of the Settings.xml please refer to 3.0 Settings.xml.

# 2.3 Internal Behavior Description

Most AUTOSAR modules require Exclusive Area and / or MainFunction handling by the RTE. The MICROSAR RTE reads this information from the so-called Internal Behavior description, which is a part of a BSWMD file. This file has to be provided to the CFG5 by placing it into the folder for InternalBehavior files (default is ./Config/InternalBehavior).

The <BSW-IMPLEMENTATION> container within the BSWMD file must have a reference to the Internal Behavior (i.e. <BEHAVIOR-REF DEST="BSW-INTERNAL-

BEHAVIOR">/VendorX/Mcu\_ib\_bswmd/BswModuleDescriptions/Mcu/McuBehavior</BEHAVIOR-REF>, see also Figure 1). The RTE reads the Internal Behavior of the module from this file and provides a solving action to create an RteBswModuleInstance with this information.

# 2.3.1 Example for Internal Behavior Description

The following example for an Internal Behavior description defines an exclusive area called MCU\_EXCLUSIVE\_AREA\_0 and a MainFunction called Mcu\_MainFunction, which has to be called with a cycle time of 0.01 seconds. The file should be placed into the folder for InternalBehavior files (default is ./Config/InternalBehavior) or the content can even be in the BSWMD file itself.



```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<AUTOSAR xmlns="http://autosar.org/schema/r4.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://autosar.org/schema/r4.0 autosar 4-0-3.xsd">
  <AR-PACKAGES>
    <AR-PACKAGE>
      <SHORT-NAME>VendorX</SHORT-NAME>
      <AR-PACKAGES>
        <AR-PACKAGE>
          <SHORT-NAME>Mcu ib bswmd</SHORT-NAME>
          <AR-PACKAGES>
            <AR-PACKAGE>
              <SHORT-NAME>BswModuleDescriptions</SHORT-NAME>
              <ELEMENTS>
                <BSW-MODULE-DESCRIPTION>
                  <SHORT-NAME>Mcu</SHORT-NAME>
                  <PROVIDED-ENTRYS>
                    <BSW-MODULE-ENTRY-REF-CONDITIONAL>
                      <BSW-MODULE-ENTRY-REF DEST="BSW-MODULE-
ENTRY">/VendorX/Mcu ib bswmd/BswModuleDescriptions/Mcu MainFunction</BSW-MODULE-ENTRY-REF>
                    </BSW-MODULE-ENTRY-REF-CONDITIONAL>
                  </PROVIDED-ENTRYS>
                  <INTERNAL-BEHAVIORS>
                    <BSW-INTERNAL-BEHAVIOR>
                      <SHORT-NAME>McuBehavior</SHORT-NAME>
                      <EXCLUSIVE-AREAS>
                        <EXCLUSIVE-AREA>
                          <SHORT-NAME>MCU EXCLUSIVE AREA 0</SHORT-NAME>
                        </EXCLUSIVE-AREA>
                      </EXCLUSIVE-AREAS>
                      <ENTITYS>
                        <BSW-SCHEDULABLE-ENTITY>
                          <SHORT-NAME>Mcu MainFunction</SHORT-NAME>
                          <IMPLEMENTED-ENTRY-REF DEST="BSW-MODULE-
ENTRY">/VendorX/Mcu ib bswmd/BswModuleDescriptions/Mcu MainFunction</IMPLEMENTED-ENTRY-REF>
                        </BSW-SCHEDULABLE-ENTITY>
                      </ENTITYS>
                      <EVENTS>
                        <BSW-TIMING-EVENT>
                          <SHORT-NAME>Mcu MainFunctionTimingEvent0</SHORT-NAME>
                          <STARTS-ON-EVENT-REF DEST="BSW-SCHEDULABLE-
ENTITY">/VendorX/Mcu ib bswmd/BswModuleDescriptions/Mcu/McuBehavior/Mcu MainFunction</STARTS-ON-EVENT-REF>
                          <PERIOD>0.01
                        </BSW-TIMING-EVENT>
                      </EVENTS>
                    </BSW-INTERNAL-BEHAVIOR>
                  </INTERNAL-BEHAVIORS>
                </BSW-MODULE-DESCRIPTION>
                <BSW-MODULE-ENTRY>
                  <SHORT-NAME>Mcu MainFunction</SHORT-NAME>
                  <CALL-TYPE>SCHEDULED</CALL-TYPE>
                  <EXECUTION-CONTEXT>TASK</EXECUTION-CONTEXT>
                </BSW-MODULE-ENTRY>
              </ELEMENTS>
            </AR-PACKAGE>
          </AR-PACKAGES>
        </AR-PACKAGE>
      </AR-PACKAGES>
    </AR-PACKAGE>
  </AR-PACKAGES>
</AUTOSAR>
```

# 2.3.2 Templates for Internal Behavior Description

Some modules provided by a MICROSAR delivery are not developed by Vector and do not have an automatic Internal Behavior creation (i.e. MCALs). For easier integration some of them provide templates for the Internal Behavior description. If these files are available within the current delivery they are positioned at .\Misc\InternalBehaviorTemplates. Please remove the starting '\_' marking the files as templates, copy them to the





InternalBehavior folder of your project and adapt them to your configuration (i.e. changing the cycle time of the MainFunction).

### 2.4 RTE configuration

If the Internal Behavior is configured as described in 2.3 the RTE will provide a solving action to automatically create the required RTE configuration. If a MainFunction has to be called by the RTE, perform the according Task Mapping afterwards. The RTE will then generate the Exclusive Area and MainFunction calls as described in the Internal Behavior.

## 2.5 CDD Configuration

For your own modules that need access to any PDU within the communication stack, use the so-called CDD module (Complex Device Driver), which is part of the SIP. For adding the CDD to the current configuration, open **Project Settings|Modules** and **Add** it via the blue plus.

🖏 Code Generation	Modules As	sistant		
🖷 External Generation Steps Sustom Workflow Custom Workflow Steps	Module Define Select the def	nitions inition of the modules to be added.		<b>•</b>
Modules Additional Definitions	Show only instantiated	those definitions, which are not yet in: I multiple times).	stantiated in the ECU configuration	on (or which can be
	Module 🗍	Definition Reference	Domain	
	Cdd	/MICROSAR/Cdd_AsrCdd/Cdd	Complex Driver	
	Com	/MICROSAR/Com	Communication	
	ComM	/MICROSAR/ComM	Network Management	
	Crc	/MICROSAR/Crc	Base Services	
	Dcm	/MICROSAR/Dcm	Diagnostics	
	Dem	/MICROSAR/Dem	Diagnostics	
	Det	/MICROSAR/Det	Base Services	
	EcuC	/MICROSAR/EcuC	Communication	
	EcuM	/MICROSAR/EcuM	Mode Management	

Figure 7 – Module Definitions CDD

In the Basic Editor you can define, where the CDD shall be placed within the communication stack.

🙀 Basic Editor 🛛 👚 Project Se	tting	s Editor						
🔄 < 🍓 Cdd 🔸 😼 CddComStackContribution								
🔍 <filter></filter>				•	Short Name:	CddComStackContribut		
▲ 😽 Cdd		٦ -						
VectorCommonData	■ VectorCommonData 母 Duplicate							
🖻 👑 Mcu	×	Delete containers						
👂 🬳 Rte		Create sub container		Cdd	lComIfUpperLayerCon	tribution		
	8	Copy Path		Cdd Cdd	IComMLowerLayerCo IGenericNmLowerLave	ntribution rContribution		
	<b>\$</b> 1	Show in		Cdd	PduRLowerLayerCont	ribution		
	<b>P</b>	Show properties		Cdd	IPduRUpperLayerCont	ribution		

Figure 8 – Configuring CDD

For further information how to configure the CDD, please refer to its Technical Reference (TechnicalReference\_Asr\_CddCfg5.pdf).



### 3.0 Settings.xml

The Settings.xml is an open interface to the CFG5. With this file the following configuration can be done:

Settings for external validation and generation steps

The file shall be placed to .\DaVinciConfigurator\Generators\<Msn> (i.e.

C:\Vector\CBD1200333\_D02\_V85x\DaVinciConfigurator\Generators\Mcu\Settings\_ExtGen\_Mcu.xml) and is automatically known at start of the CFG5.

The following example creates the same generator settings as the example in 2.2.1.

```
<Settings>
 <!-- external generator -->
 <Settings Name="com.vector.cfg.gui.core.generators.ExtGenSteps">
   <Settings Name="ExtGenSettings DrvMcu">
     <Setting Name="Active" Value="true"/>
      <Setting Name="CommandLine" Value="C:\Vector\CBD1200333 D01 V85x AddOn\Mcu\Mcu.exe"/>
     <Setting Name="GenerationParameters" Value="$(GenDataFolder) $(ProcessingEcuCFile) --generate"/>
      <Setting Name="SupportsStandAloneValidation" Value="true"/>
     <Setting Name="ValidationParameters" Value="$(GenDataFolder) $(ProcessingEcuCFile) --validate"/>
     <Setting Name="TransformationRequired" Value="false"/>
      <Setting Name="TransformationXsltFile" Value=""/>
     <Setting Name="TransformationOutput" Value=""/>
     <Setting Name="WorkingDir" Value="C:\Vector\CBD1200333 D01 V85x AddOn\Mcu"/>
     <Setting Name="SpecificAsVersionRequired" Value="true"/>
     <Setting Name="RequiredAsVersion" Value="4.0.3"/>
   </Settings>
  </Settings>
</Settings>
```

### 4.0 Integration Into the Build Project

AUTOSAR has introduced a mechanism to integrate standardized code into different  $\mu$ C and Compilers. To adapt the modules into a project with specific compiler and linker settings the files MemMap.h and Compiler\_Cfg.h have been introduced. If the module that shall be integrated into a build project that makes use of those mechanisms, these files have to be adapted.

For further information on this topic please also refer to TechnicalReference\_Asr\_MemoryMapping.pdf within the SIP.

### Example code for the following chapters:

```
#define MCU_START_SEC_VAR_INIT_8BIT
#include "MemMap.h"
VAR (uint8, MCU_INIT_DATA) Mcu_InitStatus = 0;
#define MCU_STOP_SEC_VAR_INIT_8BIT
#include "MemMap.h"
#define MCU_START_SEC_PUBLIC_CODE
#include "MemMap.h"
FUNC(void, MCU_PUBLIC_CODE) Mcu_Init (P2CONST(Mcu_ConfigType, AUTOMATIC, MCU_APPL_CONST) ConfigPtr)
{
...
}
#define MCU_STOP_SEC_PUBLIC_CODE
#include "MemMap.h"
```



## 4.1 Compiler\_Cfg.h

Add all used compiler abstraction defines from your module to this file, even if they are defined to nothing.

Required Compiler\_Cfg.h content for above given example:

```
#define MCU_PUBLIC_CODE
#define MCU_APPL_CONST
#define MCU_INIT_DATA
```

# 4.2 MemMap.h

Add all used memory abstraction defines from your module to this file.

Required MemMap.h content for above given example:

```
#ifdef MCU_START_SEC_VAR_INIT_8BIT
    #undef MCU_START_SEC_VAR_INIT_8BIT
    #define START_SEC_VAR_INIT_8BIT
#endif
#ifdef MCU_STOP_SEC_VAR_INIT_8BIT
    #undef MCU_STOP_SEC_VAR_INIT_8BIT
    #define STOP_SEC_VAR
#endif
#ifdef MCU_START_SEC_PUBLIC_CODE
    #undef MCU_START_SEC_PUBLIC_CODE
    #define START_SEC_PUBLIC_CODE
#endif
#ifdef MCU_STOP_SEC_PUBLIC_CODE
    #undef MCU_STOP_SEC_PUBLIC_CODE
    #undef MCU_STOP_SEC_PUBLIC_CODE
    #undef MCU_STOP_SEC_PUBLIC_CODE
    #define STOP_SEC_PUBLIC_CODE
    #define STOP_SEC_CODE
    #define STOP_SEC_CODE
    #define STOP_SEC_CODE
    #define STOP_SEC_CODE
    #define STOP_SEC_CODE
    #define STOP_SEC_CODE
```



# 5.0 Additional Resources

VECTOR TECHNICAL REFERENCES TechnicalReference\_Asr\_MemoryMapping.pdf TechnicalReference\_Asr\_CddCfg5.pdf

# 6.0 Contacts

# Germany and all countries not named below:

#### Vector Informatik GmbH

Ingersheimer Str. 24 70499 Stuttgart GERMANY Phone: +49 711-80670-0 Fax: +49 711-80670-111 E-mail: info@de.vector.com

#### United Kingdom, Ireland:

### Vector GB Ltd.

Rhodium, Central Boulevard Blythe Valley Park Solihull, Birmingham West Midlands B90 8AS UNITED KINGDOM Phone: +44 121 50681-50 Fax: +44 121 50681-69

E-mail: info@uk.vector.com

#### USA, Canada, Mexico:

#### Vector CANtech, Inc.

39500 Orchard Hill Place, Suite 550 Novi, MI 48375 USA

Phone: +1 248 449 9290 Fax: +1 248 449 9704 E-mail: info@us.vector.com France, Belgium, Luxemburg:

### Vector France S.A.S.

168, Boulevard Camélinat 92240 Malakoff FRANCE Phone: +33 1 42 31 40 00 Fax: +33 1 42 31 40 09 E-mail: information@fr.vector.com

#### China:

#### Vector Automotive Technology (Shanghai) Co., Ltd. Sunyoung Center Room 1701, No.398 Jiangsu Road Changning District Shanghai 200050 P.R. CHINA Phone: +86 21 6432 53530 Fax: +86 21 6432 5308 E-mail: info@cn.vector.com

#### Japan:

Vector Japan Co. Ltd. Tennozu Yusen Bldg. 16F 2-2-20 Higashi-shinagawa, Shinagawa-ku, Tokyo 140-0002 JAPAN Phone: +81 3 5769 7800 Fax: +81 3 5769 6975 E-mail: info@jp.vector.com

# Sweden, Denmark, Norway, Finland, Iceland:

VecScan AB Theres Svenssons Gata 9 41755 Göteborg SWEDEN Phone: +46 31 764 76 00 Fax: +46 31 764 76 19 E-mail: info@se.vector.com

#### India:

Vector Informatik India Pvt. Ltd. 4/1/1/1, Sutar Icon, Sus Road, Pashan, Pune - 411 021 INDIA

Phone: +91 20 2587 2023 Fax: +91 20 2587 2025

E-mail: info@in.vector.com

#### Korea:

Vector Korea IT Inc. 5F, Gomoas bldg. 12 Hannam-daero 11-gil, Yongsan-gu Seoul, 140-889 REPUBLIC OF KOREA

Phone: +82 2 807 0600 Fax: +82 2 807 0601 E-mail: info@kr.vector.com